ARX CLI Maintenance Guide

810-0045-00

Publication Date

This manual was published on May 13, 2013.

Legal Notices

Copyright

F5 Networks, Inc. (F5) believes the information it furnishes to be accurate and reliable. However, F5 assumes no responsibility for the use of this information, nor any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent, copyright, or other intellectual property right of F5 except as specifically described by applicable user licenses. F5 reserves the right to change specifications at any time without notice.

Trademarks

Access Policy Manager, Advanced Client Authentication, Advanced Routing, APM, Application Security Manager, ARX, AskF5, ASM, BIG-IP, BIG-IQ, Cloud Extender, CloudFucious, Cloud Manager, Clustered Multiprocessing, CMP, COHESION, Data Manager, DevCentral, DevCentral [DESIGN], DNS Express, DSC, DSI, Edge Client, Edge Gateway, Edge Portal, ELEVATE, EM, Enterprise Manager, ENGAGE, F5, F5 [DESIGN], F5 Management Pack, F5 Networks, F5 World, Fast Application Proxy, Fast Cache, FirePass, Global Traffic Manager, GTM, GUARDIAN, IBR, Intelligent Browser Referencing, Intelligent Compression, IPv6 Gateway, iApps, iControl, iHealth, iQuery, iRules, iRules OnDemand, iSession, L7 Rate Shaping, LC, Link Controller, Local Traffic Manager, LTM, Message Security Manager, MSM, OneConnect, OpenBloX, OpenBloX [DESIGN], Packet Velocity, Policy Enforcement Manager, PEM, Protocol Security Manager, PSM, Real Traffic Policy Builder, Rosetta Diameter Gateway, ScaleN, Signaling Delivery Controller, SDC, SSL Acceleration, StrongBox, SuperVIP, SYN Check, TCP Express, TDR, TMOS, Traffic Management Operating System, Traffix Diameter Load Balancer, Traffix Systems, Traffix Systems (DESIGN), Transparent Data Reduction, UNITY, VAULT, VIPRION, vCMP, virtual Clustered Multiprocessing, WA, WAN Optimization Manager, WebAccelerator, WOM, and ZoneRunner, are trademarks or service marks of F5 Networks, Inc., in the U.S. and other countries, and may not be used without F5's express written consent.

All other product and company names herein may be trademarks of their respective owners.

Patents

This product may be protected by U.S. Patents 7,877,511; 7,958,347. This list is believed to be current as of May 13, 2013.

Export Regulation Notice

This product may include cryptographic software. Under the Export Administration Act, the United States government may consider it a criminal offense to export this product from the United States.

RF Interference Warning

This is a Class A product. In a domestic environment this product may cause radio interference, in which case the user may be required to take adequate measures.

FCC Compliance

This equipment has been tested and found to comply with the limits for a Class A digital device pursuant to Part 15 of FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This unit generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user, at his own expense, will be required to take whatever measures may be required to correct the interference.

Any modifications to this device, unless expressly approved by the manufacturer, can void the user's authority to operate this equipment under part 15 of the FCC rules.

Canadian Regulatory Compliance

This Class A digital apparatus complies with Canadian ICES-003.

Standards Compliance

This product conforms to the IEC, European Union, ANSI/UL and Canadian CSA standards applicable to Information Technology products at the time of manufacture.

Acknowledgments

This product includes software from several third-party vendors. Each vendor is listed below with applicable copyright.

Revision History

September 2006 - Rev A

October 2006 - Rev B, updates for Software Release 2.4.2

March 2007 - Rev C, updates for Software Release 2.5.0

May 2007 - Rev D, updates for Software Release 2.5.1

November 2007 - Rev E, new Security-Table Appendix for Software Release 2.7.0

December 2007 - Rev F, updates for Software Release 3.0.0

March 2008 - Rev G, updates for Software Release 3.1.0; convert to F5 format

June 2008 - Rev H, updates for Software Release 4.0.0

October 2008 - Rev J, re-brand the OS

June 2009 - Rev K, updates for Software Release 5.00.000

July 2009 - Rev L, minor updates for Software Release 5.00.001

July 2009 - Rev M, re-orient policy discussion toward latest best-practices, for Software Release 5.00.005

November 2009 - Rev N, updates for Software Release 5.01.000

March 2010 - Rev P, updates for Software Release 5.01.005

August 2010 - Rev Q, updates for Software Release 5.01.007

November 2010 - Rev R, updates for Software Release 5.02.000

June 2011 - Rev S, updates for Software Release 6.00.000

September 2011 - Rev T, updates for Software Release 6.01.000

January 2012 - Rev U, minor updates for 6.01.001.14111 hot fix

July 2012 - Rev V, updates for Software Release 6.02.000

October 2012 - Rev W, updates for Software Release 6.03.000
June 2013 - Rev X, updates for Software Release 6.04.000

Table of Contents

1
Introduction

2
Configuring Volume Snapshots

8
Troubleshooting Tools

9
Troubleshooting Network Connections

10
Troubleshooting Managed Volumes

11
Troubleshooting CIFS Services

1

Introduction

Overview

This manual contains instructions and best practices for troubleshooting and maintaining the Adaptive Resource Switch (ARX®) after it is fully connected to the network and running storage services. These instructions focus on the Command-Line Interface (CLI).

Use this book after the ARX is installed, connected to the IP network, and serving clients. The platform's *Hardware Installation* manual explains how to install the ARX. You can set up basic networking through a GUI wizard, or work with the more-advanced CLI features described in *ARX® CLI Network-Management Guide*. The GUI provides wizards for configuring most of the storage features in the ARX; the *ARX® CLI Storage-Management Guide* provides procedures and best-practices for configuring all of them through the CLI.

The ARX

The Adaptive Resource Switch (ARX®) is a highly available and scalable solution that brings resource awareness to a file storage infrastructure, and adapts these resources to meet the demands of users and applications in real time. The ARX provides a file-virtualization layer that aggregates the total capacity and performance of your file storage. A *namespace* provides location-independent, transparent mapping of user requests onto the appropriate storage resource. You can configure policies that the switch enforces for the placement, replication and migration of files. Through policy configuration, the ARX adapts to the real-time demands of users and applications. The ARX thereby serves as a *resource proxy* for the files and services behind it.

Back-end Storage and Servers

The Adaptive Resource Switch aggregates heterogeneous file systems and storage into a unified pool of file storage resources. Through this unification, you can manage these resources to adapt to user demands and client applications. File storage assets can be differentiated based on user-defined attributes, enabling a class-of-storage model. You can reclaim stranded capacity through policy implementation for more effective storage utilization, and you can add capacity without disruption. Back-end resources are monitored for availability and performance, as well as user-access patterns that drive policy decisions.

Front-end Services

The Adaptive Resource Switch acts as an in-band file proxy for the Network File System (NFS) and Microsoft's Common Internet File System (CIFS) protocols.
Front-end services provide the file virtualization layer that masks the physical file storage from the user and application. The switch becomes the file access point, as opposed to the actual physical resource, providing file access through a namespace. Users and applications maintain a single consistent file path that is transparently mapped to the proper physical resource where the information resides.

Policy

The Adaptive Resource Switch provides policy-based resource switching. Through *policy* configuration, you can optimize the placement of files onto the appropriate storage resources and automatically adapt these resources based on user and application demand. The ARX performs file replication and migration based on performance, usage or other life-cycle characteristics, enabling you to implement a flexible file services strategy. Examples of policies include: migrating files to reclaim stranded capacity; migrating files across different tiers of storage based on access patterns and/or value; and replicating frequently accessed files for performance. The result is more efficient utilization and greater flexibility in file storage management.

Resilient Overlay Network (RON)

You can connect multiple ARXes with a Resilient Overlay Network (RON), which can reside on top of any IP network. This provides a network for distributing and accessing file storage. ARXes can replicate storage to other switches in the same RON, updating the replicas periodically as the writable master files change. This is called a *shadow copy*, where a source volume on one switch periodically copies its files to one or more *shadow volumes* on other switches. Clients can access the shadow volumes at multiple geographic locations, independent to where the source volume resides.

Audience for this Manual

This manual is intended for

- network technicians responsible for layer 1 and 2 networks,
- network engineers responsible for the Internet Protocol (IP) layer (layer 3),

- storage engineers who design and manage storage systems (SANs, NASes, and DASes), and
- crypto officers who manage all of the Critical Security Parameters (CSPs) of a network.

The text presumes that all readers are comfortable with a command-line interface (CLI), especially one based on the Cisco IOS.

Using this Manual

This manual contains instructions for maintaining an ARX after it has been fully installed, it has been fully provisioned, and it has provided service to your clients. Other manuals explain installation and provisioning. The *Hardware Installation Guide* contains instructions to install the switch, set up its management IP, and prepare it for CLI provisioning. After a successful installation, a network engineer can then use the ARX Manager or the ARX® CLI Network-Management Guide to set up all networking parameters. Finally, a storage engineer can use the GUI wizards or the *ARX® CLI Storage-Management Guide* to aggregate storage and set up storage policies.

The chapters in this manual contain maintenance instructions, such as

1. backing up and restoring client files through the ARX,
2. backing up configuration data with show running-config and similar commands,
3. performing software upgrades,
4. maintaining the metadata used by managed volumes,
5. reading the syslog and working with other basic-maintenance facilities,
6. using some network-troubleshooting tools (such as ping),
7. taking down and repairing managed volumes, and
8. powering down the switch for a planned outage.

Document Conventions

This manual uses the following conventions:

`this font` represents screen input and output;

- **`bold text`** represents input, and
- *`italic text`* appears for variable input or output.

this font is used for command-syntax definitions, which use the same rules for bold and italic.

Command-syntax definitions also use the following symbols:

- [*optional-argument*] - square brackets ([ ]) surround optional arguments;
- *choice1* | *choice2* - the vertical bar (|) separates argument choices;
- {*choice1* | *choice2* | *choice3*} - curly braces ({ }) surround a required choice;
- [*choice1* | *choice2*]* - an asterisk (*) means that you can choose none of them, or as many as desired (for example, "choice1 choice2" chooses both);
- {*choice1* | *choice2*}+ - a plus sign (+) means that you must choose one or more.

CLI Overview

The Command-Line Interface (CLI) has its commands grouped into modes.

Modes are structured as a tree with a single root, *exec* mode. This section summarizes the mode structure and explains some CLI conventions.

Exec Mode

When you log into the CLI, you begin in exec mode. If the hostname is "bstnA," the prompt appears as shown below:

```
bstnA>
```

You can access all global commands (such as show commands) from exec mode, and you can use the enable command to enter priv-exec mode.

```
bstnA> enable
```

Global Commands

You can access global commands from any mode, not just exec. Global commands include all show commands and terminal-control commands.

