COMPAQ
White Paper

May 2001
14MP-0501A-WWEN

Prepared by Industry Standard Servers Division

Compaq Computer Corporation

Contents

Compaq TaskSmart N-Series Appliance Network Attached Storage Planning Guide

Abstract: This guide is provided to assist Compaq customers in planning for Network Attached Storage (NAS) deployments using Compaq *TaskSmart*™ N-Series appliances. It is intended to assist customers in the following areas:

- Determine storage needs
- Understand both onsite and offsite planning issues
- Plan effectively for NAS performance requirements

Notice

Planning Issues

Performance Planning and Sizing

TaskSmart N-Series Default File Size Settings

For systems holding a large number of files that are smaller than 16 KB, disk space is wasted. In this case, disk space is wasted because each file allocation unit must be at least 16 KB (rather than the default 4 KB). Thus, in a system that houses millions of 2-KB files, for instance, 16 KB of actual disk space is used for each file. In such a case, no performance increase over the default is realized because the system has to perform a single I/O for every file access. That single I/O reads or writes a 16-KB chunk of data, most of which is actually empty space. In such scenarios, Compaq recommends changing the default file allocation unit size from 16 KB to 4 KB, or even smaller than 4 KB if the average file size is less than 4 KB, so that actual disk space is not wasted.

In contrast, when many files larger than 16 KB are stored on the TaskSmart N-Series appliance, disk space is not wasted because most files use more than a single file allocation unit. In these cases, the default 4-KB size incurs a performance penalty compared to 16-KB allocation unit sizes, because four I/O operations are required to read 16-KB chunks of data rather than just one I/O operation. In such cases, the larger file allocation unit size is advantageous because data for these larger files are read and written in larger chunks, resulting in fewer I/Os per file read or write. Moreover, because most files use more than a single file allocation unit, wasted disk space is minimized or non-existent. In general, as the average file size increases, the amount of wasted space decreases, so that it quickly becomes inconsequential. However, wasted space can be significant if the average file size is close to the file allocation unit size of the file system. For example, if average file size is 24 KB, then there is an issue of wasted space because an average file uses 32 KB of space on the file system, which means an average of 8 KB is wasted per file, or about 33.3% of the average file size. If, however, the average file size is 56 KB, then on a file system with a file allocation unit size of 16 KB, each file uses 64 KB of actual space spread across four file allocation units. The wasted space is still 8 KB per file on average, but this represents only 12.5% of the average file size. If the average file size is 136 KB, then each file uses 144 KB of space on average, spread across nine file allocation units. Again, the wasted space is 8 KB per file, but here that represents only 5.6% of the average file size.

The TaskSmart N-Series appliance defaults to a 16-KB file allocation unit size because enterprise applications currently in use tend to create much larger files than older applications. One simple case that demonstrates this point is that of word processors. Older word processors tend to output relatively small files. In some cases such word processors only understand ASCII text formats. Modern word processors include features to store metadata about text formatting, page formatting, and other features. Such programs result in larger output files that very often are much larger than 16 KB. Thus, selecting 16 KB as the default file allocation unit size makes sense if one assumes that most customers deploy NAS to service storage needs of applications deployed within the past few years. For those customers who require a storage solution for many of the much smaller files, a change from the default 16 KB to 8 KB or even 4 KB is recommended.

In practice, the client operating system has a large impact on the effective performance improvement of using a relatively large file allocation unit size. When using Microsoft Windows NT 4.0 Workstation clients, for example, there is a marked improvement in performance when

increasing the file allocation unit size from 4 KB to 16 KB. However, Microsoft Windows 2000 Professional clients do not demonstrate such a large increase. In neither case does an increase in file allocation unit size beyond 16 KB obtain much of a performance increase.

Figure 1 and Figure 2 show the impact of default file allocation unit size and operating system on performance. The figures show results from test runs of the NetBench Enterprise Disk Mix test run with 48 clients spread across four 100BaseTX full-duplex, switched Ethernet networks. In one series of tests all 48 clients were deployed with Windows NT 4.0 Workstation, Service Pack 4. In the other series of tests, all 48 clients were deployed in the same environment with Windows 2000 Professional. In the case of Windows 2000 Professional clients, client-side caching was not enabled. Tests were run in all cases against the same TaskSmart N2400 appliance, with fourteen 18.2-GB drives of attached storage. A full disclosure is included in the appendix, “Test Methodology,” at the end of this document.

When Windows NT 4.0 Workstation clients were used for testing, there was a significant difference in performance depending on the file allocation unit size that was used, as shown in Figure 1.

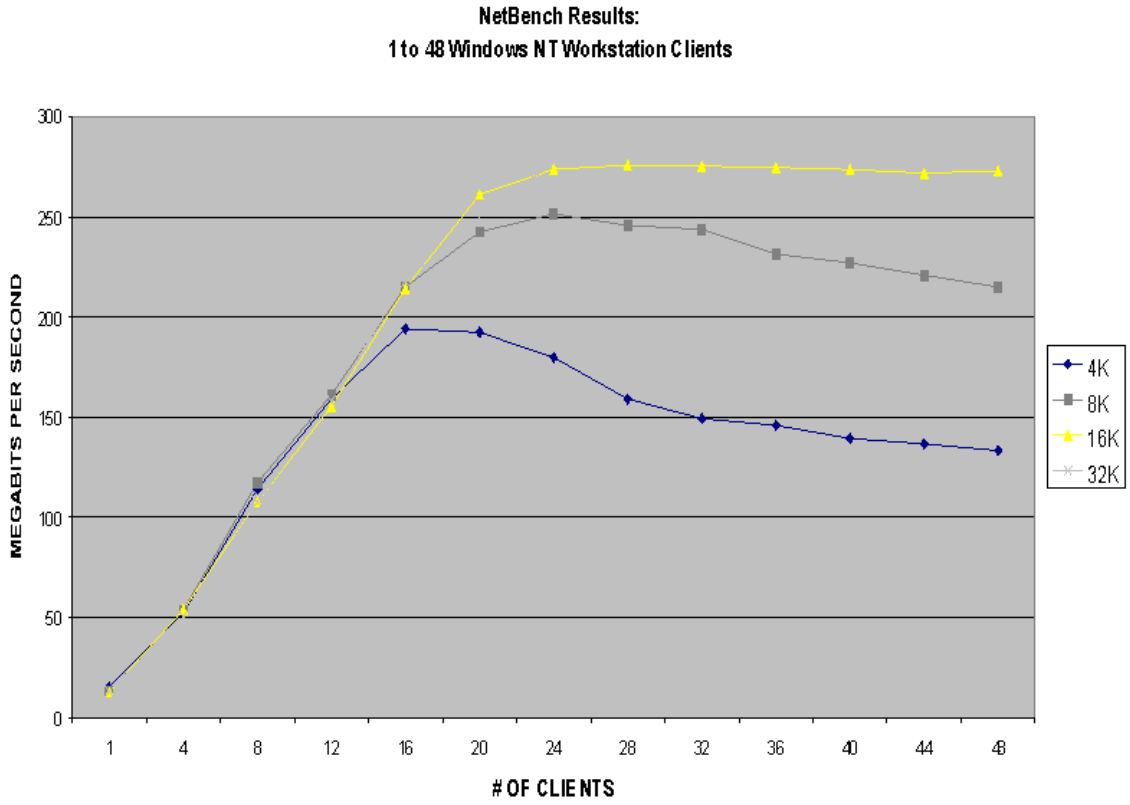

Figure 1. NetBench client results with 4-KB, 8-KB, 16-KB, and 32-KB file allocation unit size and Microsoft Windows NT 4.0 Workstation clients

In contrast, when all Microsoft Windows 2000 clients were used for testing, there was little difference in performance regardless of the file allocation unit size that was used.

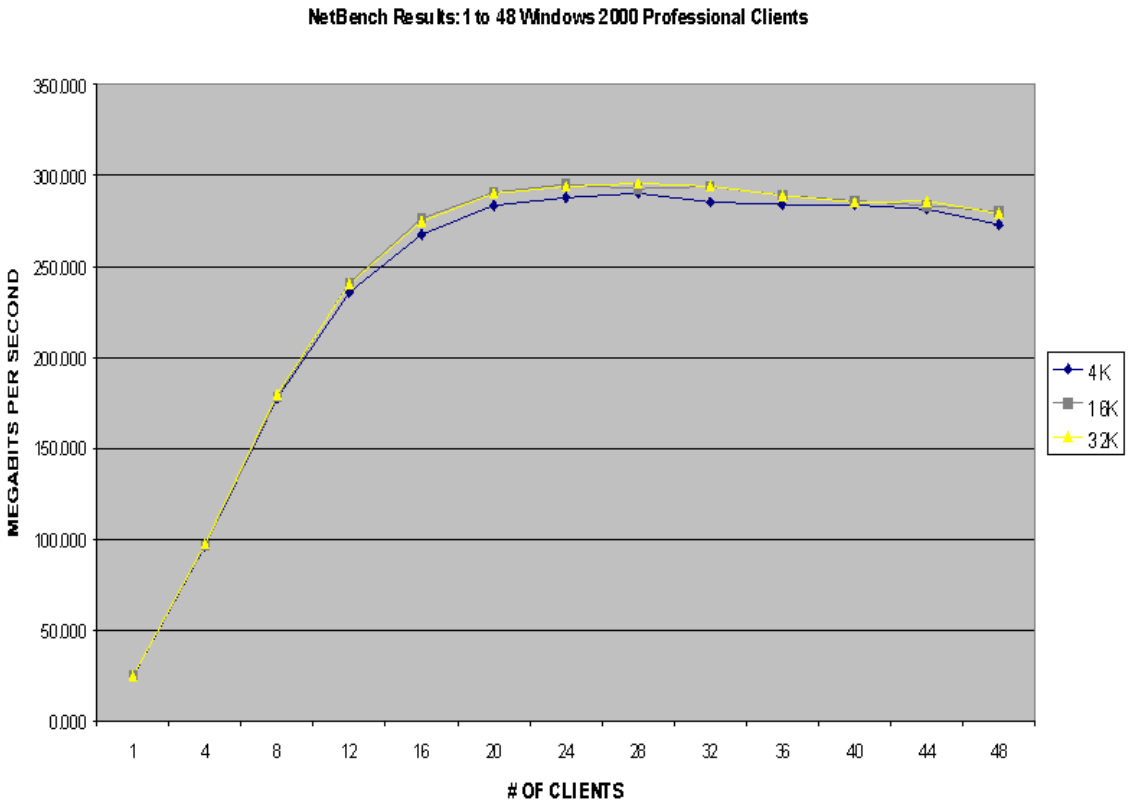

Figure 2. NetBench client results with 4-KB, 16-KB, and 32-KB file allocation unit size and Windows 2000 Professional clients

As shown, the actual effect of file allocation unit size depends greatly on the client mixes used. As organizations migrate toward Windows 2000 Professional clients, even small 4-KB file allocation unit sizes allow for high performance. While older versions of Windows, especially Windows NT 4.0, are still used, planning for a 16-KB file allocation unit size is recommended. Note that even in the scenario where all Windows 2000 Professional clients were used for testing, there is a slight performance increase between 4-KB and 16-KB file allocation unit sizes. Therefore, 16 KB is the recommended size. Moreover, it is the default built into the product when virtual disks are formatted using the SWVR Snapshot Manager interface. Again, it is important to remember that deployments using very small files should plan to use file allocation unit sizes that are small, perhaps even smaller than 4 KB if the average file size is known to be less than 4 KB. Note that using allocation units higher than 4 KB disables the ability to directly use the compression and encryption settings on files and volumes in NTFS file systems.

Impact of Spindle Count on Performance

Note: Both 18.2-GB and 36.4-GB drives are supported in the TaskSmart N-Series appliance, whether deployed exclusively as Ultra2 drives, as Ultra3 drives, or in combination.

