Advanced HYPERtape Usage for Network Backup at the University of Kiel

Uwe H. Olias
Computer Center of Christian-Albrechts-University at Kiel
Kiel, Germany

ABSTRACT: *The University of Kiel continues to improve her HYPERtape usage for network backup with a UNICOS server by accomplishing a bundle of security strategies relying upon UNIX features, HYPERtape features and organizational steps. This talk describes the current situation with our strategies and gives a perspective for future enhanced possibilities. The talk is completed with the actual statistics table. Overall, the HYPERtape-Product may be of great interest for SGI/IRIX sites as DMF becomes available under IRIX OS.*

I. Introduction

To introduce newcomers to HYPERtape (those people who didn't hear anything about HYPERtape, e.g. at the last two CUG conferences), I'll give a rather brief snapshot of the product:

HYPERtape is a software solution for multi-vendor and multi-protocol platforms for an automated, unattended network backup and recovery. To achive this, it uses existing hardware and software components of the computer center network. The idea behind HYPERtape is to use a client-local backup-utility and send the data to be backed directly into a virtual-device. This virtual-device is represented by an existing network-protocol. The data get received by the server and stored on disk or cheap media according to the preset instructions. At this point, the backup itself is finished. We distinguish

- Backing-Data are written on disk: To put these data from there to cheap media, you need have some kind of HSM like DMF under UNICOS (this year available under IRIX), or AMASS and DataManager.

- Backing-Data are written directly from the network to cheap media: You need have HYPERtape/Media for the server platform. HYPERtape/Media is both a rather performant tool, a management aid for the backup and recovery part of work on the server, and it delivers segmentation of savesets and allows filtered restore.

The HYPERtape Concept

HYPERtape consists of three distinct logical components (which need not be consequently distinct physical instances):

- ControlNode

- ServiceNode (elsewhere known as client)

- BackupNode (elsewhere known as server)

The ControlNode uses the control path to initiate backup from a ServiceNode over the data path to the BackupNode and to the predestined storage media. These three nodes work together to create a connection that enshures a reliable network data transfer. You may choose any network protocol e.g. TCP/IP to establish sessions between the nodes. For performance reasons, it is recommended to choose the most performant path and protocol for data transfer between two instances. If the backup operation is completed or even aborted, the ServiceNode notifies the ControlNode about the status. This concept allows for a fully centralized network backup management and control.

The processes are organized in three of the most common backup related tasks: SAVE, RESTORE, and LIST. The SAVE operation creates a SAVESET containing the contents of the selected objects (normally disk volumes, a group of files, a user directory ...). From the point of view of the BackupNode, a saveset appears as one single file. The RESTORE operation restores a selected file from the remote saveset. The LIST operation lists the date and time a saveset was created, and the names of the files within the saveset.

(This paragraph has partly been derived from a HYPERtape-description by MultiStream Systems under personal permission. For more details, have a look at http://www.multistream.com)

II. HYPERtape Usage at UoK

We have been starting with HYPERtape in Fall 1993, but with quite another BackupNode concept. Still, our todays HYPERtape configuration concept has been planned already in the early phase from 1993 and acts according to the following

Organization and Backup Technique

We use one single SUN with Solaris as a ControlNode, but we have an option to use an additional CN, if necessary. A

CN is the scheduler, dispatcher or just the controller of a backup complex. Our one and only BackupNode is a CRAY Y-MP EL. The clients, called ServiceNodes, run their client-local backup-utility and put the files to be backed directly into the virtual device ftp-put, for example. Thus, they send their savesets over the network to the BackupNode CRAY Y-MP EL, where the savesets are primarily stored on a special filesystem on disk. Comparatively, a saveset is a container which exactely contains the group of files that have been saved within one backup job. From the point of view of the UNICOS BackupNode (the server), this container appears as one file. From the point of view of the ServiceNode (the client), this container is a virtual-device consisting of a group of backed files. When a saveset has been stored on the CRAY Y-MP EL (the BackupNode), the backup-part of the job is completed.

We have said the savesets are stored on the BN's disk. We chose a special filesystem for these savesets, and this filesystem is under control of UNICOS DMF. If the high watermark is reached, the savesets are migrated to the dedicated part of cartridges within the StorageTek ACS 4400. In case of a recall, a single file (or a list of files) will be restored from the container it is included in. That is, the according saveset is either on BN-Disk, or a dual state file, or it will be recalled from migration level to foreground storage, that's the disk.