Priv-exec Mode

Priv-exec mode has the following prompt:

```
bstnA#
```

Priv-exec mode contains chassis-management commands, clock commands, and other commands that require privileges but do not change the network or storage configuration.

Priv-exec has two sub modes, cfg and gbl.

Cfg Mode

To enter cfg mode, use the config command:

```
bstnA# config
bstnA(cfg)#
```

Config mode contains all modes and commands for changing the configuration of the local switch, such as network configuration.

Gbl Mode

To enter gbl mode, use the global command:

```
bstnA# global
bstnA(gbl)#
```

Gbl mode controls all parameters that are shared in a redundant pair, such as namespaces and global servers.

Exiting a Mode

From any mode, use the exit command to return to its parent mode. From priv-exec mode, this command exits the CLI; to go from priv-exec mode back to exec mode, use the no enable command.

From any submode of cfg or gbl mode, you can return immediately to priv-exec mode by using the end command or pressing <Ctrl-z>.

Prompts

Prompts contain information about your position in the mode hierarchy as well as the name of the object you are configuring. For example, suppose you use the following command in gbl mode:

```
bstnA(gbl)# namespace wwmed
bstnA(gbl-ns[wwmed])#
```

This command places you into a new mode, as indicated by the new CLI prompt. The prompt shows the name of the mode, "gbl-ns," and the name of the configuration object, a namespace called "wwmed." Abbreviations are used for mode names (for example, "ns" instead of "namespace") to conserve space on the command line.

When you descend to lower modes in the config tree, the prompt offers more information. To extend the previous example, suppose you enter the following command to configure the "/local" volume in the wwmed namespace:

```
bstnA(gbl-ns[wwmed])# volume /local
bstnA(gbl-ns-vol[wwmed~/local])#
```

The tilde character (~) separates a parent object from its child: "wwmed~/local" shows that you are in the "/local" volume under the "wwmed" namespace.

The no Convention

Most config commands have the option to use the "no" keyword to negate the command. For commands that create an object, the no form removes the object. For commands that change a default setting, the no form reverts back to the default. As an example,

```
bstnA(gbl-ns[wwmed])# no volume /local
```

removes the "/local" volume from the "wwmed" namespace.

The enable/no enable Convention

Many objects and configurations require you to enable them using the enable command before they can take effect. Likewise, many objects and configurations require you to first disable them using the no enable command before you can complete a related command or function. The no

enable command does not remove an object; it only disables it until you re-enable it. The enable/no enable commands exist in many modes and submodes in the CLI.

For example, the following command sequence enables the namespace named “wwmed:”

```
bstnA(gbl)# namespace wwmed
bstnA(gbl-ns[wwmed])# enable
bstnA(gbl-ns[wwmed])# ...
```

Getting Started

For the initial login, refer to the instructions for booting and configuring the switch in the appropriate *Hardware Installation Guide*.

For subsequent logins, use the following steps to log into the F5 CLI:

1. If you are on-site, you can connect a serial line to the serial console port. This port is labeled ‘Console’ or ‘10101’ (depending on your ARX platform). By default, the port is set for 9600 baud, 8, N, 1.

   You can also telnet to the switch’s management interface. For example:

   ```
   telnet 10.10.10.10
   ```

   In either case, a login prompt appears:

```
Username:
```

2. Enter your username and password. For example:

```
Username: admin
Password: acopia
```

   The CLI prompt appears:

```
SWITCH>
```

   The name, “SWITCH,” is the default hostname. The hostname is reset as part of the initial-boot process, so it is likely that yours will differ.

Entering Cfg or Gbl Mode

The CLI contains two major configuration modes: cfg and gbl. The cfg mode contains submodes for configuring locally-scoped parameters, only applicable to the local ARX. These parameters include layer-2, layer-3, and chassis configuration. Gbl mode applies to configuration that is shared among both switches in a redundant pair, such as namespaces and global servers.

After you log into the CLI, use the config command to enter cfg mode:

```
SWITCH> enable
SWITCH# configure
SWITCH(cfg)#
```

To enter gbl mode, use the global command instead:

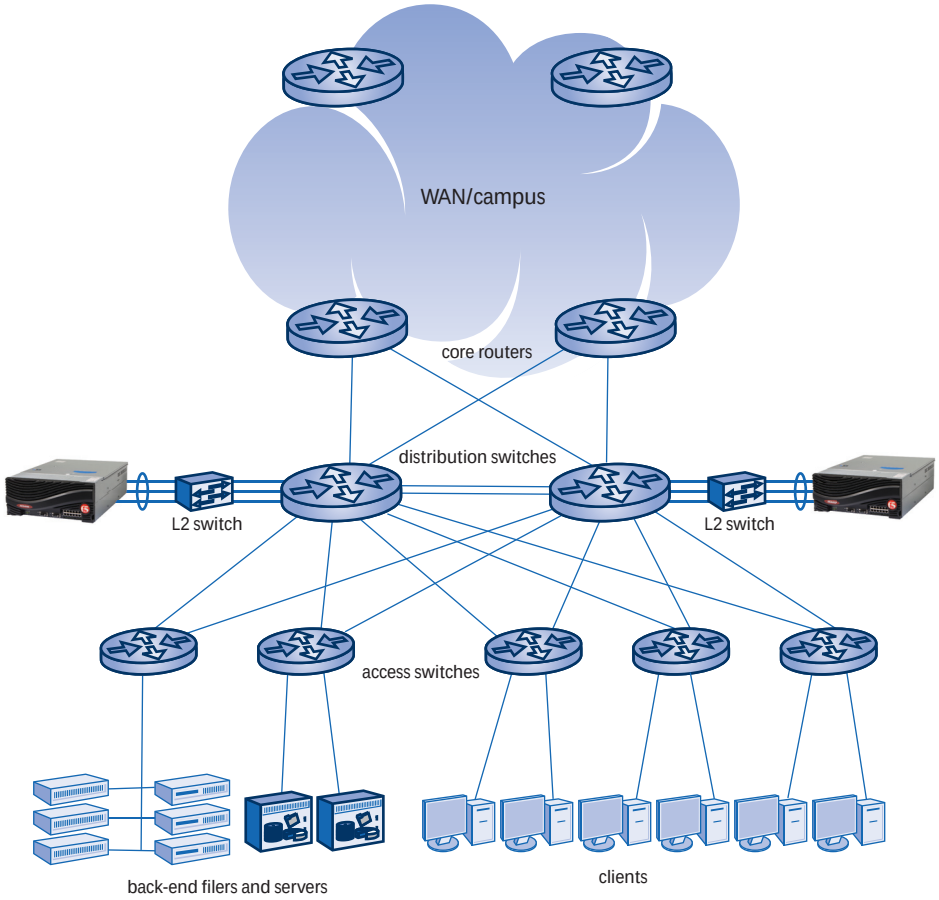
```
SWITCH> enable
SWITCH# global
SWITCH(gbl)#
```

The command sequences in this manual all begin either in cfg mode or gbl mode.

Sample Network

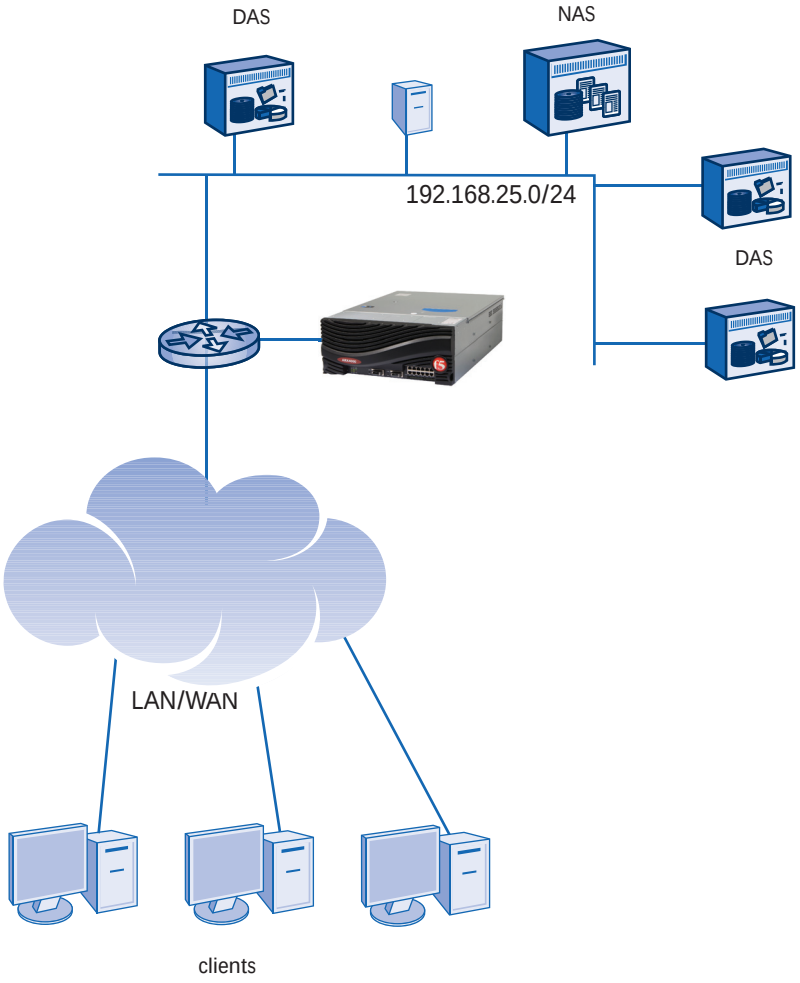
The examples in this manual draw from a single, fictitious network. This section shows the topology of the network and the placement of the ARX.

The environment is assumed to be a three-tiered network in a large data center. The first tier is core routers that provide connectivity to a campus or WAN. The second tier has redundant distribution switches that distribute all data-center traffic between the access switches; the access switches constitute the third tier of the network. All LAN clients, back-end servers, and back-end filers connect to the access switches, which are layer-2 bridges. Through the access switches and distribution switches, LAN clients connect to all back-end servers/storage, to one another, and to the WAN.



The sample network has redundant ARXes connected at each of the distribution switches. Physically, this is a one-armed connection; conceptually, the ARX has front-end clients in front of it and back-end servers and filers behind it. To simplify the examples, the access switches are removed from the remaining illustrations in the book.

The network filers all live on a class-C subnet at 192.168.25.x. These filers are called *back-end* filers, since they are the storage behind the *front-end* services of the ARX. The filers can be heterogeneous: NAS devices and file servers (possibly with additional DAS) need only support CIFS or NFS to be on the back end of the ARX.