Note: Raw physical disk drive size is calculated by taking the total number of bytes available for storage and dividing by 1,000 to get KB, 1,000,000 to get MB, and 1,000,000,000 to get GB. Thus 18.2 GB denotes 18,206,425,088 bytes. If a base of 1024 is used to calculate successively KB, MB, and GB, then the description of drive capacity changes. In such cases, 18,206,425,088 bytes are successively equivalent to 17,779,712 KB, 17,363 MB, and 16.96 GB. In either case, the total raw storage capacity is identical. Only the abbreviated expression of that capacity is different.

When planning for optimal file serving performance, the number of disk spindles (disk drives), necessary to maintain an optimum performance level must be determined. As a general rule, the greater the number of spindles available, the greater the performance achieved. There are some caveats, however.

Specifically, the SWVR software included with the TaskSmart N-Series appliance does not stripe across RAID arrays that get incorporated into a storage pool. Rather, RAID arrays are concatenated together to form larger storage pools. The net effect is that no performance increase is achieved until such time as the physical capacity of one RAID array is exceeded so that new data are written to the second and subsequent RAID arrays in a pool. Even in this case, no effective spindle count increase is achieved unless the (generally older) data from the previous RAID arrays are still accessed and modified. To maximize the effective number of spindles available, the following different strategies are available:

- Using Compaq Array Configuration Utility to incorporate more disks into a single RAID array or Logical Drive set
- Spreading out data shares across multiple logical drives (carving)

Incorporating More Disks into a Single RAID Array/Logical Drive Set

Hardware incorporates the extra spindles into a single logical set and stripes reads and writes across the set. Using the Compaq Smart Array 5300 Controller included with the TaskSmart N-Series, it is even possible to cross RAID channels to incorporate additional spindles into a single RAID array.

Customers can choose either RAID 5 or RAID Advanced Data Guarding (ADG.) For those using a fault-tolerant RAID 5 configuration, Compaq recommends limiting the array size to fourteen physical disk drives to reduce the likelihood that a multi-drive failure could bring down the entire disk subsystem, resulting in data loss.

It is also possible to dynamically grow the size of an existing storage array to incorporate a greater number of spindles in the array. Such capacity expansion does not affect the size of any logical drives in the storage array. Rather, it spreads any existing logical drives across the new number of spindles and makes the additional space available for the extraction of additional logical drives. For instance, if a TaskSmart N-Series appliance is initially deployed with just seven drives in a single RAID 5 array that presents a single logical drive to the operating system, then the appliance has an initial data capacity of 218.4 GB, as shown by the following equation:

7 drives - 1 drive worth of parity = 6 drives * 36.4 GB = 218.4 GB

As the customer increases the number of spindles and the capacity, choosing RAID ADG adds protection against multiple-drive failure. RAID ADG also allows an array of more than fourteen physical disk drives with only the capacity of two drives set aside as dual sets of distributed parity data. RAID ADG allows continued access to the data in the event of a two-drive failure. Other than the following calculation, the examples in this paper are for RAID 5 environments. A RAID ADG example is calculated as follows:

7 drives - 2 drives worth of parity = 5 drives * 36.4 GB = 182 GB

Therefore, the single logical drive has a total capacity of 218.4 GB of data space spread across seven spindles. If seven more spindles are subsequently added, one can create a separate array and extract a second logical drive of 218.4 GB in size. An alternative, however, is to incorporate the additional seven logical drives into the existing array. This increases the space available in the existing array by 254.8 GB, since only one drive worth of parity space is required regardless of the size of a RAID 5 array. This does not, however, increase the size of the existing logical drive in the RAID array, but it is nevertheless possible to extract one or more additional logical drives from the same array whose total size amounts to 254.8 GB. This can be an effective growth strategy when the appliance is deployed in an environment where usage lulls occur from time-to-time, because one can plan to add drives and grow storage arrays during such a usage lull to mitigate against the performance loss that occurs during the time-consuming process of growing the storage array.

A fourteen-drive RAID 5 volume has a greater chance of data loss than the original seven-drive volume, however. When using the Smart Array 5300 controller, the logical drives could be converted to RAID ADG fault tolerance at the same time as the new drives are added. In this case, the second logical drive would only be 218.4 GB since the capacity of two drives are now used for fault-tolerance data, but the chances of data loss would actually be reduced due to the higher level of fault tolerance provided by RAID ADG.

Planning for such growth is a requirement. It can take hours to grow a seven-drive storage array into a fourteen-drive storage array. During this time, performance can suffer because the data, currently spread across seven drive spindles, must essentially be copied so that it spreads out across fourteen drive spindles when the operation completes. The array can still be used during this time, but because of the low-level reorganization of data on the drive that takes place, one must expect significant performance degradation.

IMPORTANT: For details on the mechanics of growing the size of an existing storage array using the Compaq Array Configuration Utility, refer to the *Compaq TaskSmart N2400 Administration Guide*. Regardless of the number of drives deployed on the appliance, Compaq recommends that no more than fourteen drives be incorporated into any single storage array, unless the RAID ADG option is chosen for the array.

Spreading Out Data Shares Across Multiple Logical Drives (Carving)

A complementary strategy for increasing effective spindle count is to spread out data shares across multiple logical drives. In this scenario, logical drives are carved from RAID 5 arrays of four to fourteen drives each. Then, a single logical drive is imported into each SWVR pool. Virtual disks are then extracted from the pools and mounted on a drive letter. Finally, Common Internet File System (CIFS) and Network File System (NFS) shares are spread across the virtual disks so that the net effect is to spread reads and writes across all the physical drive spindles in the system. Using the maximum configuration of the TaskSmart N2400 appliance as an example, it is possible to envision a high-performance, highly reliable system with fifty-six hard drives and

2 TB of raw storage capacity if using 36.4-GB drives. One can create, for instance, eight RAID 5 logical drives, each of which contains seven physical drives, or create one RAID ADG logical drive. Using the SWVR Snapshot Manager tool, one can then import these logical drives into eight separate pools and extract one or more virtual disks from each pool, mounting each on its own drive letter. Finally, shares can be spread across all eight virtual disks so that reads and writes are effectively distributed across all spindles in the system. This layout is summarized in Figure 3.

An additional side effect of this strategy is to mitigate downtime in the event that data must be restored from tape. When eight arrays of seven drives each are used for data storage, the maximum amount of data loss per logical drive is limited to the size of one of the drives, or 36.4 GB, leaving 218.4 GB of available storage, as shown in the following equation:

7 drives - 1 parity drive = 6 drives; 6 drives * 36.4 GB per drive = 218.4 GB

Using DLT7000 technology, it is possible to restore this amount of data in about 1.5 hours (four drives) or about 3 hours (two drives). At the same time, all other data on the system is still available to users or applications. Moreover, since Enterprise Backup Systems that utilize tape libraries can generally use only a single tape device per disk device, having several disk devices available can increase the number of tape drives available for a backup or restore. More tape drives generally means greater throughput potential, decreasing both backup windows and downtime required to complete a restore.

Note: For throughput rates on DLT tape drives, see the Compaq website:

www.compaq.com/products/storageworks/Tape-and-Optical-Storage/dlt-tapearray2Quickspec.html

All calculations are based on a 5 MB/s throughput rate per drive and the assumption that 1024 MB = 1 GB.

For detailed information on TaskSmart N-Series appliance backup and restore considerations, see the Compaq website:

www.compaq.com/TaskSmart/

Figure 3 demonstrates one way to carve disks in this fashion. This particular example uses the same online spare for each of eight RAID arrays spread out across fifty-six drives, the maximum configuration for the Compaq TaskSmart N2400 appliance. All RAID arrays are the same size, seven drives, except for one which is six drives in size to make a single physical drive available as an online spare for all arrays in the system. From each RAID array, a single logical drive is carved. The seven larger arrays have a single logical drive of approximately 218.4 GB in size. The single six-drive array has a logical drive of approximately 182 GB in size. The single online spare is then available to automatically begin a drive rebuild on any of the arrays which has a single drive failure. Obviously, failed drives should be replaced as soon as possible so that the single spare can revert back to acting as a spare rather than being actively used by a particular array which has experienced a single drive failure. When failed drives are replaced, the corresponding RAID 5 arrays immediately begin rebuilding the data and parity metadata on the replaced drive, and the shared online spare immediately and automatically reverts to its role as spare available to all RAID arrays for which it is configured as such.

Each logical drive is then inserted into its own storage pool as the only storage unit in the pool. One or more virtual disks is then extracted from the pool and made available to the TaskSmart N-Series appliance. After this task is done, the administrator can create directories on each virtual disk and share them out as CIFS shares, NCP shares, NFS exports, or any combination of thereof.

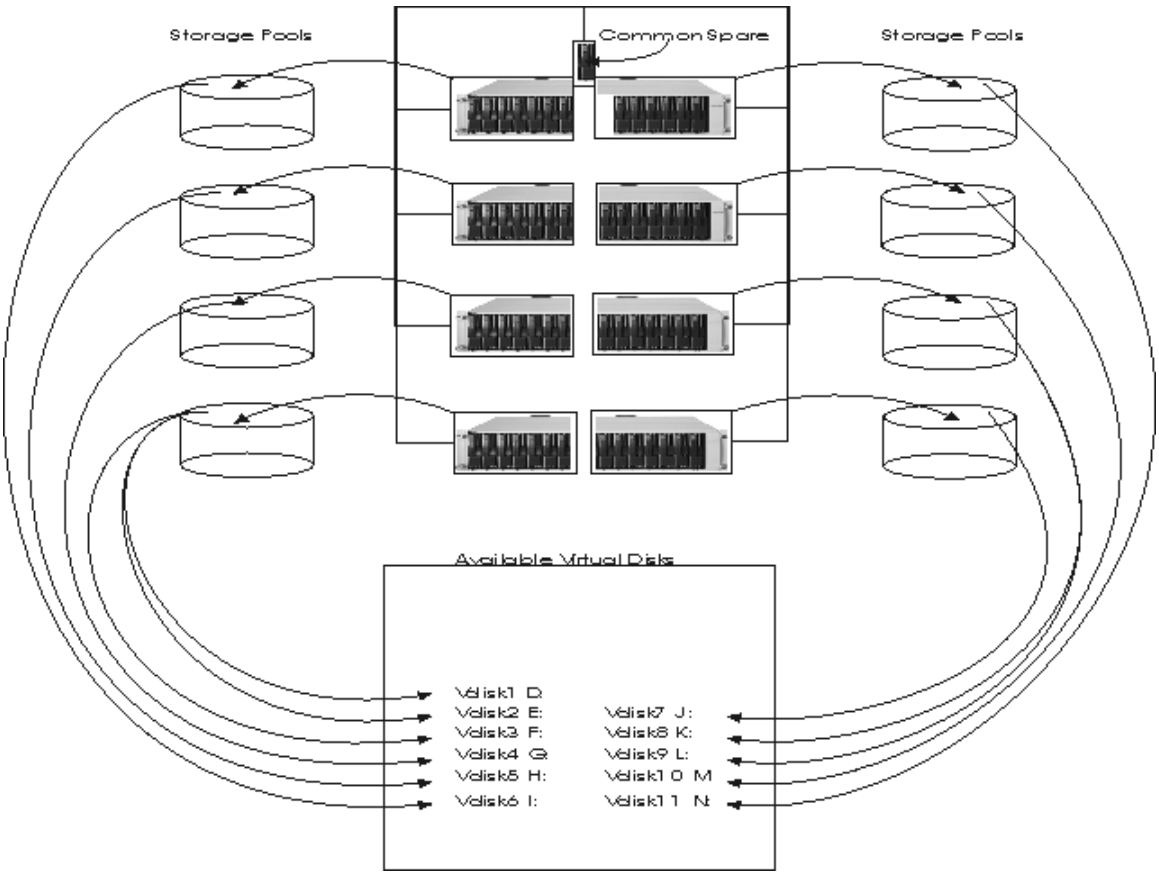

Figure 3. Sample disk carving with eight storage pools

It should be noted here that this example is focused on achieving maximum performance rather than flexibility in growing virtual disks after initial allocation. SWVR does not stripe across all of the storage units in a particular pool. Rather it concatenates the storage units together, using each in succession as each prior one fills up. Because of this, an optimally performing disk carving strategy places a single storage unit into each pool. Consequently, no disk concatenation takes place, just as in the example shown in Figure 3 and Figure 4, which documents a similar carving strategy using fourteen-drive RAID sets rather than seven-drive sets. Following this strategy, and spreading CIFS shares, NCP shares, and NFS exports across all virtual disks, allows for reads and writes to be spread fairly evenly across all disk spindles in the system, which in turn allows for optimal performance. As mentioned, however, the tradeoff here is flexibility in growing the size of the virtual disk. If a full 2 TB of disk storage is carved initially, then each virtual disk can only grow as large as the available space in the pool, minus adequate space set aside for snapshots.