Since this March, a distinct department of our University, the Department for Meteorology and Oceanographics, uses a DEC ALPHA as a separate BackupNode for a distinct set of workstations for their needs of some special archival. This BackupNode has HYPERtape/Media for Digital UNIX, and a Quantum 4700 Tape Loader for bulk storage. The Tape Loader is driven by a special tape-loader-software.

III. Backup Service at UoK

We support the following types of clients, totally numbered 58 when writing this paper:

HW Platform	SW Platform
• DEC Alpha	Digital UNIX
• SUN	Solaris & SunOS
• IBM RS6000	AIX
• SGI	IRIX
• SNI RM400	Reliant UNIX
• VAX , Single &Cluster	VMS
• Tapeless CRAY Y-MP EL	UNICOS

- Planned client support in 1997 at UoK:
  Windows NT, HP-UX, LINUX

Our current backup scenario looks like follows: We are a computer center that offers several central services to the campus. One of these services is backup. And our users, irrespective which platforms they are accustomed to, they are untrained and feel insecure handling restore of files. And, above all, they traditionally would forget about manual backup at all! Looking at the system administrators, even these feel backup and restore is more than they can handle. That are the reasons why every user of the CCN estimates our backup- and restore-service:

- **Backup:** We back an agreed part of their relevant files or filesystems according to an agreed scheduling, both full and incremental.

- **Restore:** The users send an e-mail to our central operators, which files from a specific date they want to have restored. This operator driven restore results in a special restore subdirectory under the users uid. Thus we even prevent a user succeeds in a self driven restore of an alien but sharable file.

Our backup- and restore-service runs like follows:

- We have one single ControlNode under Solaris, maintained by one single HYPERtape-Administrator. The operator crew is servicing on this ControlNode as restore-agent for the restore of single file(s).

- We have a central BackupNode CRAY Y-MP EL (which is the central fileserver in parallel). As bulk storage, we use a partition of 2,000 cartridges, 36 tracks for backup media within our StorageTek ACS 4400, four drives included.

- We support diverse workstations under several platforms as clients. They are located both, central and decentral.

- A restore is executed according to e-mail order of the user by the operator crew.

- If a filesystem failes totally, or a disk is destroyed, repair steps are under control of our systems department.

IV. Security Scenario

Our current HYPERtape-Scenario will extend to this one:

- May be we'll have a second ControlNode (solaris, probably Windows NT, but then both CNs will be NT)

- We plan to establish several decentralized fileservers with a dedicated decentralized tape robot each

- These fileservers have to support a regional network backup with HYPERtape, too

- There may be an individual RESTORE permission for every instance. Possible regulations are

  - All Ht-actions prohibited: Regulations may be per uid, per object or per client

  - All Ht-actions allowed: Regulations may be per uid, per object or per client

  - Single Ht-admittance for LIST, RESTORE, SAVE

  We'll use these distinguished gradations very thrifty and according to the arrangements with our users

- Topic dictate of any rule is security. We don't introduce any feature or permission which might affect security

And how do we get this? Well, there are several special aids and organizational steps, and they have to be applied very carefully and sophisticated and combined. These are offered

- by the concerned operating systems (ACLs, grouping, any access-rights, encryption, restricted shell). You only need to apply them

- by HYPERtape-Features

- by HYPERtape/Media-Features

- by organizational steps, additional tools (tcp-wrapper, e.g.)

And we now show how this (current and oncoming) scenario is configured.

1. *Root Privilege*

   There is absolutely no HYPERtape(/Media)-Software which is obliged to run under root-uid. We therefore don't install any HYPERtape-Components under root-uid, but under special HYPERtape service-uids. And these service-uids do nothing but HYPERtape work.

   - ControlNode: HYPERtape administration-tools and operators recovery-tools. No s-bit necessary, so we don't set it.

   - ServiceNode: For every user, the HYPERtape ServiceNode component must be able to find out the files to be backed, to back and to restore them. We therefore introduced a client-specific HYPERtape service-uid on every client and installed the platform specific HYPERtape ServiceNode-Software with appropriate s-bit settings if and only if requested. There is absolutely no need for s-bit within the LIST function!

   - BackupNode: On our todays BackupNode, we store savesets directely on disk and then migrate them to cheap media by DMF. Thus, we need not have any HYPERtape BackupNode-Software. But we have introduced a HYPERtape BackupNode uid which receives the savesets coming from the ServiceNodes. Additionally we have installed some private scripts with no s-bit, e.g. for deleting a saveset that has become a dead body for some reason.

   - In deed, having no HYPERtape/Media for UNICOS, we really need not have HYPERtape-Software on our BackupNode! In order to increase performance, we have separated the savesets into subdirectories specific to every client. This improves inode-search etc strongly, for we have some 12,500 savesets!