Contacting Customer Service

You can use the following methods to contact F5 Networks Customer Service:

F5 Networks Online Knowledge Base Online repository of answers to frequently-asked questions.	http://support.f5.com
F5 Networks Services Support Online Online customer support request system	https://websupport.f5.com
Telephone	Follow this link for a list of Support numbers: http://www.f5.com/support/support-services/contact/

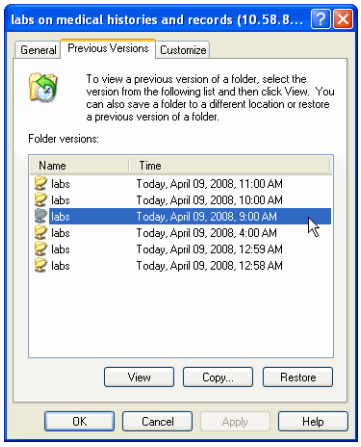
2

Configuring Volume Snapshots

Overview

An ARX volume can support regular *snapshots*, or copies from a particular point in time, of its client files and directories.

Most Microsoft Windows clients can view these snapshots with Windows Explorer: they can click Properties on any file or directory and select the Previous Versions tab. The snapshots appear as instances of the same file or directory with progressively-older time stamps. This is called the Volume Shadow-copy Service, or *VSS*, for shared folders:



A Windows client can recover an older version of any file or subdirectory through the VSS interface.

NFS clients can access snapshots via the ".snapshot" directory in every directory in a managed volume. This ".snapshot" directory is populated with a subdirectory for each snapshot that is present, with each snapshot being of the tree at that location in the managed volume.

The ARX volume coordinates snapshots on multiple back-end filers by issuing snapshot commands to each of the filers. At least one of the filers behind the volume must be properly configured to accept the volume's snapshot commands, as described below.

The ARX presents only snapshots that it created or that were imported using the manage snapshot command.

◆ Note

If you add snapshot support to a volume that already has active CIFS clients, those CIFS clients must close all instances of Windows Explorer for the Previous Versions *tab to appear. Windows Explorer only checks its network shares for snapshot support during startup.*

Before You Begin

The ARX volume must be backed by NetApp filers, Windows servers, Data Domain systems, and/or EMC Celerra servers that support snapshots or checkpoints. The Release Notes show the specific releases supported by ARX snapshots. The volume can also be backed by filers that do not support snapshots, though shares from these filers are excluded from the coordinated-ARX snapshot.

Given at least one back-end filer running a supported release, use the instructions in this section to prepare the filer(s) for snapshot support.

NetApp Configuration

To support ARX snapshots, each NetApp volume must have the nosnapdir option turned off. From the NetApp CLI, use the vol options command on each NetApp volume behind the ARX:

```
vol options vol-name nosnapdir off
```

where ***vol-name*** identifies the NetApp volume.

For example, the following commands access a NetApp filer named "nas10" through SSH, and then permit snapshots in two volumes:

```
juser@mgmt17:~$ ssh root@192.168.25.49
root@192.168.25.49's password: password

nas10*> vol options datavol1 nosnapdir off
nas10*> vol options datavol2 nosnapdir off
nas10*> ...
```

WinRM on Windows

The ARX uses the Windows Remote Management (WinRM) service to manage snapshots on a Windows server. WinRM must be installed on any Windows server behind a snapshot-supporting volume, and a WinRM listener must be listening on HTTP port 80.

For a Windows 2003 R2 server, you must add the WinRM service through the Add/Remove Programs interface, and it installs with an incompatible default. Configure WinRM on the server to allow for HTTP unencrypted negotiation. Encrypted HTTP is not supported on this version of the Windows server. To ensure a high level of security, the ARX uses Kerberos authentication over HTTP.

To support this Kerberos authentication, the ARX must understand the Windows-domain hierarchy in the local Active-Directory (AD) forest. See *Discovering the Active-Directory Forest (Kerberos), on page 3-9* of the *ARX® CLI Storage-Management Guide*.

EMC Celerra Best Practices

We recommend that your ARX volumes contain no more than two EMC volumes (called “file systems” in EMC documentation) behind any one EMC management station. EMC-Celerra checkpoints typically take 5-10 seconds when the Celerra is under moderate load, where NetApp snapshots rarely take longer than 5 seconds. Further, the EMC management station serializes its checkpoints; if a given management station needs a checkpoint from file-server A and file-server B at the same time, the file-server-A checkpoint must finish before the file-server B checkpoint can begin.

EMC Data Domain Best Practices

EMC Data Domain servers provide a unique presentation to NFS clients; they are only accessible from volume root’s .snapshot directory. The .snapshot directory under each lower directory does not include the Data Domain snapshots.

This limitation does not apply to CIFS clients, which can access the Data-Domain snapshots in every .snapshot directory.

You can avoid this NFS limitation if every directory on the Data Domain server(s) is striped to another filer type. To ensure universal striping, migrate all directories in the volume to a share from a different vendor and promote them to “master:”

- create a *filename-fileset* that recursively matches everything in the volume (using *path* exact / and *recurse*),
- create a *place-rule*, and
- use the *from (gbl-ns-vol-plc)* command to have that fileset match directories and promote them.

In a tiered volume, you can use this place rule to migrate all master directories to a Tier 1 share. This is a best practice for Tier 1 volumes.

Creating a Proxy User for Accessing the Filer’s CLI

To invoke snapshots at the back-end filer, the ARX volume requires administrative privileges at the filer’s CLI. You add the proper administrative username and password as a proxy user, which typically holds Windows-client credentials.

For EMC Celerra, EMC Data Domain, and NetApp equipment, these are UNIX credentials that do not require a Windows domain; they are for SSH or RSH logins only.

For Windows servers, these credentials require a Windows domain with a full FQDN, so that they can authenticate with Kerberos.

For example, this command sequence adds a proxy user named “nas_admin:”

```
bstnA(gbl)# proxy-user nas_admin
```

```
bstnA(gbl-proxy-user[nas_admin])# user root
Password: rootpasswd
Validate Password: rootpasswd
bstnA(gbl-proxy-user[nas_admin])# exit
bstnA(gbl)# ...
```

As another example, this command sequence adds a proxy user named "cifs_admin" for logging into Windows servers:

```
bstnA(gbl)# proxy-user cifs_admin
bstnA(gbl-proxy-user[cifs_admin])# user Administrator
Password: adminpasswd
Validate Password: adminpasswd
bstnA(gbl-proxy-user[cifs_admin])# windows-domain MEDARCH.ORG
bstnA(gbl-proxy-user[cifs_admin])# exit
bstnA(gbl)# ...
```

For details on these proxy-user commands and others, see *Adding a Proxy User, on page 3-4* of the *ARX® CLI Storage-Management Guide*.

Preparing an External-Filer Configuration

The ARX requires parameters for using the CLI or API at each snapshot-supporting filer. This includes the type of filer (a vendor name) and proper login credentials for the filer's CLI or API. The commands for setting these parameters are described in *Preparing the Filer for ARX-Snapshot Support, on page 6-12* of the *ARX® CLI Storage-Management Guide*.

For example, the following command sequence sets up snapshot access for the "nas10" filer:

```
bstnA(gbl)# external-filer nas10
bstnA(gbl-filer[nas10])# filer-type network-appliance management-protocol rsh
bstnA(gbl-filer[nas10])# proxy-user nas_admin
bstnA(gbl-filer[nas10])# manage snapshots
bstnA(gbl-filer[nas10])# exit
bstnA(gbl)# ...
```

As another example, this command sequence sets up snapshot access for the "fs2" filer, a Windows server:

```
bstnA(gbl)# external-filer fs2
bstnA(gbl-filer[fs2])# filer-type windows
bstnA(gbl-filer[fs2])# proxy-user cifs_admin
bstnA(gbl-filer[fs2])# manage snapshots
bstnA(gbl-filer[fs2])# exit
bstnA(gbl)# ...
```

Creating a Snapshot Rule

An ARX volume can create snapshots on a scheduled basis, so you set it up as a rule. You can create snapshot rules in managed or direct volumes; snapshot rules do not require metadata, so direct volumes can support them.

However, direct volumes do not support the VSS-client interface; direct-volume clients access their snapshots through a special "~snapshot" directory described later in the chapter.

From gbl-ns-vol mode, use the snapshot rule command to create a new snapshot rule:

snapshot rule *name*

where ***name*** (1-64 characters) is a name you choose for the rule. This name is used in the names of the rule's individual snapshots, which can be viewed by ARX clients.

The CLI prompts for confirmation before creating a new snapshot rule; enter **yes** to continue. This puts you into gbl-ns-vol-snap mode, where you enable the new rule. There are also some optional commands you can invoke from this mode, such as applying a schedule.

For example, the following command sequence creates an empty snapshot rule, "dailySnap," for the volume, "medarcv~/lab_equipment:"

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot rule dailySnap

This will create a new policy object.

Create object 'dailySnap'? [yes/no] yes
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~dailySnap])# ...
```

Changing the Name of the Snapshot Directory (optional)

If the ".snapshot" directory that is created by default in each directory in the volume is not named to your liking, you can rename it.

These commands rename the hidden .snapshot directory in every directory in the current volume:

snapshot directory cifs-name

snapshot directory nfs-name

The syntax is:

snapshot directory cifs-name *newdirname*

snapshot directory nfs-name *newdirname*

where *newdirname* is the name that will be substituted for .snapshot in every directory in the current volume.

For example,

snapshot directory nfs-name ~ckpt

This will substitute the name "~ckpt" for ".snapshot" in every directory in the current volume.

Take care to avoid specifying a directory name that users might want to use for regular files or directories.

Changing the Number of Retained Snapshots (optional)

The ARX volume retains a limited number of snapshots for each snapshot rule. This conserves disk space on each back-end filer. The default is three snapshots. From gbl-ns-vol-snap mode, use the retain command to change the number of retained snapshots:

retain *number*

where ***number*** (1-1024) is the number of snapshots to retain. If you choose a number lower than the current number of snapshots, the volume removes any excess snapshots the next time the rule runs.

For example, the following command sequence causes the "dailySnap" rule to retain 5 snapshots:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot rule dailySnap
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~dailySnap])# retain 5
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~dailySnap])# ...
```

Reverting to the Default

Use no retain to keep the default number of snapshots for this rule (three):

no retain

For example:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot rule hourlySnap
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~hourlySnap])# no retain
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~hourlySnap])# ...
```

Excluding a Share (optional)

The volume may contain one or more shares that should be excluded from the coordinated snapshot. For example, a share could be backed by a back-end volume that is approaching a size limit, though the filer has other volumes where snapshots are permissible. To exclude a single share from the current snapshot rule, use the exclude command from gbl-ns-vol-snap mode:

exclude *share-name*

where ***share-name*** (1-64 characters) identifies the volume share to exclude. Use the share name from the ARX-volume configuration, not from the filer configuration; type **?** in place of the *share-name* for a valid list of share names.

For example, the following command sequence excludes the "backlots" share from the "hourlySnap" rule:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot rule hourlySnap
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~hourlySnap])# exclude backlots
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~hourlySnap])# ...
```

Including a Formerly-Excluded Share

For a share that was previously excluded, you can use the no exclude command to include it in future runs of the snapshot rule:

no exclude *share-name*

where ***share-name*** (1-64 characters) identifies the volume share to include in future snapshots.