Alternate configuration strategies, such as allowing more than a single storage unit into each pool, allow for the administrator to add storage units to a given pool as space is used up. However, since this introduces disk concatenation into the system, data write performance can suffer. Over time, however, there might not be any performance penalty in data read performance. The reason for this is fairly straightforward. All new writes can only be written across the disk spindles of a storage unit which has space available, and the writes can only be striped across the disk spindles in a single storage unit. The same is true of disk reads, except that reads can read data from any storage unit that already has data written to it. If the older data written to the first storage unit(s) in the concatenation continues to be accessed at the same time as the newer data, which is written only to the last storage unit in the concatenation, then reads are effectively distributed across the disk spindles of multiple storage units. Thus, in systems where reads happen more frequently than writes, the added flexibility of adding multiple storage units to a pool likely is more important than guaranteeing that all disk I/O is spread evenly across all disk spindles in the system. On the other hand, if disk writes predominate, then the administrator has to take into account the potential performance degradation of concatenating storage units when deciding on a disk carving strategy.

A second example described here shows how one might carve a full TaskSmart N-Series appliance disk configuration using fourteen-drive storage arrays rather than seven-drive arrays. This configuration has the advantage of reducing wasted parity space without appreciably increasing the risk of drive subsystem failure due to the simultaneous failure of two or more drives in a single RAID set. Since only four storage arrays are used to set up the fifty-six physical hard drives, only 145.6 GB (4 * 36.4 GB = 145.6 GB) of raw disk space is lost to store parity information. Twice this amount of space (291.2 GB) is lost to hold parity data when seven-drive RAID sets are used instead. At the same time, fourteen-drive RAID sets have been tested in Compaq laboratories to make sure that the risk of subsystem failure is still small. While the risk of drive subsystem failure when using fourteen-drive storage arrays is roughly twice that of seven-drive arrays – and at the same time restoration of data to a fourteen-drive array could potentially take twice as long as it does for a seven-drive array because there is more potential data to lose in a larger array – such risks are still fairly small. Thus, fourteen-drive RAID sets can be effective tools for maximizing performance by using all available disk spindles without increasing risk in the way a twenty-eight-drive RAID array might.

If the need arises for such a large storage array, Compaq recommends that ADG be used to minimize the chances for lost data by increasing the fault tolerant parity sets to two. This uses the equivalent of two disk drives in each storage array, but with large numbers of drives in the arrays, the actual percentage of space lost to parity is still relatively small.

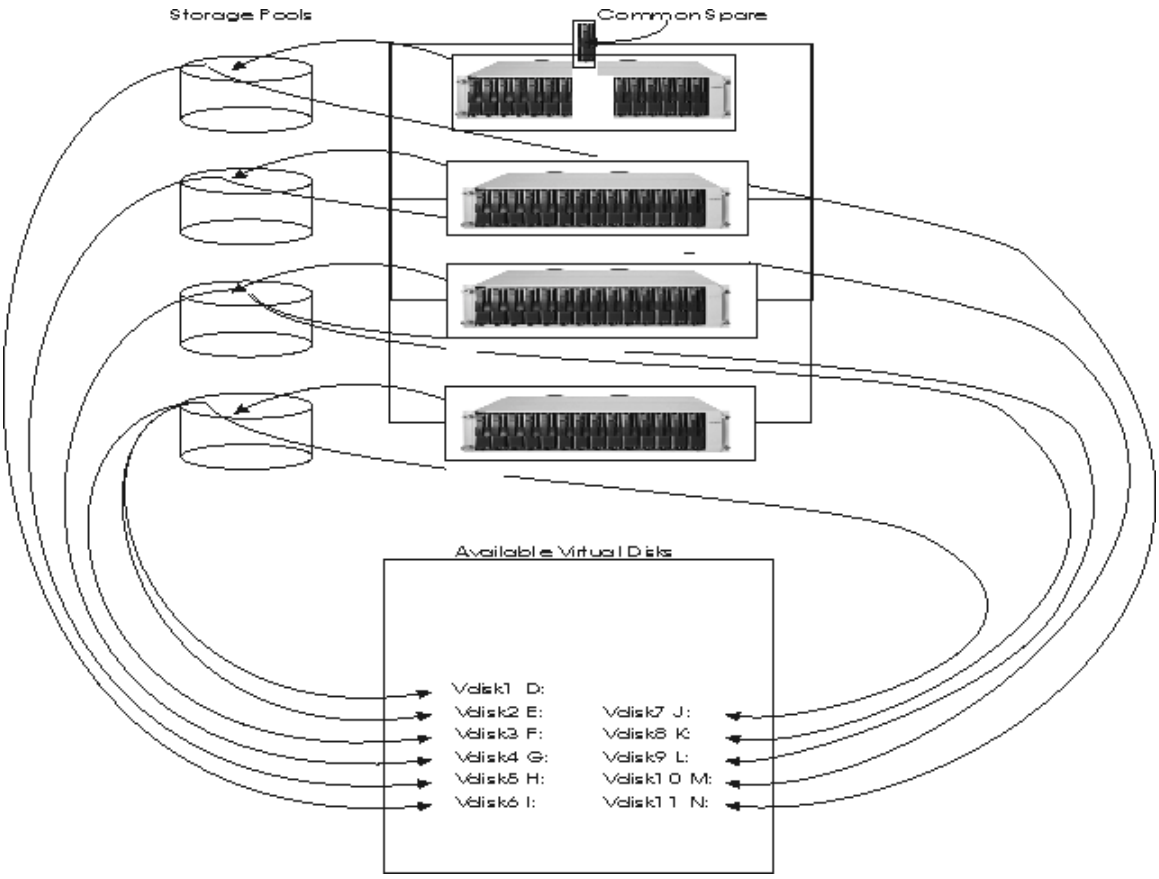

Figure 4. Sample disk carving with four storage pools

In the example shown in Figure 4, the same number of virtual disks is extracted from storage pools as in the seven-drive array example. Only in this case, those virtual disks are extracted from just four storage pools, each of which is a container for a single logical drive on a fourteen-drive RAID array. This setup is a bit easier to manage, because every virtual disk is automatically spread across fourteen spindles (except, of course, those extracted from the first storage pool, which has sacrificed a single spindle to serve as a hot spare for all RAID arrays). Since each virtual disk is spread across fourteen spindles, managing share allocation for optimal performance (as described in the following section) is simpler.

Managing Shares for Optimal Performance

There is one more consideration when thinking about systems where data writes happen frequently. If data is written often but rarely deleted, then the amount of space used tends to increase over time. As used space increases, it might begin to push up against the size limit of the single storage unit in a particular pool, especially if snapshots are used. Thus, it might be impractical in the long term to build disk configurations that perfectly optimize performance. Figure 3 should therefore be viewed primarily as an ideal that can be deviated from. One might use fourteen-drive RAID arrays to increase the size of each storage unit in a pool, thus guaranteeing a large number of spindles across which to read and write data while also ensuring that the upper size limit of single-storage-unit pools is high. In cases where space availability absolutely trumps performance, the administrator can add multiple storage units into some pools while directing the most performance intensive data to those pools that have just one storage unit available. Hybrid strategies are definitely encouraged. The important issue is that the

administrator recognizes the potential tradeoffs between performance and disk growth flexibility so that the best configuration for his or her environment can be carved.

Storage Sizing

IMPORTANT: The external storage enclosures that are part of the TaskSmart N-Series appliance can accommodate both Ultra2 and Ultra3 LVDS SCSI hard drives. Thus, it is possible to combine both 18.2-GB and 36.4-GB hard drives in the same storage enclosure or across storage enclosures attached to the same appliance server. If the larger hard drives eventually supplant the smaller drives over time, customers may have both types of drives in their appliances.

It is possible, and even desirable, to deploy such a hybrid, or mixed, system. However, it is best to create each storage array out of drives of the same size. If two drives of different sizes are deployed within the same storage array, then the larger drives are incorporated at the same effective size as the smaller drives, sacrificing the excess space. Thus, a single 18.2-GB hard drive in the same array as thirteen of the 36.4-GB hard drives causes the waste of 236.6 GB of space, as shown in the following equation:

36.4 GB - 18.2 GB = 18.2 GB * 13 = 236.6 GB

While this example is extreme, it best illustrates the problem with such a deployment strategy. A better strategy is to create arrays out of 18.2-GB drives that include ONLY 18.2-GB drives, while reserving available 36.4-GB drives for arrays composed exclusively of same-size drives. When carving disks this way, space is lost only for spares, parity, and snapshots. No space is lost to force effective drive sizes into the least common drive size in the array.

If a hybrid deployment model is used, it is best to have an appropriately sized online spare for each collection of RAID sets. For instance, if there are four RAID sets that use 18.2-GB drives and four sets that use 36.4-GB drives, then an 18.2-GB drive should be used as the online spare for all of the four RAID sets using 18.2-GB drives, and a 36.4-GB drive should be used as the online spare for all of the four RAID sets using 36.4-GB drives.

Along with server performance, sizing the disk storage capacity of a file-serving appliance is important. Obviously it is necessary to plan for storage needs to accommodate existing data that is consolidated on the server. However, it is also important to take into account the expected growth over time of the data set size and the average size of each file in the data set. Both impact the effective disk storage required.

First, examine data growth rates by looking at an example. Assume that a TaskSmart N-Series appliance is obtained to consolidate file storage currently spread across five general-purpose file servers. Each of those file servers has 100 GB of attached or internal storage. Some subset of this available storage actually has data stored on it. Table 1 illustrates an example:

Table 1. General-Purpose Server Storage Example

Server	Available Disk Storage (GB)	Used Disk Storage (GB)	Remaining Disk Storage (GB)
1	100	50	50
2	100	60	40
3	100	80	20
4	100	90	10
5	100	60	40
Total	500	340	160

Table 1 suggests that sizing storage requirements is a fairly straightforward task. It is necessary to consolidate 340 GB of data onto a TaskSmart N-Series appliance. Therefore, it would appear to be reasonable to purchase enough storage to accommodate 340 GB of data, and a bit more for future growth. The 500 GB of available disk storage that used to be spread across five general-purpose servers seems to be a reasonable amount. However, further analysis is necessary, as shown in the remainder of this section.

First of all, just as many general-purpose servers do, the TaskSmart N-Series appliance supports hardware RAID. RAID 5 is a configuration that maximizes available disk space, performance, and fault tolerance on the disk subsystem. However, RAID 5 requires one disk worth of space to be dedicated to holding parity data. It often makes sense to allocate an additional disk as an online spare so that disk recovery can happen without user intervention when a single drive fails. Of these extra disks, an extra disk for parity must be allocated for each RAID array that is carved. The online spare can be allocated as one spare for all RAID arrays, one spare per RAID array, or it can be left out.

As shown, before deciding upon how much attached storage to purchase with a TaskSmart N-Series appliance, it is necessary to plan for an appropriate RAID configuration. The previous discussion on the impact of spindle count on performance also has implications on effective data space. Continue with the current example, assuming that 500 GB of effective data space is needed to accommodate 340 GB of existing data, using RAID 5 arrays with a single online spare shared among all arrays.