   - Within a short time, we'll introduce some special HYPERtape BackupNode uids which are holder ids for the savesets of departments or department groupings. This will support both, autonomy and keeping of privacy.

2. *Verification*

   - HYPERtape needs no trusted-host usage, we therefore avoid that feature

   - HYPERtape administration offers password-encryption from ControlNode to ServiceNode, and we use it. Of course!

   - We hope to have that encryption feature from ServiceNode to BackupNode, too. But this requires HYPERtape/Media for UNICOS, and password encryption towards BackupNode included. We'll demand that when ordering Ht/M for UNICOS! For that purpose, a new parameter has to be introduced in the Object DataBase, and the ServiceNodes and BackupNodes software have to be modified to recognize this new behaviour.

3. *Scope of HYPERtape-Operations*

   - Files that are backed with HYPERtape are residing in a container on background storage which is called a saveset. The name of that saveset is built by HYPERtape rule according to a parameterized template. The resulting name is only known to the HYPERtape administrator. And the saveset is only accessible to the BackupNodes HYPERtape service-uid. From there it is passed (or in case of filtered restore the appropriate file) to the ServiceNode, which receives the input extracted from a virtual-device. A saveset is therefore inaccessible to aliens.

   - We are going to inhibit dialog and batch jobs under HYPERtape service-uids, wherever possible.

- HYPERtape operations for a specific ServiceNode are performed
    - either by the ControlNode administrator, invoked from his uid at the ControlNode. Normally this administrator is the only person capable to handle all HYPERtape relevant things and operations,
    - or you may introduce special HYPERtape subadministrator uids on the ControlNode. For example, we did it for the operating crew with the same scope like the central administrator has with regard to SAVE, LIST, RESTORE. But we don't allow the crew to administer HYPERtape databases.
- On ControlNode, we plan to offer some more subadministrators, if required, with specific rights as are
    - scope is restricted to one ServiceNode, or list of it (= departmental administrator)
    - rights are restricted to LIST and RESTORE within their scope
    - allow right away SAVE runs within their scope
- Uids running on a ServiceNode can only treat files backed from that ServiceNode, but with strictly the same rights which the filesystem/directory and the client-local backup-utility would allow. That's what security is called. So there would not occur any security problem to allow user driven RESTORE. But our users don't want to go for a self driven RESTORE, and we don't grant that, accordingly. The only HYPERtape access rights are granted to the ServiceNode specific HYPERtape service-uid, with access to all HYPERtape ServiceNode operations and all files of that ServiceNode.
- The introduction of semi privileged HYPERtape rights for individuals (so called, but improved and diversified "secondary operator" from VAX/VMS ControlNode) is enabled within WINDOWS NT ControlNode as a user-profile feature, which will be ported to Solaris ControlNode. We'll take advantage of that.

4. *Restricted Shell*

   For every HYPERtape operational uid, we have introduced a job-specific restricted shell. So that keeps track that under HYPERtape service uid only HYPERtape work is done.

5. *HYPERtape related Uids and Passwords*

   Uids and login passwords are different for every instance and location. Frequent password changes according to complexity rules keep privacy and integrity clean.

6. *HYPERtape/Media*

   HYPERtape/Media uses free sockets. We'll then restrict the usage of these sockets to HYPERtape services only and HYPERtape partner workstations by means of the tcp-wrapper. Additionally, password-encryption is a basic need for security.

This is not the complete security scenario with HYPERtape at our site, of course. Some of these other security steps depend on tcp-wrapper, inhibit spoofing and so on. Perhaps you got an impression what things can be done. And we hope we did all of our best to make HYPERtape usage secure by applying anything that is offered by the operating systems, HYPERtape (and future HYPERtape/Media), useful utilities and possible organization.

V. Statistical Evaluation

This is an extract from our statistical evaluation:

• monthly # of Ht-Client-Jobs	9,000
• monthly backup-amount (MB)	200,000
• monthly job-connect (hours)	370
• number of clients	53
• total sum of backup-objects	370
• avarage size of a full-backup (MB)	290
• avarage incremental-size (MB)	13
• % from full	4.5

Recent changes concerning bulk storage:

During HYPERtape usage step 2, we used 2,000 cartridges with 250MB native capacity each. In HYPERtape usage step 3 we have the same amount of cartridges, but 800MB native capacity each (according to 36-track devices and E-Tapes). That is, our native capacity dedicated to HYPERtape increased from 500GB to 1,600GB, which is 1.6TB. This helps us to extend our support from some 60 to some 200 clients.

❖