For example, this command sequence ensures that the "leased" share is included in all future snapshots from the "hourlySnap" rule:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot rule hourlySnap
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~hourlySnap])# no exclude leased
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~hourlySnap])# ...
```

Sparse Snapshots

A *sparse snapshot* is a coordinated snapshot with at least one component snapshot missing. A snapshot rule creates a sparse snapshot if any of the following conditions exist when the rule runs:

- the rule has any exclude share,
- the volume uses an external-filer with no manage snapshots, or
- the back-end share is unresponsive to snapshot commands.

All of the files on the excluded shares appear to be missing from the client view of a sparse snapshot. For example, if a file named "\mydir\mysub\file.txt" is on Share A and another file named "\mydir\mysub\otherFile.txt" is on an excluded share, a coordinated snapshot only contains "\mydir\mysub\file.txt" from the included share. You can use file-placement rules to ensure that files that should be backed up reside on the volume's included shares: refer to in *Placing Files on Particular Shares, on page 14-4* of the *ARX® CLI Storage-Management Guide* for instructions on creating file-placement rules.

Applying a Schedule (optional)

The ARX volume can create snapshots on a fixed schedule. A schedule is not required because you can invoke a snapshot rule manually (as described later in the chapter).

To create a schedule, use the gbl schedule command (refer to *Appendix 12, Creating a Policy Schedule* of the *ARX® CLI Storage-Management Guide* for details). To apply a schedule to the snapshot rule, use the schedule command in gbl-ns-vol-snap mode:

schedule *name*

where ***name*** (1-64 characters) identifies the schedule. Use the show schedule command to list all schedules.

For example, the following command sequence applies a schedule, "daily4am," to the "dailySnap" rule:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot rule dailySnap
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~dailySnap])# schedule daily4am
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~dailySnap])# ...
```

You can change to a new schedule by re-running the command.

If the schedule has no specific start time, it starts when you first enable the snapshot rule (as described later in this section). That is, it takes its first snapshot at enable time. For instructions on setting the schedule's start time, see *Setting the Start Time (optional), on page 12-7* of the *ARX® CLI Storage-Management Guide*.

Snapshot Grouping

If two or more snapshot rules use the same schedule, the ARX performs all of their back-end snapshots in a single group. This ensures that there are no redundant snapshots on any back-end volumes.

Two different schedules with the same timing (for example, two weekly schedules with the same start times) do not have the same effect on their snapshot rules.

Important

*Snapshot grouping can be problematic for EMC-Celerra-backed volumes. As mentioned in EMC Celerra Best Practices, on page 2-5, EMC servers serialize their snapshots (called "checkpoints" in EMC documentation). If multiple ARX volumes have the same snapshot schedule **and** use shares from multiple file systems on an EMC, the ARX waits for the EMC to perform its checkpoints serially.*

For example, suppose 12 ARX volumes have the same snapshot schedule and are backed by 2 different EMC file systems each (for a total of 24 EMC file systems). Further suppose that all 24 EMC file systems reside on a single EMC file server. The EMC server waits for each checkpoint to complete before it begins the next one. The ARX-snapshot operation is not complete until all 24 EMC checkpoints are finished, even though each ARX volume requires only 2 EMC checkpoints.

Removing the Schedule

If you remove the schedule, you can only invoke snapshots manually. Use no schedule to remove the schedule from the current snapshot rule:

```
no schedule
```

For example:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot rule test_snap
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~test_snap])# no schedule
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~test_snap])# ...
```

Configuring a Progress Report (optional)

A progress reports shows all the milestones and results of a snapshot. We recommend this to verify that all filer snapshots succeeded, or to diagnose problems if one of them failed. From gbl-ns-vol-snap mode, use the report command to generate this report every time the rule creates a snapshot:

report *prefix*

where ***prefix*** (1-1024 characters) is the prefix to be used for the report. The report has a name in the following format: *prefix*_0_create_*YearMonthDayHourMinuteSecondsMilliseconds*.rpt (for example, mySnap_0_create_20071112133047318.rpt for a report with the "mySnap" prefix).

We recommend some common characters at the start of all your snapshot-report prefixes, to prepare for snapshot reconstitution (described later). For example, three snapshot rules could have the following report prefixes: "snapHourly," "snapDaily," and "snapWeekly."

Use the show reports command for a list of all reports, or show reports type Snapshot for a list of snapshot reports. Use show reports *report-name* to read the report, show reports status *report-name* to get a one-line summary of the report, grep to search through the full report, or tail reports *report-name* follow to follow a report as it is being written.

For example, the following command sequence enables a report for the "dailySnap" rule:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot rule dailySnap
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~dailySnap])# report snap_daily
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~dailySnap])# ...
```

Generating a Report Only When an Error Occurs

To save space on the ARX's internal disks, you can make the rule skip the report for snapshots that succeed. To accomplish this, use the optional error-only flag at the end of the report command:

report *prefix* error-only

where ***prefix*** is explained above.

For example, the following commands cause the hourlySnap rule to generate no reports unless there is an error:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot rule hourlySnap
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~hourlySnap])# report snap error-only
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~hourlySnap])# ...
```

Disabling Reports

From gbl-ns-vol-snap mode, use no report to prevent the snapshot rule from generating reports:

no report

For example, the following command sequence disables reporting for the rule, “test_snap:”

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot rule test_snap
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~test_snap])# no report
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~test_snap])# ...
```

Sample Snapshot-Create Report

The snapshot-create report contains configuration details for each rule, followed by details about the particular snapshot. The configuration details are in the first two tables in the report, “Snapshot Summary” and “Snapshot Properties.” The “Snapshot Summary” table identifies the namespace, volume, and snapshot rule. The “Snapshot Properties” table shows the configuration settings for the rule. The “Snapshot Summary - *rule-name*_0” table shows timestamps and the status of the coordinated snapshot. The “Included Shares” section has a separate table for each snapshot on each back-end share: these show the snapshots’ back-end locations. An “Excluded Shares” section and/or “Offline Shares” section appears for a *sparse snapshot*; these sections list all shares that were not used in the snapshot operation.

For example, this shows a snapshot-create report from the “dailySnap” rule:

```
bstnA(gbl)# show reports snap_daily_0_create_20120411005104661.rpt

  Snapshot Rule Summary
  ---------------------
   Namespace Name:                medarcv
   Volume Name:                   /lab_equipment
   Snapshot Rule Name:            dailySnap


  Snapshot Properties
  -------------------
   Snapshots Enabled:             Yes
   Guarantee Consistency:         Disabled
   Retain Count:                  7
   Schedule:                      daily4am
   CIFS Directory Name:           ~snapshot
   Directory Display:             All Exports
   Hidden File Attribute:         Not Set
   Restricted Access Configured:  Yes
   VSS Mode:                      None
   Contents:
     Metadata:                    No
     Volume Configuration:        No
     User Snapshots:              Yes
   Archive:
     Total Archive Operations:    0
     Total Successful Operations: 0
     Total Failed Operations:     0
     Total Saved Metadata:        0 B
     Total Saved Volume Config:   0 B
     Average Copy Rate:           0 b/s

  Snapshot Summary - dailySnap_0
  ------------------
   Snapshot Name:                 dailySnap_0
```

```
 Snapshot Operation:           Create
 Result:                       Success
 Time Requested:               04/11/2012 00:51:04 -0400
 Time Created:                 04/11/2012 00:51:04 -0400
 Last Time Verified:
 Request:                      Create
 Snapshot State:               Sparse
 Snapshot Origin:              Manual
 Report Name:                  snap_daily_0_create_20120411005104661.rpt


Included Shares
---------------
     Share Name:               equip (user data)
     Filer:
       Name:                   nas10
       CIFS Share:             equipment
       Volume:                 vol2
       Filer Snapshot:         acopia_1_201204110451_d9bdece8-9866-11d8-91e3-f48e42637d58_vol2

     Share Name:               leased (user data)
     Filer:
       Name:                   nas10
       CIFS Share:             for_lease
       Volume:                 vol1
       Filer Snapshot:         acopia_1_201204110451_d9bdece8-9866-11d8-91e3-f48e42637d58_vol1

     Share Name:               backlots (user data)
     Filer:
       Name:                   fs2
       CIFS Share:             backlot_records
       Volume:                 E:\
       Filer Snapshot:         {0db33dfa-b2f9-45e1-bb35-b061c03dd732}
       Time Created:           04/11/2012 00:51:07 -0400


Excluded Shares
---------------
     Share Name:               scanners
     Filer:
       Name:                   fs5
       CIFS Share:             xraysScanners
     Reason:                   Snapshots were not supported on this type of back-end filer.
```

Snapshot Naming on the Filer

The Volume Snapshot fields in the “Included Shares” section show the names of the snapshot directories on the back-end filers. On EMC servers, Data Domain systems, and NetApp filers, these are created with the following format:

acopia_*id_time-stamp_uuid_filer-volume*

where

id is an integer that the snapshot software uses internally.

time-stamp is in *yyyymmddHHMM* format. This is the time that the snapshot was created.

uuid is the Universally-Unique ID that identifies the ARX. In a redundant pair, this is the UUID for the peer that was originally configured as senior/active, no matter which peer is currently active. An ARX's UUID appears in the output of the show chassis chassinfo command.

filer-volume is the name of the filer's volume.