The first decision to be made is how many arrays should exist. More specifically, how many spindles are to be allocated per array. As in the performance discussion, more spindles translates to more space, and more spindles per array means that fewer spindles are required for parity. However, the effective reliability of the underlying disk system is lessened as more and more spindles are added to any individual array. This is because a RAID 5 array fails with possible data loss if two or more drives fail in any given array. Comparing two seven-drive arrays to a single fourteen-drive array clarifies the situation. With two seven-drive RAID 5 arrays, two drives could potentially fail without impacting the integrity of the drive subsystem. As long as the two failed drives are in different storage arrays, no data loss occurs. However, with a single fourteen-drive RAID 5 array, the failure of two drives causes the drive subsystem to fail, with possible data loss. Using RAID ADG effectively eliminates the risk of increasing the RAID set size.

While fourteen drives is still a reasonable RAID 5 array size, increasing the size to twenty-eight, forty-two, or fifty-six drives steadily increases the potential for multiple-drive failure within the same array. Furthermore, as array sizes increase, so also does the potential restore window for data recovery from offline media in case of disk subsystem failure. Fourteen drives worth of data may be restorable in just a few hours, and users of data resident on other fourteen-drive arrays can continue using that data while the restore takes place. However, the time required to restore twenty-eight, forty-two, or fifty-six spindles worth of data is much greater. And since the failure of a twenty-eight-, forty-two-, or fifty-six-drive array takes with it a greater chunk of the potential data storage capacity of the TaskSmart N-Series appliance, fewer arrays are available to fewer users while a restore takes place to recover from a potential disaster. Thus, planning for storage space allocation, precisely because it requires planning for disk carving, must take into account issues ranging from performance to backup and restore windows. Keeping these considerations in mind, Compaq recommends seven-drive arrays, unless RAID ADG is chosen. Based on this choice, the following statements are considered in planning for reasonable space:

- A single seven-drive RAID 5 array can provide 218.4 GB of data space.
- 7 drives - 1 drive worth of parity = 6 * 36.4 GB = 218.4 GB

Thus, to provide 500 GB worth of available data space, at least three 7-drive arrays must be deployed.

The next step is to decide whether online spare(s) are to be used. Compaq recommends that at least one online spare be allocated for the entire drive subsystem. Allocating a single online spare for each RAID array increases the fault tolerance of the entire subsystem since no RAID array shares its online spare with any other. However, the storage capacity loss of this strategy is large, so sharing a single online spare across multiple RAID arrays provides an increase in fault tolerance without sacrificing as much data storage capacity.

In the current example, a single online spare is allocated to be shared between the three RAID 5 arrays. This means that a total of twenty-two drives must be purchased, seven each for the three arrays, plus an additional drive to be used as an online spare. Each TaskSmart N-Series storage enclosure accommodates fourteen drives, so a total of two storage enclosures must be purchased for the twenty-two-drive storage solution. A total of six additional bays remain available for future drive purchases, and two more storage enclosures can be obtained later if disk growth requires it. In any event, the three seven-drive arrays provide a total of 655.2 GB of usable storage space.

The next item to take into consideration is whether snapshots are to be used. When setting up the TaskSmart N-Series appliance, RAID arrays are imported into SWVR storage pools, and virtual disks are extracted from these pools and mounted on a drive letter. This disk virtualization makes it possible to take a snapshot of data on disk. A snapshot is an instantaneous copy of pointers to data on disk. When the data on disk change, the original blocks are copied to a snapshot area of the same storage pool in which the parent virtual disk is located. In systems where a large volume of data changes rapidly after snapshots are taken, it is possible to overrun the storage capacity of the pool. Thus, it is necessary to plan ahead to allow for adequate space for each snapshot stored.

In theory, the only way to guarantee that enough space is available in a storage pool for snapshots is to allocate enough space in the storage pool for data and an identical amount of space for each snapshot to be stored. In practice, however, many environments do not ever modify much of the original data. Thus, only enough additional storage pool space must be modified for the various copies of blocks to be maintained in the various snapshots. Since SWVR copies data to snapshots at the block level, only changed blocks – rather than entire files – get propagated into snapshots. This can further reduce the effective amount of space required for snapshots. Obviously, in cases

where many snapshots are kept online at the same time, the blocks – which can change multiple times and therefore be copied into multiple snapshots – can consume a lot of space. However, when only two or three snapshots are kept online simultaneously, this space requirement is less. In such cases, Compaq recommends that an additional 30% of disk space beyond storage requirements be allocated for snapshots.

In the current example, 655.2 GB of effective storage space is allocated as RAID 5 arrays. These arrays are imported into one or more storage pools, keeping in mind the performance implications discussed previously. Then, one or more virtual disks are exported providing 655.2 GB of potential disk space for storage. If there is no plan to ever use snapshots, then this amount of storage space exceeds our goal of 500 GB and leaves us some breathing room. However, if snapshots are to be used, this scenario requires further examination. Using the recommended figure of 30% of storage space to be reserved for snapshots obtains the following result:

655.2 GB - (0.3 *655.2 GB)

= 655.2 GB - 196.56 GB

= 458.64 GB

That is not quite the 500-GB figure from the previous five general purpose servers. It is close, and it may be enough to get started, but since there is already 340 GB of data to consolidate, only 118 GB of capacity remains. A fourth RAID 5 array is called for. The fourth array is only six drives in size, since twenty-two of the twenty-eight available drive slots are already used. This 6-drive array has a capacity of 182 GB, as shown by the following equation:

6 drives - 1 drive worth of parity = 5 drives * 36.4 GB = 182 GB

Adding this to the current capacity obtains 837.2 GB of useful disk storage space. The snapshot calculation now results in the following:

837.2 GB - (0.3 *837.2 GB)

= 837.2 GB - 251.16 GB

= 586.04 GB

Now, the 500-GB goal has been calculated with room to grow. More importantly, the TaskSmart N-Series appliance still has capacity to add two more external storage enclosures with an additional twenty-eight drives total. Using the same online spare for all additional RAID arrays permits adding an effective additional drive capacity of 611.52 GB, as shown by the following equation:

(586.04 + 0.7) * 36.4 GB (since we do not need an additional online space) = 611.52 GB

In terms of raw data space, a full complement of fifty-six 36.4-GB Ultra3 SCSI hard drives provides 2038.4-GB, or about 2 TB, of raw space. This compares to 1019.2 GB, or about 1 TB, of raw space available when using fifty-six 18.2-GB Ultra2 or Ultra3 SCSI hard drives. Twice the amount of space per drive translates to twice the amount of total raw and effective space available for carving into RAID arrays and virtual disks with the 36.4-GB drives.

The full 2-TB raw capacity can be carved up in other ways, using the guidelines illustrated in this example, to achieve slightly different total capacities and slightly different levels of fault tolerance. Notably, if fourteen-drive arrays are used instead of seven-drive arrays, then only four drives worth of space is lost to store parity data, instead of the eight drives worth of space spread across eight seven-drive RAID arrays. In this case, one gets back a total of 145.6 GB of raw data space, each of which sacrifices 30% of raw space for snapshots, if spread across four storage pools, providing a net gain of 101.92 GB of effective data storage space:

145.6 GB * 0.7 = 101.92 GB

Thus, when deploying fourteen-drive RAID arrays of 36.4-GB drives, the effective disk capacity is 1299.48 GB, or about 1300 GB of available storage space.

Specifically, planning for eight seven-drive RAID arrays, each of which shares the same online spare, effectively achieves seven arrays that provide 218.4 GB of data space. This data space is available after taking into account the single drive worth sacrificed to hold parity data in a RAID 5 set and one array that provides 182 GB of data space because it also loses one drive to provide a common online spare.

7 drives - 1 drive for parity = 6 drives * 36.4 GB = 218.4 GB

6 drives - 1 drive for parity = 5 drives * 36.4 GB = 182 GB

Thus, available disk space in such a seven-drive array strategy is 1710.8 GB. When allowing for 30% of space to be reserved for snapshots, the net effective space available for data is 1197.56 GB, or about 1200 GB. Note that this effective available disk space in a maximum configuration is also exactly twice the available disk space when using seven-drive RAID arrays to carve up fifty-six of the 18.2-GB drives (previously outlined 598.78 GB * 2 = 1197.56 GB). Thus, when using a fourteen-drive RAID array disk carving strategy, with a single, common online spare, net effective disk space when using fifty-six of the 36.4-GB hard drives is exactly twice that provided by the same strategy with 18.2-GB hard drives, or 1299.48 GB. This effective storage space when allowing for snapshots is about 100-GB more than when carving eight arrays of seven drives each.

For additional information on RAID disk carving issues, see the Compaq website:

www.compaq.com/support/techpubs/whitepapers/0291-0799-a.html

For additional information on the Smart Array 5300 Controller, see the Compaq website:

www.compaq.com/products/storageworks/smartarray-controllers/SA5300index.html

For additional information on SWVR, which is used in the TaskSmart N-Series appliance to provide disk virtualization and snapshots, see the Compaq website:

www.compaq.com/products/storageworks/swvr/swvr_index.html

Security

Security planning is important for a product that integrates the security models of both NFS, NCP, and CIFS. Windows clients, Novell clients, and UNIX clients address file security in slightly different ways. These sets of clients generally authenticate against separate account databases. UNIX users log on to their workstations by authenticating against a local **passwd** and **group** file or by authenticating against a centralized account database on a Network Information Service (NIS) server. Windows clients also authenticate either locally or against a centralized account database. However, when authenticating locally, there is no inherent unification of security models even across different Windows machines. And when authenticating against a centralized account database, Windows machines hit a Windows Domain Controller rather than an NIS server. Novell clients will also authenticate in the same manner as Windows clients. User accounts in a domain or in a workgroup model will be NetWare-enabled, which allows them to authenticate to Microsoft servers.

This disparity in security models can cause difficulties in merging files from a heterogeneous environment onto a single unified file server. Furthermore, the slight differences in the file security models for NFS, NCP, and CIFS can provide additional challenges. Even when only

Windows clients, only Novell clients, or only UNIX clients are sharing files on a single centralized file server, there can be challenges to providing adequate data security. This section discusses strategies for addressing these security challenges, in both homogeneous (NFS-only, CIFS-only, NCP-only) and heterogeneous environments.

Homogeneous CIFS Environments

The TaskSmart N-Series appliance is based on Windows 2000 technology. Therefore, integration with CIFS environments is fairly simple on the surface. CIFS is the native network file-sharing protocol for Windows clients, and while UNIX and other clients also can use CIFS shares, it is common for CIFS-only environments to consist primarily or entirely of Windows clients.

CIFS is a network protocol that requires a client authentication before allowing access to network file resources. A state is maintained between client and server that includes the client's user and group membership credentials, which are used by the server to grant or deny access to file resources.

Two common deployment environments exist. A "Workgroup" environment is one in which each CIFS user logs in locally to his or her local workstation, usually providing a username and sometimes a password. A "Domain" environment is one in which account credentials are stored on a Domain controller, and logins are authenticated against this centralized account database. Sometimes other authentication mechanisms are used, such as a Smart card or a fingerprint. The difference is the location of the account database.

Workgroup Environment

The Workgroup strategy can be effective and less costly for smaller deployments than using a separate server or servers as Domain controllers.

In a Workgroup environment, a username context is created on the local machine. This context is important because it is also the default username sent over the network when such a client attempts to connect to a CIFS share. If a password is used for local authentication, then this password is also sent as the default password for the authentication. If both username and password match an entry in the TaskSmart N-Series appliance local account database, then access is provided to that CIFS share based on the CIFS client's username and that username membership in a group in the appliance local account database. This distinction is important. A user, **CIFSUser**, might be a member of the **Administrators group** in the login workstations local account database, but if **CIFSUser** is a member only of the **Users group** on the appliance, then group-level access is determined by membership in the appliance **Users group**, not the login clients **Administrators group**. The server alone determines what access is granted or denied to a particular user or group.

Note: Some CIFS clients, such as Windows 95 or Windows 98, do not require a password or even a username for local access. In such an environment, these clients must nevertheless provide a username context and an appropriate password in order to authenticate themselves for access to a CIFS share on the TaskSmart N-Series appliance. Windows 95 and Windows 98 must typically provide the username context at the login screen. While this username is not typically used to grant or deny access to local workstation resources, it is the only context that can be used for authentication against a CIFS file server. A different password can be provided, but the same username is used.