A Windows server uses a UUID surrounded by curly braces ({}) to identify each of its snapshots. For example:

```
{d316724d-1e31-478a-853a-1f7d4418c599}
```

Enabling the Snapshot Rule

The final step in configuring a snapshot rule is to enable it. By default, the rule is disabled and ignored by policy software. You must enable the rule to run snapshots on a schedule *or* to invoke the snapshots manually. Use the enable command to enable the rule:

enable

For example, the following command sequence enables the "dailySnap" rule:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot rule dailySnap
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~dailySnap])# enable
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~dailySnap])# ...
```

Disabling the Rule

Disabling the rule prevents it from creating any snapshots. Use no enable from gbl-ns-vol-snap mode to disable a snapshot rule:

no enable

For example, the following command sequence disables the "test_snap" rule:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot rule test_snap
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~test_snap])# no enable
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~test_snap])# ...
```

Removing all Snapshots Behind a Rule

You can use the priv-exec command, snapshot remove, to delete all of the back-end snapshots behind a snapshot rule, without removing the rule configuration. You can use this to clear snapshots from several filers at once. From priv-exec mode, use the snapshot remove command to remove the filer snapshots and/or checkpoints behind a snapshot rule:

snapshot remove *namespace vol-path rule*

where

namespace (1-30 characters) is the namespace,

vol-path (1-1024 characters) selects the ARX volume, and

rule (1-64 characters) is the snapshot rule.

The CLI prompts for confirmation before removing the snapshots and/or checkpoints from the back-end filers. Enter **yes** to proceed with the snapshot removal. Then it displays the names of one or more removal reports, one per ARX snapshot, which show the results of each removal operation. Use show reports *report-name* to view any report, or tail reports *report-name* follow to watch the removal operation as it progresses.

For example, this command sequence exits to priv-exec mode and removes all snapshots behind the "hourlySnap" rule:

```
bstnA(gbl)# end
bstnA# snapshot remove medarcv /lab_equipment hourlySnap

Confirmation of this command results in the removal all snapshots associated with snapshot rule
'hourlySnap' in namespace 'medarcv' volume '/lab_equipment'.  The snapshot rule is not deleted.

 Proceed? [yes/no] yes

  Starting snapshot operation in volume /lab_equipment, report name:
snap_hourly_2_remove_20090326082729516.rpt


  Starting snapshot operation in volume /lab_equipment, report name:
snap_hourly_1_remove_20090326082729516.rpt


  Starting snapshot operation in volume /lab_equipment, report name:
snap_hourly_0_remove_20090326082729516.rpt

bstnA(gbl)# ...
```

Removing the Filer Snapshots Behind One ARX Snapshot

You can remove the back-end snapshots from behind a single ARX-volume snapshot. Add the name of a specific snapshot to the end of the snapshot remove command:

snapshot remove *namespace vol-path rule snapshot*

where

namespace vol-path, and ***rule*** are described above, and

snapshot (1-68 characters) is the specific snapshot to remove. Snapshots are typically named *rule-name_n*, where *rule-name* is the same as ***rule*** and *n* is the number of the specific snapshot (for example, "hourlySnap_2"). Type **?** for a complete list of snapshots in the current snapshot rule.

The CLI prompts for confirmation before removing the snapshots and/or checkpoints from the back-end filer(s). Enter **yes** to proceed with the snapshot removal. Then it displays the name of the removal report, which shows the results of the removal operation.

For example, this command sequence exits to priv-exec mode and removes all filer snapshots behind "dailySnap_2," an ARX-volume snapshot:

```
bstnA(gbl)# end
bstnA# snapshot remove medarcv /lab_equipment dailySnap dailySnap_2


Confirmation of this command results in the removal of snapshot 'dailySnap_2' created by snapshot
rule 'dailySnap' in namespace 'medarcv' volume '/lab_equipment'.

Proceed? [yes/no] yes

  Starting snapshot operation in volume /lab_equipment, report name:
snap_daily_2_remove_20090326095909994.rpt

bstnA# ...
```

Sample Snapshot-Remove Report

The snapshot-remove report has a similar format to the snapshot-create report, shown in *Sample Snapshot-Create Report, on page 2-12*. This report contains configuration details for each rule, followed by details about the particular ARX-snapshot removal. The configuration details are in the first two tables in the report, “Snapshot Summary” and “Snapshot Properties.” The “Snapshot Summary” table shows identifies the namespace, volume, and snapshot rule. The “Snapshot Properties” table shows the configuration settings for the rule. The “Snapshot Summary - *rule-name*_0” table shows timestamps and the status of the overall snapshot removal. The “Included Shares” section has a separate table for each snapshot on each back-end share: these show the results and timing of each filer-snapshot removal. If any of the volume’s shares were administratively excluded from the snapshot, an “Excluded Shares” section lists them. Similarly, an “Offline Shares” section appears if any of the volume’s shares were unreachable when the original snapshot was taken.

For example, this shows a snapshot-remove report from the removal of the “dailySnap_2” snapshot:

```
bstnA(gbl)# show reports snap_daily_2_remove_20120411005258066.rpt

  Snapshot Rule Summary
  ---------------------
   Namespace Name:               medarcv
   Volume Name:                  /lab_equipment
   Snapshot Rule Name:           dailySnap


  Snapshot Properties
  -------------------
   Snapshots Enabled:            Yes
   Guarantee Consistency:        Disabled
   Retain Count:                 7
   Schedule:                     daily4am
   CIFS Directory Name:          ~snapshot
   Directory Display:            All Exports
   Hidden File Attribute:        Not Set
   Restricted Access Configured: Yes
   VSS Mode:                     None
   Contents:
     Metadata:                   No
     Volume Configuration:       No
     User Snapshots:             Yes
```

```
  Archive:
    Total Archive Operations:    0
    Total Successful Operations: 0
    Total Failed Operations:     0
    Total Saved Metadata:        0 B
    Total Saved Volume Config:   0 B
    Average Copy Rate:           0 b/s

 Snapshot Summary - dailySnap_2
 ------------------
  Snapshot Name:                 dailySnap_2
  Snapshot Operation:            Delete
  Result:                        Success
  Time Requested:                04/11/2012 00:51:04 -0400
  Time Created:                  04/11/2012 00:51:04 -0400
  Last Time Verified:
  Request:                       Delete
  Snapshot State:
  Snapshot Origin:               Manual
  Report Name:                   snap_daily_2_remove_20120411005258066.rpt


 Included Shares
 ---------------
      Share Name:                equip (user data)
      Filer:
        Name:                    nas10
        CIFS Share:              equipment
        Volume:                  vol2
        Filer Snapshot:          acopia_1_201204110451_d9bdece8-9866-11d8-91e3-f48e42637d58_vol2

      Share Name:                leased (user data)
      Filer:
        Name:                    nas10
        CIFS Share:              for_lease
        Volume:                  vol1
        Filer Snapshot:          acopia_1_201204110451_d9bdece8-9866-11d8-91e3-f48e42637d58_vol1

      Share Name:                backlots (user data)
      Filer:
        Name:                    fs2
        CIFS Share:              backlot_records
        Volume:                  E:\
        Filer Snapshot:          {0db33dfa-b2f9-45e1-bb35-b061c03dd732}
        Time Created:            04/11/2012 00:51:07 -0400


 Excluded Shares
 ---------------
      Share Name:                scanners
      Filer:
        Name:                    fs5
        CIFS Share:              xraysScanners
      Reason:                    Snapshots were not supported on this type of back-end filer.

bstnA# ...
```

Removing a Snapshot Rule

Removing a snapshot rule deletes its configuration from the ARX without affecting any back-end snapshots. This also removes the Previous Versions tab from the client-side view of the ARX volume.

From gbl-ns-vol mode, use the no snapshot rule command to remove a snapshot rule from the current volume:

no snapshot rule *name*

where ***name*** (1-64 characters) identifies the rule to remove.

The CLI prompts for confirmation before deleting the snapshot rule. The confirmation prompt shows all of the individual snapshots that will remain on the filers after the rule is gone. You will need to access the filer directly if you decide to remove these snapshots later. Enter **yes** to continue.

For example, the following command sequence removes the "test_snap" rule from the "medarcv~/lab_equipment" volume:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# no snapshot rule test_snap

The following snapshots are being managed by the rule 'test_snap':
        test_snap_2(acopia_7_200707101300_501f705c-8735-11d8-8936-a58a9e4556df_datavol2)
        test_snap_1(acopia_8_200707101400_501f705c-8735-11d8-8936-a58a9e4556df_datavol2)
        test_snap_0(acopia_10_200707101500_501f705c-8735-11d8-8936-a58a9e4556df_datavol2)

If this command is confirmed, the rule is deleted and all associated data related to the
aforementioned snapshots will be removed from the switch.  The snapshots on the filers will not be
removed.

Proceed? [yes/no] yes

bstnA(gbl-ns-vol[medarcv~/lab_equipment])# ...
```

Incorporating a Pre-Existing Snapshot

You can incorporate an existing, back-end snapshot into a snapshot rule. After incorporating all of the pre-existing snapshots into the ARX volume, Windows clients can access them as they did before the introduction of the ARX. From priv-exec mode, use the snapshot manage command to import a back-end snapshot into a snapshot rule:

snapshot manage *namespace vol-path share rule filer-snap* created-on *date-time*

where

namespace (1-30 characters) is the namespace,

vol-path (1-1024 characters) selects the ARX volume, and

share (1-64 characters) is the ARX share. The volume software seeks the back-end snapshot on the filer behind this share.

rule (1-1024 characters) is the snapshot rule to accept the back-end snapshot.

filer-snap (1-255 characters) is name of the snapshot or checkpoint at the back-end filer or file server.

date-time is the date and time that the ARX snapshot was created, in *mm*/*dd*/*yyyy*:*HH*:*MM*:*SS* format. This identifies a particular ARX snapshot from the above *rule*. You can use show snapshots (described later) to see the time stamps for existing snapshots. If no ARX snapshot exists for the given time, this command creates a new one.

This command runs synchronously; the CLI prompt does not return until the snapshot import is complete. If it fails, a report name appears. You can use show reports *report-name* to examine the report and find the cause of the failure.

For example, this command sequence exits to priv-exec mode and imports an existing snapshot ("weekly.1" on the filer behind the "leased" share) into a rule named "weekly_snap" in the "medarcv~/lab_equipment" volume:

```
bstnA(gbl)# end
bstnA# snapshot manage medarcv /lab_equipment leased weekly_snap weekly.1 created-on
03/30/2009:01:05:24
bstnA# ...
```

Using a Different Prefix for the Error Report

In case of an error, the CLI creates a report. By default, the report name follows this format:

snapshot-manage-*namespace-volume-rule-yyyymmddHHMM*.rpt.

To change the "snapshot-manage" prefix, you can use the report-prefix option at the end of the snapshot-manage command:

```
snapshot manage namespace ... rule filer-snap created-on date-time [report-prefix prefix]
```

where

all options are described above except

prefix (optional, 1-64 characters), which is the new prefix for the report name.

For example, this command uses a report prefix of "getWk3" for a manage-snapshot operation, though no report is generated when the operation succeeds:

```
bstnA(gbl)# end
bstnA# snapshot manage medarcv /lab_equipment leased weekly_snap weekly.3 created-on
03/16/2009:01:05:22 report-prefix getWk3
bstnA# ...
```

Generating a Report When the Operation Succeeds

The volume software only creates a report when the snapshot manage operation fails, by default. To create a report whether or not the operation fails, you can use the verbose option at the end of the command:

snapshot manage *namespace ... filer-snap* created-on *date-time* [report-prefix *prefix*] [verbose]

The CLI shows the report name after you enter the command. For example:

```
bstnA(gbl)# end
bstnA# snapshot manage medarcv /lab_equipment leased weekly_snap weekly.3 created-on
03/16/2009:01:05:22 verbose
Results for this operation will be written to the report file
snapshot-manage-medarcv-/lab_equipment-weekly_snap-200803171422.rpt
bstnA# ...
```

Clearing the Snapshot from the ARX Configuration

If you mismatched the back-end snapshot with its ARX counterpart, you can use a command to disassociate the snapshot from the ARX configuration. From priv-exec mode, the clear snapshot command clears one or more back-end snapshots from the ARX:

snapshot clear *namespace* [*vol-path* [*rule* [snapshot *snap-name* [share *share-name*]]]]

where

namespace (1-30 characters) identifies the namespace where you want to clear at least one snapshot record.

vol-path (optional, 1-1024 characters) focuses on one ARX volume. The command clears all snapshot records in the above namespace if you omit this.

rule (optional, 1-64 characters) is a snapshot rule with one or more snapshots to clear. If omitted, the command clears all of the volume's filer-snapshot records.

snap-name (optional, 1-255 characters) identifies a particular ARX snapshot where you want to clear filer-snapshot associations. If you omit this, the command clears the filer-snapshot records for all retained snapshots in the above rule.

share (optional, 1-64 characters) focuses the command on a single ARX share. This causes the command to clear the single record of a particular snapshot on this share. You can use the show snapshots command to see the exact name of the snapshot on this share. If you omit this, the command clears the records of all the filer snapshots associated with the chosen ARX snapshot, and therefore removes this particular ARX snapshot from the rule.

The CLI prompts for confirmation before clearing any back-end snapshots from the ARX configuration; enter **yes** to proceed. This does not affect the back-end snapshots themselves, nor does it affect the configuration of any snapshot rule.

For example, the following command clears all of the filer snapshot records from the "dailySnap_2" snapshot.

```
bstnA# snapshot clear medarcv /lab_equipment dailySnap snapshot dailySnap_2

Confirmation of this command results in the clearing of snapshot 'dailySnap_2' associated with
snapshot rule 'dailySnap' in namespace 'medarcv' volume '/lab_equipment'.  This command does not
remove the snapshot(s) from the filer(s) in the volume.
```

```
Proceed? [no] yes
bstnA# ...
```

As a result, the “dailySnap_2” snapshot is entirely cleared from the ARX database (though the rule configuration is unaffected):

Supporting VSS for Pre-XP Clients

This section only applies to a managed volume; direct volumes do not support VSS for their clients. Skip to the next section if you are configuring a direct volume.

Windows-2000 client machines use an older software protocol for exchanging snapshot data, one that requires more server-side processing to support the "Previous Versions" view of snapshots. Windows XP and later clients use a newer protocol that simplifies processing for the CIFS server. To avoid extra processing at the CIFS service, volumes do not support pre-Windows-XP clients by default. From gbl-ns-vol mode, you can use the following command to support Windows 2000 clients as well as some later versions of Windows:

```
snapshot vss-mode pre-xp
```

◆ Note

This option makes VSS unusable for Windows 7 or later clients. Use this option only if you have Windows 2000 clients and no Windows 7 or later clients.

For example, the following command sequence supports VSS for Windows 2000 clients of the "medarcv~/lab_equipment" volume:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot vss-mode pre-xp
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# ...
```

Restricting VSS Support to XP and Later Clients

As mentioned above, support of the VSS interface (the Previous Versions tab in file/directory Properties) incurs a slight performance penalty if extended to Windows 2000 clients. By default, a volume only supports VSS for releases after Windows 2000, such as Windows XP and Windows 2003. Use the following command (also in gbl-ns-vol mode) to return to this default:

```
snapshot vss-mode xp
```

◆ Note

There is another method for the volume's Windows-2000 clients to access snapshots. The volume offers snapshots through a pseudo directory, "~snapshot." This directory is described later in the chapter.

For example, the following command sequence edits the "access~/G" volume and stops VSS support from pre-Windows-XP clients:

```
bstnA(gbl)# namespace access volume /G
bstnA(gbl-ns-vol[access~/G])# snapshot vss-mode xp
bstnA(gbl-ns-vol[access~/G])# ...
```

Restricting Access to Snapshots

You have several options for restricting snapshot access:

- For CIFS clients, you can identify the administrative clients in a Windows-Management-Authorization (*WMA*) group, and limit snapshot access to those clients.
- Also for CIFS clients, you can stop supporting the VSS ("Previous Versions") interface altogether.
- Leave the "~snapshot" pseudo directory as the only means to access snapshots. This is the ordinary practice for NFS clients, but may not be for CIFS clients. This directory exists in every directory in the volume, but it is not displayed by default. There are four methods for managing the visibility of the ~snapshot directory:
  - For CIFS shares, use WMA groups.
  - For NFS or CIFS, display the ~snapshot directory in the root of any share.
  - For NFS or CIFS, Display the ~snapshot directory only in shares that export the volume root.
  - For CIFS shares, use the "Hidden" flag on the ~snapshot directory.

  Direct volumes do not support VSS. The ~snapshot directory is the only means that direct-volume clients have for accessing their snapshots.

You can use any or all of these management options, as described in the subsections below.

Client-Based Access Control: WMA Groups

You can use a Windows-Management-Authorization (*WMA*) group to restrict snapshot access to a limited number of clients, such as administrators. A WMA group is typically used to identify users who can use MMC and similar management tools in a namespace. *Authorizing Windows-Management (MMC/Snapshot) Access, on page 3-25* of the *ARX® CLI Storage-Management Guide* describes how to create a WMA group, and *Opening Windows-Management Access (optional, MMC/Snapshots), on page 7-16* describes how to assign one to a namespace. You can create a WMA group for authorized snapshot clients, permit access to snapshots for that group, assign it to the volume's namespace, and then enforce the WMA group for snapshots in the current volume. All of these steps are described below.

Adding a WMA Group for Snapshots

From gbl mode, you can use the windows-mgmt-auth command to create a new WMA group. This places you into gbl-mgmt-auth mode, where you can use the user command once for each Windows client in the WMA group, and the permit snapshot command to allow them access to snapshots. For example, this creates a WMA group named "snapViewers" with two users:

```
bstnA(gbl)# windows-mgmt-auth snapViewers
bstnA(gbl-mgmt-auth[snapViewers])# user juser windows-domain MEDARCH.ORG
bstnA(gbl-mgmt-auth[snapViewers])# user jquser windows-domain MEDARCH.ORG
bstnA(gbl-mgmt-auth[snapViewers])# permit snapshot monitor
bstnA(gbl-mgmt-auth[snapViewers])# exit
bstnA(gbl)# ...
```

These commands are detailed in *Authorizing Windows-Management (MMC/Snapshot) Access, on page 3-25* of the *ARX® CLI Storage-Management Guide*.

Assigning the WMA Group to the Namespace

From gbl-ns mode, you can use another windows-mgmt-auth command to assign a WMA group to your namespace. For example, this command sequence assigns the "snapViewers" WMA group to the "medarcv" namespace:

```
bstnA(gbl)# namespace medarcv
bstnA(gbl-ns[medarcv])# windows-mgmt-auth snapViewers
bstnA(gbl-ns[medarcv])# ...
```

This is described in detail in *Opening Windows-Management Access (optional, MMC/Snapshots), on page 7-16* of the *ARX® CLI Storage-Management Guide*.

Restricting Snapshot Access to Members of the WMA Group

The snapshot processes ignore the namespace's WMA group(s) by default. From gbl-ns-vol-snap mode, use the snapshot privileged access command to restrict all snapshot access to privileged Windows users (that is, to users in a WMA group with permit snapshot monitor):

```
snapshot privileged-access
```

For example, the following command sequence restricts access to all snapshots in "medarcv~/lab_equipment:"

```
bstnA(gbl-ns[medarcv])# volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot privileged-access
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# ...
```

Re-Opening Snapshot Access

Use no snapshot privileged-access to allow all users to access the volume's snapshots:

```
no snapshot privileged-access
```

This is the default.

For example, this allows any Windows client to access the snapshots in "access~/G:"

```
bstnA(gbl)# namespace access volume /G
bstnA(gbl-ns-vol[access~/G])# no snapshot privileged-access
bstnA(gbl-ns-vol[access~/G])# ...
```

Shutting Down Support for the VSS Interface

You can also limit client access to snapshots by shutting down support for the VSS interface.

Well-informed clients always have access to the ~snapshot directory, which is not displayed by default. The ~snapshot directory is a pseudo directory under every directory of the volume. Each subdirectory under ~snapshot is one snapshot for the directory, named *rule-name_n* (such as hourlySnap_0, hourlySnap_1, and so on). Administrators can traverse these ~snapshot directories and recover files as needed.

From gbl-ns-vol mode, use the snapshot vss-mode none command to shut down support for VSS (the "Previous Versions" interface), leaving only the obscure ~snapshot directory for accessing snapshots:

```
snapshot vss-mode none
```

For example, this command sequence stops VSS access to snapshots in the "medarcv~/lab_equipment" volume:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot vss-mode none
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# ...
```

Then use one of the snapshot directory display commands to display the root ~snapshot directory to administrators. The following subsections describe these commands.

Displaying the Root ~snapshot Directory

Well-informed clients can use the special ~snapshot directory to access their snapshots. This is a subdirectory under every directory in the volume. The volume does not display these pseudo directories by default, but they are accessible to any user that knows their exact path names.

◆ Note

Direct volumes do not support VSS, so the ~snapshot directory is the only method that direct-volume clients can use to access their snapshots.

From gbl-ns-vol mode, use the following command to display the ~snapshot directory in the root of any CIFS share:

```
snapshot directory display all-exports
```

If the volume has snapshot privileged-access, described earlier, only privileged clients can see or access the ~snapshot directories. The ~snapshot directory appears on the next CIFS **dir** command or the next refresh of the graphical view.

For example, the following command sequence shows the ~snapshot directory in all exports of the "/lab_equipment" volume:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot directory display all-exports
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# ...
```

Contents of the ~snapshot Directory

The ~snapshot directory contains one subdirectory for each snapshot. Its subdirectories are named *rulename_n*, where *n* is the number of the snapshot. Snapshot 0 (zero) is the most-recent snapshot, and the numbers are progressively higher for older snapshots. For example, clients of the "medarcv~/lab_equipment" volume with the "dailySnap" rule would see snapshots in directories named dailySnap_0, dailySnap_1, and so on. Every time the rule runs, it creates a new snapshot and moves it to the "..._0" directory (dailySnap_0), moves all the intermediate directories (the earlier dailySnap_0 moves to dailySnap_1, dailySnap_1 moves to dailySnap_2, and so on), and removes the oldest snapshot. The "Date Modified" stamp for each directory is the time the snapshot was created.