On the other hand, Windows NT and Windows 2000 clients can provide an arbitrary username context, regardless of the username provided when logging in to the local account database on the client workstation. However, that user's group membership is determined by the configuration of the account database on the TaskSmart N-Series appliance. This is standard behavior of a server in a Workgroup environment so that the server, and not the client, determines what access is granted or denied to a particular group or user in that group.

There are some obvious limitations to Workgroup deployment. Effectiveness depends on managing accounts both on local workstations and on the TaskSmart N-Series appliance. Account names must be managed so that they are identical in both places. Furthermore, passwords can get out of sync, requiring users to remember multiple passwords. When passwords expire, only Windows NT and Windows 2000 clients have a way of changing their password on the TaskSmart N-Series appliance. (To change a password on the appliance, a client presses the **Ctrl+Alt+Delete** keys, then clicks **Change Password**, and enters the appliance host name in the **Domain** field.) Clients under other operating systems may experience the following situations:

- Users are granted temporary physical access to the appliance as the appliance Administrator so that they can change their passwords on a regular basis, or
- Administrators change the password to something requested by the user, allowing administrators to know each user's password, or
- Passwords never change.

All of these strategies can create security problems because users and administrators cross over their roles, or because a policy of periodic password changes is not implemented. Furthermore, as the number of users of the appliance grows, maintaining username synchronization across many workstations and the TaskSmart N-Series appliance can quickly become unmanageable.

Domain Environment

For larger deployments, a Domain environment is the better choice.

In such an environment, one or more Windows NT Servers are typically deployed as Primary and Backup Domain controllers, or a Windows 2000 server or servers are deployed as Primary or Backup Active Directory controllers.

Note: This document does not attempt to explain all the possible deployment planning scenarios of these two environments, but instead attempts to focus on those areas that are specific to the CIFS file-sharing function. More information about Windows Domain controllers can be found in the Domain controller help files or on the Microsoft website at

www.microsoft.com/

Additional information is available in the TaskSmart N-Series appliance administration guide, available with the product itself or from the TaskSmart website at

www.compaq.com/TaskSmart/

Further information about planning, deployment, and operations of Windows Domain controllers is available on the Compaq *ActiveAnswers*™ website in the eBusiness Infrastructure Solutions Area at

www.compaq.com/ActiveAnswers/

A Domain environment provides a far more systematic authentication model. Instead of multiple account databases, a single account database is maintained on a Domain controller. Users log in to their local workstations by authenticating against the central Domain account database. In many cases, those local workstations have machine accounts in the Domain database, and users are not even permitted to authenticate against the Domain unless their workstations also have accounts. Under some operating systems, machine accounts are not required, typically because the OS does not have this functionality. However, machine accounts under these operating systems also must supply a username and password at login in order to gain access to Domain resources, including resources that might be physically located on a TaskSmart N-Series appliance that is a member of the Domain.

In cases where the clients are not workstations but servers, such as Windows NT or Windows 2000 servers running an application like IIS that accesses network file resources, a login takes place. In these cases, the service either runs within the context of a Domain user or a service resource (such as an IIS virtual directory) that provides username and password information in order to properly gain access to the TaskSmart N-Series appliance file resources whose access is controlled by the centralized Domain controller. In either case, CIFS resources are provided only to properly authenticated users, even if the authentication is provided as part of application server configuration.

Note: Windows 95 and Windows 98 clients do not necessarily need to log in to a Domain in order to gain access to local workstation resources. They are, however, required to provide username, password, and Domain name information at login if they are to be granted access to any resources in the Domain, including those that might be physically located on a TaskSmart N-Series appliance that is a member of the Domain.

Windows NT and Windows 2000 client workstations must generally have machine accounts in the Domain before users can authenticate against the Domain from the particular machine. This is also true of the TaskSmart N-Series appliance, which is based on Windows 2000 technology. In either case, users can log in locally to their workstation account database or they can log in to the Domain. Logging in locally requires that users provide their Domain credentials when attempting to access Domain resources. If users initially log in to the Domain, these credentials are provided implicitly whenever an attempt is made to access a Domain resource.

User and Group Account Strategies for Effective CIFS File Sharing

The TaskSmart N-Series administration guide documents in detail the particulars of both the TaskSmart N-Series file system and its CIFS share security models. The important thing to note here is that the CIFS security model is a subset of the file system model, and the two models integrate. This integration can be used to the administrator's advantage in allocating file-sharing privileges to specific users and groups. A single share, for example, might be configured so that anyone in an organization has access to it. Then, at the file system level, more specific access privileges might be set up so that only members of a particular sub-organization or project have access. This allows for the inherent integration of file system and CIFS security to be used to the system administrator's advantage.

This same hierarchical way of thinking can be used to organize users and groups in such a way that the hierarchical placement of files on the TaskSmart N-Series appliance, using both CIFS share and file system security configurations, can streamline the process of administering access to file resources. Users of network file resources can generally be grouped logically based on the organizational division they work in, or what project or projects they work on. A specific user in the marketing department might be in the documentation group that reports to Mary Jones. Within this group, this user might be working on the product documentation for WidgetX and BigBoxY. This user might also provide support for the development of advertising collateral for PylonZ over in John Smith's advertising group. Other employees in Mary Jones's organization likely are assisting in the development of product documentation for WidgetX and BigBoxY. At the same time, other employees in other organizational divisions may be providing support for these efforts or their approval may be required before the documentation, and thus the product, can ship.

These organizational and specific project groupings can serve as a model for user and group organization on the account database that authenticates users of the TaskSmart N-Series appliance CIFS resources. WidgetX, BigBoxY, and PylonZ could actually be used as the basis for group names. Since some users require read-only access, and others require both read and write access to files associated with those products, an administrator might create the WidgetXROnly, WidgetXRW, BigBoxYROnly, BigBoxYRW, PylonZROnly, and PylonZRW groups to signify that those groups are granted read-only or read-write access to a particular file resource. Each user can then be made a member of the appropriate group so that he or she has read-write or read-only access, as appropriate, to the file resources that are already configured to grant appropriate access to the particular group. If further granularity of access privilege is required (the *Compaq TaskSmart N2400 Administration Guide* offers details of more than a dozen specific user access privileges that can be individually configured), then additional groups that are project-

and access-specific can be created to accommodate this need. Furthermore, there may be network file resources that are not project-specific. To accommodate this situation, an organizational-specific group can be created, such as MaryJonesOrgROnly or MaryJonesOrgRW.

In all cases, groups, not individual users, must be used as the access-level metaphor. In most organizations, there are many users, several of whom form part of a sub-organization or work on a particular project. If one configures access to file resources at the group level only, then two benefits are realized:

- Fewer access configurations need to be done because there are generally fewer groups than users.
- As users enter into or depart from a particular organization or project, they can simply be added to or removed from the corresponding group without the need for reconfiguring the particular access configuration of the file resource.

Finally, one needs to consider that many projects are part of an overarching larger project, and many organizations are themselves part of larger organizations. WidgetX, BigBoxY, and PylonZ might be part of the AlphabetSoup parent project. Mary Jones's and JohnSmith's groups have already been defined as parts of the marketing group, which in turn might be part of a larger organization in the company. This same planning mechanism can apply here, as well. Users can be members of multiple groups, or better yet, groups can themselves be members of other groups, allowing for an elegant hierarchical metaphor to be used as the planning principle for the organization of user and group access rights. For example, WidgetXROnly, BigBoxYROnly, and PylonZROnly might all be members of AlphabetSoupROnly, which can be used to configure access privileges on file resources that need to be available in a read-only format to all three sub-projects.

Each environment has its own particular needs. Any hierarchical organization method needs to guard against creating so many levels of organization that only one individual is a member of a given group. If this happens in planning for CIFS access privileges, then it defeats the purpose of creating a hierarchical model. One might as well grant or deny access privileges to individual users. Conversely, there are always exceptions to any strict hierarchical model. For example, an employee in Mary Jones's organization may directly support one or more projects being directed by the divisional vice president. As such, that individual user's account may need to be given file access privileges directly, because the hierarchical user and group model cannot take into account this specific exception to the rule. However, these examples must be understood as exceptions. The general hierarchical model is quite useful in minimizing administrative problems when configuring user and group access privileges.

Homogeneous NFS Environments

A homogeneous NFS environment presents a similar set of challenges. There is, however, one fundamental difference. Where CIFS is an authenticated protocol that passes a full user credential to the server, NFS does not authenticate the user. Instead, NFS grants or denies access to network file resources based on the host identification (name or IP address) of the client attempting to connect, regardless of the specific user or group membership currently attempting access from the connecting client. The user ID (UID) and primary group ID (GID) are passed along with each NFS request so that the NFS server knows who the user is and what group that user is primarily a member of. This is necessary so that the NFS server can apply a proper file system credential (known as an Access Control List, or ACL, on the TaskSmart N-Series appliance) to the file or directory being created or modified. It is also necessary so that the NFS server knows whether to grant or deny access to a particular file based on the standard UNIX World-Group-User security

lists, which are stored on the file system of the appliance and presented to NFS clients. NFS is not the most secure of protocols, but it can be quite useful in environments where other security measures (such as physical login security) are taken.

A UNIX user is generally required to log in to the machine that he or she wishes to use. This login is authenticated in one of two ways:

- Against a local **/etc/passwd** and **/etc/group** file pair which contains username, encrypted password, and group information about every user on the local workstation or server, or
- Against a Network Information Service (NIS) server which contains user, group, and encrypted password information in a centralized account database.

Other security models exist. Sometimes local account information is stored in an LDAP directory or database instead of in **passwd** and **group** files. Centralized UNIX account databases are also sometimes stored in a more secure NIS+ server. Regardless, passwd/group files and NIS servers are the most common UNIX account databases, and in any event, they are the only authentication mechanisms supported for integration with the TaskSmart N-Series appliance.

One of the difficulties with **passwd/group** or NIS authentications, which is the fundamental security problem with NFS, is that it is possible for a user to imitate another user fairly easily. Imitation is accomplished by simply setting up a new UID and GID entry in a client's local account database, associating those entries with a username for which he or she knows the correct local password, and sending that user's UID and GID information across the wire in NFS requests. This is most dangerous, of course, when someone imitates the root file, sending UID 0 across the NFS wire in order to gain access to root file resources on the NFS server. This is the reason why NFS servers generally require that explicit access to root-level resources be set up so that root users can access NFS file resources over a network. NFS does not send an account database context (such as the local machine Domain name as in the case of CIFS) over the network when clients connect to servers. Therefore, one can, in fact, imitate UID 0 (or some other UID) without specifying *which* UID 0. Since no password is required to connect, the server has to trust that the client has already been properly authenticated against an account database that the server may or may not know about.

As a practical matter, NIS is more secure than **passwd/group** files, but it still has some security gaps. Most specifically, unlike a Windows Domain controller, the machine that becomes a member of a NIS domain does not have a specific machine account in the Domain. Thus, while it is impossible for another machine with the same host name to imitate a Windows machine's membership in a Windows Domain (because each machine – and therefore the machine account on the Domain – has a unique SID that is independent of the host name), no such mechanism exists for NIS. NIS knows about hostnames but not unique machine identifiers. In effect, in NIS, a machine points at an account database and says, "I trust you," and the account database assumes that the machine is not lying about its identity. In a Windows Domain controller, the account database allows machines to become members and then demands that all member machines identify themselves appropriately when logging in. The trust relationship is thus reversed.

NFS is a popular file sharing protocol for UNIX environments, and if adequate policy and physical security measures are taken (such as firewalls and intrusion detection software sitting between the NFS server and a public network), it can be made reasonably secure. This discussion elaborates on some of the best practices for setting up NFS user and group security and integrating it with the TaskSmart N-Series appliance so that file-level security happens in a systematic and expected manner. For information on additional security measures that can be taken to for the network on which the appliance operates, including planning, deployment, and

operations information for firewalls and intrusion detection solutions, see the Compaq ActiveAnswers website at

www.compaq.com/ActiveAnswers

Integrating with a Passwd/Group File Account Database

Generally, when **passwd** and **group** files are used for UNIX security, deployments are small. Unfortunately, small deployments have a tendency to become large. While there is no substitute for adequate planning of centralized account databases as UNIX deployments increase over time, there are some steps that can be taken to mitigate against problems that occur with growth, even when using **passwd** and **group** files initially for small deployments.