Displaying ~snapshot Only in Volume-Root Exports

Some installations share out directories below the root of the volume (such as "lab_equipment\incubators," "lab_equipment\miscDiags," and "lab_equipment\chemlab") to their CIFS clients, and share out the root of the volume, "lab_equipment," to administrators only. This is particularly popular for home directories. Administrators can use the volume-root export to maintain all of the clients' lower-level exports. At these sites, you can limit access to the snapshot directory by showing it only in the volume-root export. Administrators who use the volume-root export can see the "~snapshot" directory, but clients with other exports cannot.

From gbl-ns-vol mode, use the following command to show the snapshot directory in root exports only:

```
snapshot directory display volume-root-only
```

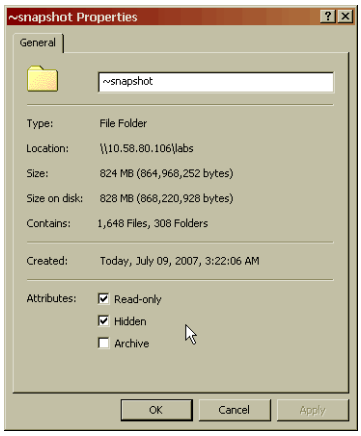
As above, this change is visible to clients (that is, the ~snapshot directory appears) on the client's next CIFS **dir** command (or the next refresh of the graphical view).

For example, the following command sequence restricts snapshots to volume-root exports of the "medarcv~/lab_equipment" volume:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot directory display volume-root-only
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# ...
```

Setting the "Hidden" Flag for the Snapshot Directory

You can further obscure the snapshot directory by raising its DOS "hidden" flag. Windows clients who are configured to view this type of "hidden" directory can see it, while other clients cannot.



Add the optional hidden argument to the end of the snapshot directory display command to raise the DOS "hidden" flag for the ~snapshot directory:

```
snapshot directory display {all-exports | volume-root-only} hidden
```

If the volume is active when you run this command, the snapshot directory disappears from client view on the next CIFS **dir** command (or the next refresh of the graphical view).

For example, the following command reveals the ~snapshot directory to the client application for all exports of "medarcv~/lab_equipment," but instructs the client applications to hide the directory from users:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot directory display all-exports hidden
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# ...
```

Lowering the "Hidden" Flag

By default, the ~snapshot directory does not have the "hidden" flag raised. All clients can see the directory at the root of the CIFS share. To lower the "hidden" flag, re-run the snapshot directory display command without the hidden argument at the end:

```
snapshot directory display {all-exports | volume-root-only}
```

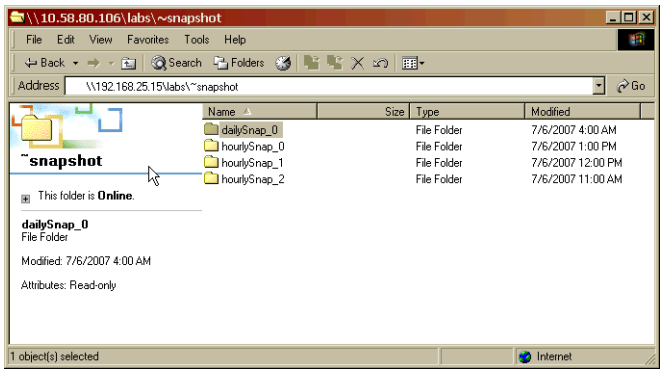
As above, the ~snapshot directory appears on the client's next CIFS **dir** command, or the next refresh of the graphical view.

For example, the following command sequence removes the "hidden" flag from the "/lab_equipment" volume's snapshots:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot directory display all-exports
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# ...
```

Changing the Name of the ~snapshot Directory

The "~snapshot" directories contain subdirectories with one or more snapshots:



From gbl-ns-vol mode, you can use the snapshot directory name command to change the name of the volume's ~snapshot directories:

snapshot directory name *new-name*

where ***new-name*** (1-32 characters) is the new name for this volume's snapshot directory. Avoid any characters that are not supported in CIFS (any control character, /, \, :, *, >, <, ", |, or ?). Also avoid any name that might already be in use by clients.

For example, the following command sequence uses "~checkpt" as the directory container for snapshots in "medarcv~/lab_equipment:"

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot directory name ~checkpt
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# ...
```

Reverting to the Default Directory Name

Use no snapshot directory name to return to the default name, "~snapshot:"

no snapshot directory name

For example:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# no snapshot directory name
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# ...
```

Enabling Snapshot Consistency

Use the snapshot consistency command to guarantee snapshot consistency in the current volume. A consistent snapshot is one with no file or directory changes between the start and end of the coordinated-snapshot operation. Consistency is achieved through a *VIP fence*, which momentarily blocks client access to the volume's VIP while the back-end filers perform their snapshot and checkpoint operations. The fence remains up until the last filer indicates that its snapshot is complete, or until a timeout expires, whichever comes first.

Important

This affects all volumes behind the volume's VIP(s). This is only recommended for databases or similar applications that need it; consult F5 Support for guidance.

This causes the volume's next snapshot operation to raise the VIP fence:

```
snapshot consistency
```

For example, the following command sequence enables snapshot consistency in the "medarcv~/lab_equipment" volume:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot consistency
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# ...
```

Fencing Timeouts

The VIP fence stops when the last filer snapshot is finished, or after the coordinated snapshot times out. The timeout occurs in 1 minute plus 80 seconds per filer snapshot.

If multiple snapshot rules use the same schedule, the ARX invokes their filer snapshots in a group. This protects against duplicate snapshots on back-end filers. However, it also extends the fencing timeout, and the time that clients cannot access any ARX volumes behind the VIP(s). This can be time-consuming for volumes backed by EMC Celerra servers, as discussed earlier (recall *Snapshot Grouping, on page 2-10*).

Disabling VIP Fencing (and Consistency)

You have the option to disable VIP fencing in the volume, thereby also disabling snapshot consistency. Without fencing, it is possible for a client to make changes on one back-end filer while another filer is still taking its snapshot. This may not be crucial for your installation; for example, consistency is rarely an issue at a site where clients use snapshots to recover one or two files at a time.

The policy engine always pauses for a volume during a coordinated snapshot, whether or not consistency is enabled.

For volumes where snapshot consistency is not required, you can use the no snapshot consistency command to disable VIP fencing:

```
no snapshot consistency
```

For example, the following command sequence disables snapshot consistency in the "medarcv~/lab_equipment" volume:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# no snapshot consistency
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# ...
```

Creating a Manual Snapshot

You can manually invoke a snapshot rule and create a coordinated snapshot at any time. From priv-exec mode, use the snapshot create command to manually invoke a snapshot rule:

snapshot create ***namespace vol-path rule*** **[***name***]**

where

namespace (1-30 characters) is the namespace,

vol-path (1-1024 characters) selects the volume, and

rule (1-64 characters) is the snapshot rule to invoke.

name (optional; 1-68 characters) is a name you choose for this snapshot. By default, the name is ***rule***_0 (for example, "dailySnap_0"). You cannot choose the name of an existing snapshot.

The CLI prompts for confirmation. If the rule has a schedule assigned to it (recall *Applying a Schedule (optional), on page 2-9*), the confirmation notifies you that the unscheduled snapshot affects the rule's retained snapshots. For a rule with a full count of retained snapshots, this removes the oldest retained snapshot and creates a new snapshot to replace it. This breaks the scheduled chain of snapshots, so it is only recommended for situations where the most-recent snapshot failed.

Enter **yes** at the prompt in order to proceed with the snapshot. The CLI then displays the name of the snapshot report; use show reports *report-name* to view this report, which contains details about the snapshot operation.

For example, this command sequence exits to priv-exec mode and manually runs the "dailySnap" rule in the "medarcv~/lab_equipment" volume:

```
bstnA(gbl)# end
bstnA# snapshot create medarcv /lab_equipment dailySnap

Snapshot rule 'dailySnap' in namespace 'medarcv' volume '/lab_equipment' has a schedule
associated with it.  Running this rule may cause existing snapshots to be
removed if the retain count is exceeded by this request.
Proceed? [yes/no]  yes

  Starting snapshot operation in volume /lab_equipment, report name:
snap_daily_0_create_20090327005716287.rpt

bstnA# ...
```

Verifying a Rule's Snapshots

It is possible for a back-end filer or file server to automatically delete one of its snapshots or checkpoints, thus corrupting the aggregated snapshot in the ARX volume. To verify that all snapshots and/or checkpoints still exist behind a snapshot rule, use the priv-exec snapshot verify command:

snapshot verify *namespace vol-path rule*

where

namespace (1-30 characters) is the namespace,

vol-path (1-1024 characters) selects the volume, and

rule (1-64 characters) is the snapshot rule to verify.

The CLI displays the name of the verification report(s), one per snapshot. Each report contains the results of one snapshot verification. Use show reports *report-name* to view any report.

For example, this command sequence exits to priv-exec mode, verifies all of the snapshots for the "hourlySnap" rule, and shows the report for "snap_0," the most-recent snapshot. The report shows that the verification succeeded.