1. Use the same **passwd** and **group** files for all UNIX deployments.
2. When a change needs to occur in the file, push those changes out to every UNIX machine deployed.

These strategies require two stipulations:

- Individual UNIX users are not granted root access to their workstations, or if they are, strict policies prohibit direct manipulation of the **passwd** and **group** files. While this can guard against malicious modification of the files for the purpose of causing problems on the network, in most cases this prohibition is imposed merely to keep the files on the various machines from getting out of sync. Keeping the files in sync is critical so that only one user of a given username exists.
- All updates to **passwd** and **group** files are done on a routine basis by pushing the modifications out to each machine and appending them at the end of the existing files. This practice prevents routine maintenance from unwittingly changing the encrypted **passwd** information stored in these files that users can (and should) change from time to time. If users access their accounts from multiple machines, they can either modify passwords in multiple places, or the administrator can push out the user information from the user's usual machine to all machines on a periodic basis. Better yet, when this becomes a concern, the administrator can set up a NIS server and import all existing account information into this server so that it is all stored in a central repository.

The synchronization of the **passwd** and **group** account repositories is critical if NFS is set up on the network. Synchronization is critical because as long as there is only one UID 500, for example, and the UID 500 is associated with the username jdoe, then the NFS servers on the network always properly interpret UID 500 when granting or denying access to jdoe's files or directories. Synchronization is also critical as UNIX deployments increase over time. In some NFS environments, separate **passwd** and **group** files are maintained in each participating server. As long as these separate files are synchronized, the users and groups always use the same name throughout the environment.

One way to synchronize these files is to use an NIS server, which effectively places a single copy of the information in the **passwd** and **group** files into a central location against which each user or group in the environment authenticates file system permissions. The net effect is to create a single namespace for the user and group information so that unique user and group names can be enforced. When proper synchronization or centralization of user and group information is planned for and implemented, there is no chance of different users or groups being assigned the same name because the single namespace is the sole authority.

However, if **passwd** and **group** files are not adequately centralized or synchronized, the effect is to create multiple namespaces for user and group information. In such a scenario, it is possible for

a particular user or group name in one namespace to be assigned to a different user or group in the other namespace. NFS, however, has no formal rules for namespace authentication that might differentiate NAMESPACE1:User1 from NAMESPACE2:User1. The first User1 encountered is assumed to be the user in question. Thus, this opens up the possibility of username and groupname collisions. Aside from the security problem (perhaps one User1 has root privilege while the other User1 has only guest privilege), such a scenario creates problems when the namespaces are unified. Even assuming that one can determine which User1 name properly applies to which physical user of the environment, there still is the task of creating unique names when the namespaces are unified. Thus, one User1 might need to be changed to User11, and that user will have to get used to a different username in order to authenticate.

The TaskSmart N-Series appliance can import **passwd** and **group** files so that it knows what username is associated with what UID and what group name is associated with what GID. Since NFS does not require a password for authentication, and since the appliance does not generally behave as an NFS client, the encrypted passwords never need to be synchronized on the appliance. However, any time a user or group is added, the **passwd** and **group** files must be pushed out to the appliance and imported anew. This process can be set up to occur automatically. For more information about this process, refer to the *Compaq TaskSmart N2400 Administration Guide*.

Integrating with an NIS Account Database

A NIS server provides a significant advantage over **passwd** and **group** files in that it centralizes all account information in a single logical location. Each UNIX machine on the network merely points at the NIS server and authenticates there. This capability has the following significant advantages over **passwd** and **group** files:

- All accounts are unified in a single account database, which means that only the administrator of that database can gain access to the database.
- Information does not need to be pushed out to every UNIX machine on the network. Since the account information is already centralized, the only thing necessary is that all UNIX machines are pointed at the NIS server for login authentication.

There are, however, some issues.

Unlike a Windows Domain controller, a NIS server establishes no context for usernames, group names, and passwords. Thus, if someone is able to imitate a root user and gain access to the NIS server in an unauthenticated fashion, such as by mounting an NFS export, then that user can wreak havoc. As a best practice, then, one should never export file systems from a NIS server unless one trusts all users and computers on the network. At the very least it is a good idea to avoid exporting the file systems on which the NIS account databases are physically located.

However, in many cases, NIS servers sit behind a structured tier of firewalls and intruder detection software is deployed at strategic locations between the NIS server and any public network that might provide access to it. Those machines that do access NIS are therefore trusted in a typical deployment, as are the users of those machines. Thus, the access control issue, while real, may not be the most critical security issue with a NIS deployment. It is far more critical to keep multiple NIS servers in sync with one another.

In many organizations, small groups become larger over time by merging with other small groups within the organization. In other scenarios, entire organizations merge with one another. In either case, each organization or group can bring with it its own account databases, including NIS servers.

When organizations merge, the lack of NIS domain context can create a large problem. While it is possible and even encouraged to operate multiple NIS servers on the same network and within the same organization, for seamless operation, those NIS servers must maintain a partition of the total user and group namespace.

This means that if NIS Server 1 has user jdoe and NIS Server 2 has user jdoe, then an inherent infrastructure conflict exists. A Windows Domain controller establishes context by attaching a specific user or group name to the Domain in which it is resident and requiring both a Domain name and a user or group name to determine which jdoe, for example, is being referenced. However, NIS makes no such provision for context. If there are 10 jdoes spread across 10 NIS servers, then only order of lookup determines which jdoe a user authenticates against. Thus, the associated password and UID for a particular jdoe in a multi-NIS environment is determined entirely by the particular NIS server that a given user authenticates against.

Similarly, when multiple groups of the same name are spread across multiple NIS servers, the user membership within a particular group depends upon the definition of such group in the particular NIS server that is being pointed at by a NIS client. Such a setup can quickly create problems because there is no systematic way for determining who is who over a network.

This situation also creates a security problem. Users could authenticate against a NIS server that provides near superuser access for their jdoe login and then access another machine on the network using an unauthenticated protocol (RSH or NFS, for example) and wreak all sorts of havoc. A bigger problem created is the organizational headache. Users of NIS will not necessarily know where their password is located at any given time. Even if users know the particular password on NIS1, there is no guarantee that they know it on NIS2. In fact, NIS2 may represent a completely different jdoe that recently came in with the merger.

These organizational problems represent a significant data retention or security problem. In a single environment, administrators do not generally want different users of the same name to collide with one another inadvertently. Such collisions are not malicious. They are just inconvenient, and could result in mysteriously disappearing or changing files. A second user, thinking them unimportant, may accidentally delete or modify another user's files.

The best practice is to keep usernames unique across multiple NIS servers, and if the usernames become nonunique because of the merger of two merged organizations, then a plan should be put into place to verify unique usernames and group names as soon as possible. As existing usernames and group names are retired, another opportunity exists for ensuring uniqueness over time. If necessary, a complete username and group name review may be regularly scheduled.

Homogeneous and Heterogeneous NCP Environments

Novell users local to the TaskSmart N-Series appliance can be given access permission to files and folders on the appliance by installing Services for NetWare on the TaskSmart N-Series appliance. This can be applied in a Domain or Workgroup model.

Domain Compared to Workgroup

The TaskSmart N-Series appliance can be deployed in a Workgroup environment and in a Domain environment. CIFS is a session-based, stateful network file sharing protocol. When mapping a network drive or a client machine, a user sends a login credential to the server. When authenticated, the server provides the corresponding access to the user. When a TaskSmart N-Series appliance is deployed in a Workgroup environment, all user and group account access permissions to file resources are stored locally on the TaskSmart N-Series appliance. By contrast, an appliance deployed into a Domain environment uses the account database from the Domain

controller, with user and group accounts stored outside the appliance. The appliance integrates with the Domain controller infrastructure. In a Domain environment, it is required that Services for NetWare is installed to allow user accounts to be NetWare-enabled. The installation of Services for NetWare on the TaskSmart N-Series appliance permits the Novell shares to be created locally on the appliance.

Client Access to Servers

Because the TaskSmart N-Series appliance will be emulating a Novell server, it will allow Novell redirectors to authenticate to the TaskSmart N-Series appliance. This can be accomplished in two ways:

1. The workstation can log in directly to the TaskSmart N-Series appliance. To accomplish this, the **Tree** and **Context** fields within the redirector must be blank. The **Server** field must read the name of the TaskSmart N-Series appliance after FPNW has been installed. For example, the server name is ALAMO. After the installation of FPNW, the server name for Novell redirectors is ALAMO_FPNW. This will not change the hostname for Microsoft networking; it only provides a name for Novell networking.
2. If login scripts have been set to the Novell volumes on the TaskSmart N-Series appliance from within NDS, then passthrough authentication occurs. Furthermore, if the user accounts and passwords are identical on both the Novell and TaskSmart N-Series appliance, no additional logins are necessary. However, if the user accounts and passwords are not identical, then a Novell login box appears asking the client to authenticate to the TaskSmart N-Series appliance.

Once FPNW has been installed on the TaskSmart N-Series appliance, the user has the ability to create Novell shares (volumes). To create a share:

1. Go to the MMC, select **Shared folders**, and then select **Shares**.
2. Right-click on **Shares**, and then select **New share**. The user now has the option to create a Microsoft or Novell NetWare share. This procedure is the only way to create a Novell share. This procedure can not be accomplished by selecting **Start, Explore**.

To manage the shares, select the **Shares** tab in MMC. From this tab the user can manage the permissions or security of the share. To view the current connections to a specific share or to disconnect users from a share, the administrator goes to a control panel and opens FPNW. This process works exactly the same way to create a Novell share from a snapshot. Upon the installation of FPNW, a **Sys** volume is created. The **Sys** volume is required for all NetWare servers and must not be removed. Users can access shares through drive mapping, by browsing the network, or through the command line.

Heterogeneous CIFS/NFS Environments

When planning for heterogeneous environments, security issues become more complex.

In a typical scenario, administrators plan for a file server to service NFS requests and a separate file server to service CIFS requests, because no mechanisms exist to unify the security model. This plan has the advantage of completely avoiding any problems that might arise when attempting to unify the disparate protocols onto a single server. However, it also has the disadvantage of doubling the administrative workload and creating an artificial data barrier for users who access data from both Windows and UNIX clients. Instead of resolving this barrier on the server where it should be resolved, the administrator is left with no other choice than to deploy CIFS client software on many UNIX machines or NFS client software on many Windows

machines. Additional administrative problems can result, especially when unexpected incompatibilities are exposed in the client software.

The best practice is to unify the security models for both NFS and CIFS access so that clients access the same data via the disparate protocols.

Each TaskSmart N-Series appliance ships with a preinstalled User Mapping Service. This service loads UNIX UID, GID, and associated user and group name information from either a **passwd/group** file pair or directly from an NIS server. The service then maps these UIDs and GIDs to appropriate native Windows user and group names, resident locally or on a centralized Windows Domain controller. This mapping capability allows similar file permissions to be expressed to both NFS and CIFS clients, and also allows for systematic planning for heterogeneous environments based on predetermined functionality.

Mapping UNIX Accounts to Windows Accounts

Once a UNIX account database is integrated with the TaskSmart N-Series appliance, either by means of **passwd** and **group** files or by means of direct integration with a NIS Domain, these accounts can be mapped to native Windows accounts. The appliance is based on Windows Powered OS, and the appliance file system is NT File System (NTFS). NTFS stores file permissions as Access Control Lists (ACLs) attached to each file and directory, and these permissions are stored in terms of Windows users and groups. Because these Windows users and groups are different from the UNIX UIDs and GIDs stored in a **passwd/group** or NIS account database, the two sets of accounts must be mapped one to the other so that proper file privileges can be expressed to both CIFS clients and NFS clients.

Mapping two sets of accounts to each other is a problem with NFS servers in general. The problem usually is resolved by unifying disparate UNIX account databases into a single NIS Domain. This strategy alone cannot address the full integration problem of merging a CIFS security paradigm with an NFS security paradigm. Since the TaskSmart N-Series appliance does merge the two paradigms, a user mapping service is required.

This user mapping service ships preinstalled on the TaskSmart N-Series appliance. This service is responsible for bridging the divide between Windows security and UNIX security. It works on three levels. First, it is possible to explicitly map one or more UNIX UIDs to a single Windows user account; and one or more UNIX GIDs to a single Windows group. If an explicit map has not been configured for a particular UNIX UID or GID, then the mapping service looks up the name of the UID or GID in the UNIX account database that has been imported into the appliance either from **passwd/group** files or from NIS. It attempts to implicitly map this name to an identical user or group name in the Windows account database. Such an implicit map must match exactly in spelling (but not case), and implicit mapping in general can be turned off. If an implicit mapping cannot resolve the UID or GID to a native user or group name, then the UID or GID is squashed down to an anonymous user and group. This user or group is set by default to have little or no access privilege on the appliance. The *Compaq TaskSmart N2400 Administration Guide* provides further details on how to set up and manage this mapping service.

The important issue to address when planning for deployment in such a heterogeneous environment is keeping track of user and group names for both the UNIX and Windows account databases. Implicit mapping is the simplest plan. If every UNIX user and group has an identically named corresponding user and group entry in the Windows account database, then account mapping happens automatically. Jdoe, UID 500 in a NIS account database, automatically becomes DOMAIN\JDoe in a Windows Domain controller account database. JDoe's group, Users, GID 100 in a NIS account database, automatically becomes DOMAIN\Users in a Windows Domain controller account database, provided the user and group exist in both account

domains. If there also exists the user, jdoe, in the NIS account database, then it is also implicitly mapped to DOMAIN\JDoe in a Windows Domain controller account database, just as users in NIS would not be implicitly mapped to Users in a Windows Domain controller. The implicit map succeeds as long as the names are identical, regardless of case.

Thus, a best practice in planning for a heterogeneous environment is to implement a systematic user and group naming procedure so that UNIX users and Windows users are named in identical fashions. An unfortunate reality, however, is that heterogeneous environments are rarely planned as such from the beginning. Instead, heterogeneous environments evolve, each with their own set of administrators who may rarely speak to one another. Sometimes multiple NIS domains must be merged, as when two companies or two separate divisions within the same company merge operations. Explicit mapping can help here to resolve differences or overlap (such as a pair of NIS domains, each of which contains a JDoe) in naming conventions. However, explicit mapping should in most cases be used as a migration strategy. Even when merging large operations, new plans should be made for all new users and groups that come into the environment. That way, all of these new users and groups can be implicitly mapped across the two account databases. As older accounts drop off due to attrition and retirement of existing account users, many of the explicit maps drop off with them. It is probably worth the effort to migrate the groups into the new, unified account naming convention. Once the work is done on what may be relatively few groups, it need not be done again.

Once users and groups are properly mapped across the two account realms, it is important to understand the effect on one realm of changing file permissions in the other. As the administration guide documents, there are more than a dozen different granular file privileges that can be explicitly granted or revoked. CIFS is a direct subset of this. However, UNIX has only three: read, write, and execute. The file system permissions are repeated for clarity in Table 2.

Table 2. Windows NTFS File Permissions

Property	Description
Traverse Folder / Execute File	If enabled for a user or group, affected users can open the folder or execute the file.
List Folder / Read Data	If enabled, affected users can list the contents of a folder or read data from a file.
Read Attributes	If enabled, affected users can read the core attributes (for example, read-only, hidden, archive) from a file or folder.
Read Extended Attributes	If enabled, affected users can read the extended attributes (for example, compressed or encrypted) from a file or folder.
Create Files / Write Data	If enabled, affected users can create files within a folder or write data to a file.
Create Folders / Append Data	If enabled, affected users can create folders within a folder or append data to an existing file.
Write Attributes	If enabled, affected users can write core attributes to a file.
Write Extended Attributes	If enabled, affected users can write extended attributes to a file.
Delete Subfolders and Files	If enabled, affected users can delete folders and files within a folder.
Delete	If enabled, affected users can delete the affected file or folder.
Read Permissions	If enabled, affected users can read this list of permissions for all configured users and groups.
Change Permissions	If enabled, affected users can change this list of permissions for all configured users and groups.
Take Ownership	If enabled, affected users can take ownership of the affected files or folders from another user.

In addition to these granular permissions, there is a smaller set of common permissions that apply to files. These are also documented in detail in the administration guide, but the list is repeated for clarity in Table 3.

Table 3. Windows NTFS Aggregate File Permissions

Access level	Description
Full Control	Allows all access-level properties and is therefore a superset of Modify, Read & Execute, Read, and Write
Modify	Includes all Advanced access properties except: Delete Subfolders and Files Change Permissions Take Ownership
List Folder Contents	Includes only the following access properties: Traverse Folder / Execute File List Folder / Read Data Read Attributes Read Extended Attributes Read Permissions
Read & Execute	Includes only the following access properties: Traverse Folder / Execute File List Folder / Read Data Read Attributes Read Extended Attributes Read Permissions
Read	Includes only the following access properties: List Folder / Read Data Read Attributes Read Extended Attributes Read Permissions
Write	Includes only the following access properties: Create Files / Write Data Create Folders / Append Data Write Attributes Write Extended Attributes

CIFS itself can apply three different permissions to a share. These permissions are a direct subset of the file system permissions, and if different file system permissions are applied to a file or sub-folder within a CIFS share, these permissions supercede the share permissions (particularly if they are more restrictive).

The same rules apply to NFS permissions. Notice that the file system has three different aggregate permissions: **Read**, **Write**, and **Read & Execute**. These roughly correspond to the UNIX permissions of **read**, **write**, and **execute**. Note also that the **List Folder Contents** aggregate permission level is identical to the **Read & Execute** level. In fact, when the user mapping service is properly set up, NFS integrates seamlessly into this paradigm as well. If an NFS user applies read, write, and execute privileges to a file for himself or herself, then the

mapped native Windows user receives **Read**, **Write**, and **Read & Execute** privileges. If the same procedure is performed at the group level, then that user's mapped group receives **Read**, **Write**, and **Read & Execute** privileges. If the world is permitted read, write, and execute privileges on the NFS side, then the native Windows **Everybody** group receives similar permissions on the file system. Since the CIFS protocol reflects the file system privileges directly, these same permissions are applied to CIFS users who enter as properly authenticated native Windows users. Table 4 summarizes the permissions map from NFS to NTFS, which in turn provides an expected level of privilege to users accessing data via CIFS:

Table 4. UNIX Permissions Mapped to NTFS Aggregate Permissions

UNIX Permission	NTFS Aggregate Permission
Read	Read
Write	Write
Execute	Read & Execute or List Folder Contents identical aggregate permissions

Table 5 documents the mapping from the UNIX user-group-world paradigm to the corresponding NTFS paradigm:

Table 5. UNIX User-Group-World Mapped to NTFS

UNIX	NTFS
User	Properly mapped user
Group	Properly mapped group
World	Everybody group

That is half the equation. The other half is what to expect on the NTFS side of files and folders created via CIFS or directly on the file system. This can be best understood again in terms of aggregate NTFS permissions. Generally, if appropriate granular permissions are enabled to meet the *minimum* definition of an aggregate permission, then an NFS client sees the aggregate permission expressed in UNIX terms. That is, if at least the **Create Files / Write Data**, **Create Folders / Append Data**, **Write Attributes**, and **Write Extended Attributes** are enabled for a file or directory on the file system (the ones that aggregate to define the write aggregate permission), then an NFS client has write permissions to the file. Of course this write permission only applies to a properly mapped user or group unless this permission is applied to the NTFS **Everybody** group, in which case an NFS sees world write permissions on the file.

The same rule applies to the **Read**, **Read & Execute**, and **List Folder Contents** aggregate permissions. If, for a particular user or group (or for the **Everybody** group), a minimum number of granular permissions are enabled so that all aggregate permission privileges are enabled, then the appropriately mapped UNIX user or group (or the world) has the corresponding UNIX-style access to the file. If more than the minimum number of granular permissions are enabled, that aggregate NTFS permission is also enabled, so **Full Control**, for example, equates with read, write, and execute on the UNIX side. Again, to understand the relationship between UNIX and NFS permissions, see Table 4 and Table 5.

Storage Architectures

There are three core storage implementation technologies:

- Direct Attached Storage (DAS)
- Storage Area Network (SAN)
- Network Attached Storage (NAS)

Compaq takes into account these three implementation technologies with the Compaq Enterprise Storage Architecture, and also with the Compaq Distributed Internet Server Array Architecture. After an explanation of the three core storage implementation technologies, both Compaq architectures are explained.

Direct Attached Storage

Direct Attached Storage (DAS) is used to describe the traditional storage deployment methods where an external storage system is connected directly to one or more servers via a copper SCSI connection. The server(s) access the storage systems as internal server devices, communicating through a device driver and perhaps a file system driver to get at data. More than one can be supported if the external storage system, such as the older Compaq *ProLiant*™ Storage System U2, has a multi-channel I/O bus, which supports multiple connections. Since some applications, such as an Oracle database, can access raw device directly, a file system driver may or may not be necessary.

DAS has the distinct advantage of being low-cost and fairly easy to deploy when only one application needs to access that data. It is typical of traditional NT or Netware file servers and small database deployments. DAS, however, can become an administrative hassle over time because of the general smallness of each deployment. In many cases, these servers and attached storage systems multiply over time to the point where an initial deployment of just a few servers can become hundreds. This alone can create a management problem the costs of which quickly overwhelm the initial low costs of each individual DAS deployment.

Storage Area Network

A Storage Area Network (SAN) is a storage-centric deployment of servers and attached storage into a network, usually Fibre-based, through which the many servers can access individual chunks of storage. A SAN has a strong advantage over DAS because the implementation is network-centric rather than server-centric, and the network is one specifically designed and implemented for storage. A Fibre-based storage network can generally provide greater effective throughput than a Fibre-based Ethernet, for example, because a SAN presents its data to attached servers as a SCSI device rather than as a LAN-aware application. Since a device level protocol like SCSI has much lower overhead than an application-level protocol such as TCP/IP over Ethernet, the net throughput of a storage area network is much higher than that of a gigabit LAN. This means that throughput-intensive applications such as database, Exchange, or large ERP deployments can benefit greatly from the high performance characteristics of a SAN. A SAN provides far greater flexibility than does a DAS, because SAN software tools support dynamic disk carving and theoretically unlimited storage growth over time.

A SAN also allows for data center administrators to take a storage-centric approach to planning. Such an approach can provide a certain level of independence from application eccentricities

because any application server with knowledge of the stored data format can be substituted for, say, an existing application server whose software has suddenly broken down.

A SAN also encompasses both data and disk storage, meaning that both types of storage can be connected to a SAN as just another storage device. And both tape and disk can be partitioned across multiple application servers connected to the SAN to provide adequate storage capacity and throughput for the application. Just like disk, tape capacity can be easily grown over time.

The fundamental disadvantage of a SAN is that the storage generally must be partitioned among the various application servers connected to the SAN. Since a SAN presents its data as a local device, the application connected to that local device takes exclusive control over it. Thus, only one server at a time can gain access to the data. That server can, in turn, expose that data to other clients over a general-purpose network, such as a LAN, at the application level. This is what is done, for example, by a database server that connects to storage via SCSI over a storage network and to database clients via ODBC over an Ethernet LAN. The SAN provides the base storage infrastructure and raw data, and the application manages the complex issue of simultaneous access to data by multiple clients through an application-level protocol. Thus, this fundamental disadvantage of a SAN really is not such a disadvantage. Rather it reflects the fact that the SAN is data-centric and application agnostic, leaving the application issues to the application itself.

Network Attached Storage

Network Attached Storage (NAS) is a specific application of DAS or SAN where the application is sharing files over a general-purpose network, such as a LAN. In the DAS sense, the current implementation of TaskSmart N-Series appliances, NAS consists of a server control unit that houses application software and SCSI-attached storage that houses data. The server control unit manages application-level issues such as snapshots, CIFS shares, NFS exports, and storage device management. The SCSI-attached storage houses data in a format that the server control unit can understand. NAS then exposes this data over a general-purpose network to multiple heterogeneous clients. The NAS application software is responsible for managing issues such as file locking across disparate protocols. It specifically is not responsible for managing the format of such data, other than at the low level so that it can read and write it on its file system. Specific application formats, such as Microsoft DOC, StarOffice SDW, or Windows and UNIX text file formats, are the exclusive domain of applications that use NAS for a storage repository.

In the SAN sense, NAS is one of the applications that hangs off the SAN, just as database, Exchange or ERP might hang off the SAN. This sort of NAS implementation consists of a server control unit that houses application software and Fibre-attached storage that houses data. Since it is Fibre-attached, the storage can be part of an existing SAN or the basis upon which to build a larger SAN. The server control unit manages application-level issues such as snapshots, CIFS shares, NFS exports, and storage device management. The Fibre-attached storage houses data in a format that the server control unit can understand. NAS then exposes this data over a general-purpose network to multiple heterogeneous clients. However, such a Fibre-based implementation allows for a storage-centric planning model.

Compaq Enterprise Storage Architecture

The Compaq Enterprise Storage Architecture (ENSA) takes into account the three basic storage models: DAS for low-cost small deployments; SAN for highly scalable, fault-tolerant, storage-centric deployments; and NAS to unify the two at the application level for presentation of data to multiple heterogeneous clients simultaneously. All three offerings are available from and supported exclusively by Compaq, and all three have been specifically integrated into the single,

overarching ENSA architecture. This allows customers to start with inexpensive DAS and DAS-based NAS implementations that can grow over time into a SAN, with SAN-based NAS, without giving up existing storage investments. It also allows for large customers to immediately deploy a SAN, and SAN-based NAS, when available, knowing that smaller remote branch offices can use DAS and DAS-based NAS in such a way that all parts integrate.

Compaq Distributed Internet Server Array Architecture

The Compaq Distributed Internet Server Array (DISA) architecture is a loosely coupled, multi-tier, highly scalable, highly available application deployment architecture commonly used in Compaq *NonStop*™ eBusiness deployments. Generally, it consists of a load-balancing layer, which load balances incoming application requests across multiple application servers in an application layer, which in turn connects to a highly available data resources layer for centralized access to database tables and files. DISA itself is well documented elsewhere. Refer to the eBusiness Infrastructure and Internet and E-Commerce solutions areas of Compaq ActiveAnswers at

www.compaq.com/ActiveAnswers

It is important to note that one of the core pieces of DISA infrastructure is a data resources layer that provides access to files. NAS is a very good solution for providing such access because it has been specifically tuned to provide greater performance for file serving than a typical general-purpose server.

Figure 5 documents how NAS might be deployed in a typical DISA environment.

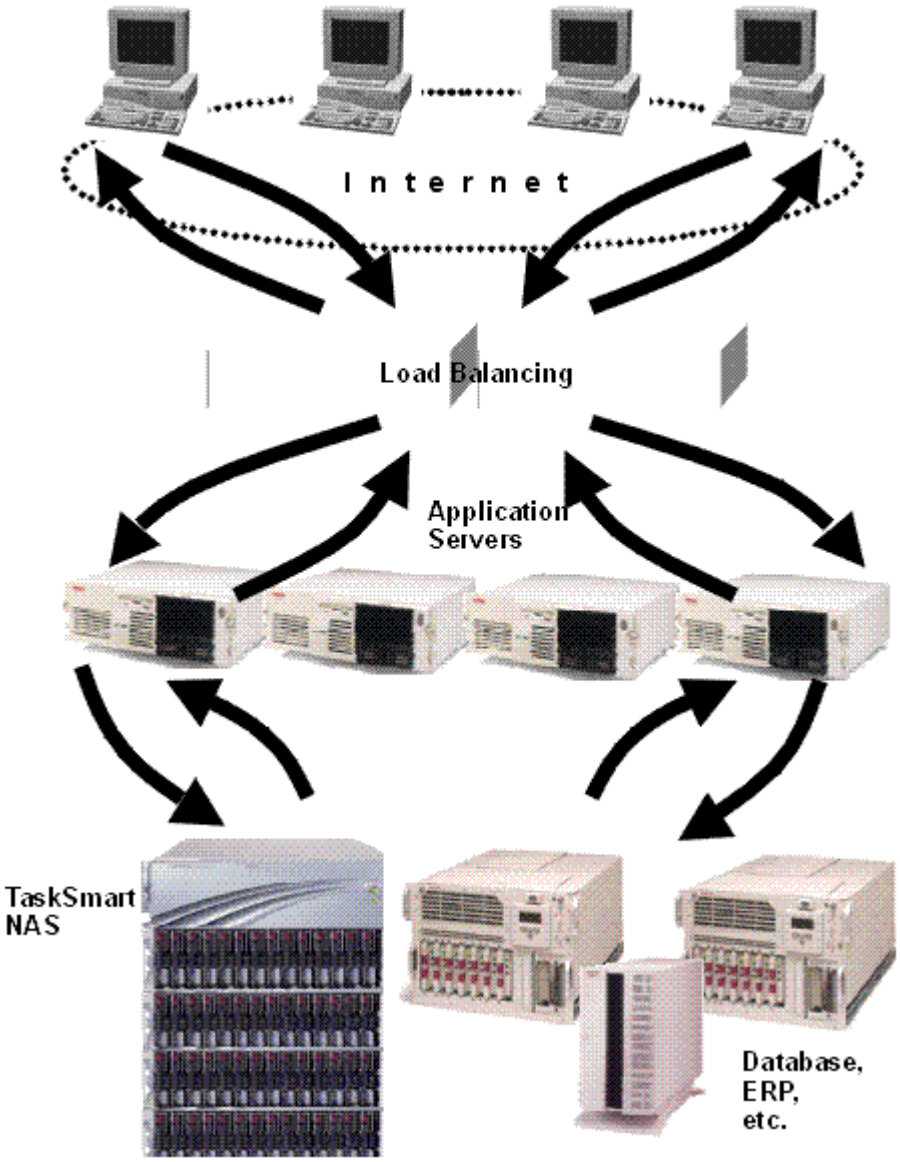

Figure 5. NAS Deployment in a DISA Architecture

In order to prepare for deployment, however, it is important to understand the performance implications of such deployment. To understand these implications, it is important to plan for deployment based on known file throughput capabilities of the TaskSmart N-Series appliance.

Integration of NAS into an Existing Infrastructure: Onsite and Offsite Planning

Integration of NAS into an existing infrastructure builds on the concepts already documented here. A common integration is the migration of several existing file servers onto a single TaskSmart N-Series appliance. A part of this is documented previously in the “Storage Sizing” section. “Storage Sizing,” however, only delves into the necessary disk size of the TaskSmart N-Series appliance. Just as fundamental is a look at the best practices surrounding maximizing performance and availability of the TaskSmart N-Series appliance, as detailed earlier in this paper.

The TaskSmart N-Series appliance ships with a configuration utility that allows for writing core network settings to a diskette. This diskette is then inserted into the diskette drive of the appliance, the appliance is powered up, and the configuration settings are automatically applied. This allows for administrators to control the setup of a TaskSmart N-Series appliance that is deployed to a remote branch office or to any location outside the immediate physical control of administrators. Once the appliance is properly configured on the network, administrators can access it remotely through the included Remote Insight Lights-Out Edition board remote console or through Telnet to apply the remaining configuration settings.

TaskSmart Configuration Utility specifically allows administrators to preconfigure most network settings for all five application network interfaces, and a subset of these settings for the Remote Insight Lights-Out Edition board network interface, including the following:

- DHCP settings
- IP address
- Netmask
- Gateway(s)
- DNS
- WINS
- NIC speed
- NIC duplex
- SNMP

TaskSmart Configuration Utility also allows for configuration of a default remote console administrator so that authenticated access to the remote console is immediately available. The utility then writes these settings to a diskette, which is inserted into the TaskSmart N-Series appliance when turned on for initial setup. Thus, as long as a remote network is properly cabled, a remote administrator has ready access to the TaskSmart N-Series appliance for further configuration.

Note: NIC teaming for network load balancing and network fault tolerance is not currently supported by way of the TaskSmart Configuration Utility. It is, however, possible to appropriately cable the TaskSmart N-Series appliance so that it works in both a NIC teaming environment and a simple network environment. Thus, the TaskSmart Configuration Utility is used to set up simple networking so that the administrator can gain remote access to the appliance, and once the administrator accesses the appliance, the standard Compaq NIC Teaming Utility is available to set up network load-balancing or fault tolerance. Further information can be obtained in the *Compaq TaskSmart N2400 Administration Guide* or by selecting the TaskSmart administration guide from the console help list.

Onsite deployments are a bit more straightforward because the appliance administrator always has the option of going directly to the physical TaskSmart N-Series appliance console to complete all administration tasks. During initial setup, the appliance looks for an offline configuration diskette. If it finds one, those settings are applied. If not, a default set of settings is applied instead. Either way, the appliance comes up and can be configured, but if default settings are used, there is no guarantee that they mesh with the rest of the appliance administrator's infrastructure. In the onsite scenario, this may not be as important, because physical console access is available. Offsite deployments are much simpler if the TaskSmart Configuration Utility is used.

For the integration of the TaskSmart N-Series appliance into a Novell environment, determine whether the network infrastructure is a mixed Novell and Microsoft environment or an exclusively Novell server environment, with no Microsoft Domain controller. See the section, "Homogeneous and Heterogeneous NCP Environments," for more information.

Compaq Professional Service Offerings

Compaq Services offers comprehensive global service and support, including complete services for the implementation, management, and support of customers environments:

- Warranty upgrades
- Installation and configuration
- Priority Service Plan
- Performance and Capacity Planning
- System Management and Monitoring

For more information, visit the Compaq website at

www.compaq.com/support

Appendix – Test Methodology

For the results shown in Figure 1 and Figure 2, tests were set up to run NetBench as the primary benchmarking tool. Four networks were independently configured with 12 clients each. In one test scenario, these clients ran Windows NT 4.0 Workstation with Service Pack 4. In another test scenario, the same clients ran Windows 2000 Professional. Each client was a Compaq *DeskPro™* EN Small Form Factor desktop configured with either a 400-MHz Intel Pentium II processor or a 500-MHz Pentium III processor. Each client was also configured with either 64 MB or 128 MB of RAM.

The NetBench controller was also configured as a DeskPro EN Small Form Factor machine deployed with a 500-MHz Pentium III processor, 256 MB of RAM, and 4 independent network interfaces to connect to the four test networks. This machine also served as a DHCP server to dynamically assign addresses to each network interface in the test and as a Windows NT 4.0 Primary Domain controller.

The target server was a TaskSmart N-Series appliance configured with fourteen data drives. These data drives were carved up into a single RAID 5 set, with a single online spare. A single logical drive was created from the RAID 5 set. This logical drive was then imported into a single SWVR pool, and a single virtual disk that consumed 70% of available pool space was extracted and presented to the operating system. The target server was configured with the standard Remote Insight Lights-Out Edition board, a quad-port Ethernet NIC, and a Smart Array 4200 Controller. It had the standard 1 GB of RAM deployed as well as the standard number of two 733-MHz Intel Pentium III processors, each with 256 KB of onboard cache.

Three modifications were made from a typical NetBench enterprise disk mix test:

1. The enterprise disk mix was reduced from 60 clients peak to 48 clients peak to accommodate the available hardware.
2. The run-up from 1 to 48 clients was iterated 10 times inside each mix, and a representative run from the series of 10 runs was used to report performance. One exception to this was in the Windows 2000 testing. To save time, the 4-KB and 16-KB test runs iterated through the enterprise disk mix just two times, and the second run was used to report performance.
3. Rather than map each network drive as the same user on each client machine, each client machine used its own user name to map the network drive. The net effect of this was to create a test in which a greater number of Windows user ACLs were on the network. As such, the test was a bit more realistic than a typical NetBench test.