```
bstnA(gbl)# end
bstnA# snapshot verify medarcv /lab_equipment hourlySnap

  Starting snapshot operation in volume /lab_equipment, report name:
snap_hourly_2_verify_20091006020708453.rpt


  Starting snapshot operation in volume /lab_equipment, report name:
snap_hourly_1_verify_20091006020708453.rpt


  Starting snapshot operation in volume /lab_equipment, report name:
snap_hourly_0_verify_20091006020708453.rpt

bstnA# show reports snap_hourly_0_verify_20091006020708453.rpt

  Snapshot Summary
  ----------------
   Namespace Name:               medarcv
   Volume Name:                  /lab_equipment
   Snapshot Rule Name:           hourlySnap


  Snapshot Properties
  -------------------
   Snapshots Enabled:            Yes
   Guarantee Consistency:        Disabled
   Retain Count:                 3
   Schedule:                     hourly
   Directory Name:               ~snapshot
   Directory Display:            All Exports
   Hidden File Attribute:        Not Set
   Restricted Access Configured: Yes
   VSS Mode:                     None
   Contents:
     Metadata:                   No
     Volume Configuration:       No
```

```
    User Snapshots:              Yes
  Archive:
    Total Archive Operations:    0
    Total Successful Operations: 0
    Total Failed Operations:     0
    Total Saved Metadata:        0 B
    Total Saved Volume Config:   0 B
    Average Copy Rate:           0 b/s

 Snapshot Summary - hourlySnap_0
 -----------------
  Snapshot Name:                 hourlySnap_0
  Snapshot Operation:            Verify
  Result:                        Success
  Time Requested:                Tue Oct  6 02:07:08 2009
  Time Created:                  Tue Oct  6 02:00:00 2009
  Last Time Verified:            Tue Oct  6 02:07:12 2009
  Request:                       Verify
  Snapshot State:                Sparse
  Snapshot Origin:               Schedule
  Report Name:                   snap_hourly_0_verify_20091006020708453.rpt


 Included Shares
 ---------------
      Share Name:                equip (user data)
      Filer:
        Name:                    nas10
        CIFS Share:              equipment
        Volume:                  vol2
        Filer Snapshot:          acopia_9_200910060600_d9bdece8-9866-11d8-91e3-f48e42637d58_vol2

      Share Name:                backlots (user data)
      Filer:
        Name:                    fs2
        CIFS Share:              backlot_records
        Volume:                  E:\
        Filer Snapshot:          {7435cc35-80d5-472f-a090-1a32946b3dbd}
        Time Created:            Tue Oct  6 02:00:01 2009

      Share Name:                leased (user data)
      Filer:
        Name:                    nas10
        CIFS Share:              for_lease
        Volume:                  vol1
        Filer Snapshot:          acopia_9_200910060600_d9bdece8-9866-11d8-91e3-f48e42637d58_vol1


 Excluded Shares
 ---------------
      Share Name:                scanners
      Filer:
        Name:                    fs5
        CIFS Share:              xraysScanners
      Reason:                    Snapshots were not supported on this type of back-end filer.

bstnA# ...
```

Verifying a Single Snapshot

To focus on a single snapshot, you can add the name of the snapshot to the end of the snapshot verify command:

```
snapshot verify namespace vol-path rule snapshot
```

where

namespace vol-path, and ***rule*** are described above, and

snapshot (1-68 characters) is a specific snapshot. Snapshots are typically named *rule-name_n*, where *rule-name* is the same as ***rule*** and *n* is the number of the specific snapshot (for example, "hourlySnap_2"). The *n* is 0 (zero) for the newest snapshot and progressively higher for older snapshots. Type **?** for a complete list of snapshots for the current snapshot rule.

The CLI displays the name of the verification report after you enter the command.

For example, this command sequence exits to priv-exec mode and verifies "dailySnap_0" snapshot:

```
bstnA(gbl)# end
bstnA# snapshot verify medarcv /lab_equipment dailySnap dailySnap_0

  Starting snapshot operation in volume /lab_equipment, report name:
snap_daily_0_verify_20091006020708676.rpt

bstnA# ...
```

Running Snapshot Operations from a Remote Host

From any remote host that supports the Secure Shell (SSH) protocol, you can run snapshot create, snapshot remove, or snapshot verify and see the report(s). Use the following syntax with **ssh**:

```
ssh admin-user@mip "snapshot ..."
```

where

admin-user is the username for a valid administrative account at the ARX (use show users to list all of them, as shown in *Listing All Administrative Users, on page 2-14* of the *ARX® CLI Network-Management Guide*),

mip is a management-IP address for the ARX (use show interface mgmt to show the out-of-band management interface, or show interface vlan to show all in-band management interfaces), and

snapshot ... is the desired snapshot command, described above. Surround this with quotation marks ("").

The output of the CLI command appears in the local shell. This also works with all show commands.

For example, the following command sequence runs the snapshot create command from a remote machine, “mgmt17.”

```
juser@mgmt17:~$ ssh admin@10.1.1.7 “snapshot create medarcv /lab_equipment dailySnap”
Command>snapshot create medarcv /lab_equipment dailySnap

  Starting snapshot operation in volume /lab_equipment, report name:
snap_daily_0_create_20090327005822216.rpt

juser@mgmt17:~$ ssh admin@10.1.1.7 “show reports snap_daily_0_create_20120423012706913.rpt”
Command>show reports snap_daily_0_create_20120423012706913.rpt

 Snapshot Rule Summary
  --------------------
   Namespace Name:               medarcv
   Volume Name:                  /lab_equipment
   Snapshot Rule Name:           dailySnap


  Snapshot Properties
  -------------------
   Snapshots Enabled:            Yes
   Guarantee Consistency:        Disabled
   Retain Count:                 7
   Schedule:                     daily4am
   CIFS Directory Name:          ~snapshot
   Directory Display:            All Exports
   Hidden File Attribute:        Not Set
   Restricted Access Configured: Yes
   VSS Mode:                     None
   Contents:
     Metadata:                   No
     Volume Configuration:       No
     User Snapshots:             Yes
   Archive:
     Total Archive Operations:   0
     Total Successful Operations: 0
     Total Failed Operations:    0
     Total Saved Metadata:       0 B
     Total Saved Volume Config:  0 B
     Average Copy Rate:          0 b/s

  Snapshot Summary - dailySnap_0
  ------------------
   Snapshot Name:                dailySnap_0
   Snapshot Operation:           Create
   Result:                       Success
   Time Requested:               04/23/2012 01:27:06 -0400
   Time Created:                 04/23/2012 01:27:07 -0400
   Last Time Verified:
   Request:                      Create
   Snapshot State:               Sparse
   Snapshot Origin:              Manual
   Report Name:                  snap_daily_0_create_20120423012706913.rpt


  Included Shares
  ---------------
       Share Name:               equip (user data)
       Filer:
         Name:                   nas10
         CIFS Share:             equipment
```

```
        Volume:                 vol2
        Filer Snapshot:         acopia_1_201204230527_d9bdece8-9866-11d8-91e3-f48e42637d58_vol2

      Share Name:               leased (user data)
      Filer:
        Name:                   nas10
        CIFS Share:             for_lease
        Volume:                 vol1
        Filer Snapshot:         acopia_1_201204230527_d9bdece8-9866-11d8-91e3-f48e42637d58_vol1

      Share Name:               backlots (user data)
      Filer:
        Name:                   fs2
        CIFS Share:             backlot_records
        Volume:                 E:\
        Filer Snapshot:         {306db370-8318-4ad4-9d58-39d80b215e2a}
        Time Created:           04/23/2012 01:27:09 -0400


  Excluded Shares
  ---------------
      Share Name:               scanners
      Filer:
        Name:                   fs5
        CIFS Share:             xraysScanners
      Reason:                   Snapshots were not supported on this type of back-end filer.

juser@mgmt17:~$
```

Showing All ARX Snapshots

To see the high-level status of all ARX snapshots, use the show snapshots command:

```
show snapshots
```

This shows a separate table of snapshots for every ARX volume. The table contains one row per snapshot, with the name of the snapshot rule, the name of the particular ARX snapshot, the time the snapshot was created (or a status indicator for a failed or in-progress snapshot), and the source of the snapshot (either a “Schedule” from a snapshot rule or a “Manual” invocation). The table is empty for volumes without any snapshots. If a status indicator appears for the Created time (such as “Incomplete”), you can use the snapshot report (show reports type Snapshot) to localize the problem.

For example, this shows that the current switch has had four successful snapshots in the “medarcv~/lab_equipment” volume:

```
bstnA(gbl)# show snapshots

Namespace: medco
Volume:    /vol

Rule                Type           Name                    Created                  Source
------------------- -------------- ----------------------- ------------------------ --------

Namespace: wwmed
Volume:    /acct

Rule                Type           Name                    Created                  Source
------------------- -------------- ----------------------- ------------------------ --------

Namespace: medarcv
Volume:    /rcrds

Rule                Type           Name                    Created                  Source
------------------- -------------- ----------------------- ------------------------ --------

Namespace: medarcv
Volume:    /lab_equipment

Rule                Type           Name                    Created                  Source
------------------- -------------- ----------------------- ------------------------ --------
hourlySnap          Snapshot       hourlySnap_0            03/14/2012 00:48:57 -0400 Manual
hourlySnap          Snapshot       hourlySnap_1            03/14/2012 00:48:27 -0400 Manual
dailySnap           Snapshot       dailySnap_0             03/14/2012 00:46:59 -0400 Manual
dailySnap           Snapshot       dailySnap_1             03/14/2012 00:46:29 -0400 Manual
mirrorSnap          Replica        mirrorSnap_0            03/14/2012 00:52:40 -0400 Manual
mirrorSnap          Replica        mirrorSnap_1            03/14/2012 00:51:52 -0400 Manual

Namespace: medarcv
Volume:    /test_results

Rule                Type           Name                    Created                  Source
------------------- -------------- ----------------------- ------------------------ --------

Namespace: insur
Volume:    /claims
```

```
Rule                Type          Name                     Created                   Source
------------------- ------------- ------------------------ ------------------------- --------
mpHourlySnap        Snapshot      mpHourlySnap_0           03/14/2012 01:06:47 -0400 Manual
mpHourlySnap        Snapshot      mpHourlySnap_1           03/14/2012 01:04:33 -0400 Manual
bstnA(gbl)# ...
```

Focusing on a Single Namespace, Volume, or Rule

You can add some options to show a smaller set of snapshots:

show snapshots [*namespace* [*vol-path* [*rule*]]]

where

namespace (1-30 characters) selects a namespace,

vol-path (1-1024 characters) narrows the scope to a specific volume, and

rule (1-1024 characters) narrows the scope further, to a specific snapshot rule in the volume (for example, ‘dailySnap’).

If you select a particular rule, the output shows the rule’s configuration details. After the rule has run at least once, its output contains a “Snapshot Summary - *rule-name_n*” section for each retained snapshot. The top table in the Snapshot-summary section shows the time and overall state of the coordinated snapshot. The “Included Shares” table shows each back-end share with a component snapshot and the location of the snapshot on that share. The “Excluded Shares” section shows all shares in the volume that were administratively excluded from this coordinated snapshot, if there were any. The “Offline Shares” section, shows all shares that were unreachable at the time of the snapshot, if there were any. (The “Excluded Shares” and “Offline Shares” sections only appear if any such shares were omitted from the snapshot.)

For example, this command focuses on the “hourlySnap” rule, which has retained two snapshots so far:

```
bstnA(gbl)# show snapshots medarcv /lab_equipment hourlySnap


  Snapshot Rule Summary
  ---------------------
   Namespace Name:               medarcv
   Volume Name:                  /lab_equipment
   Snapshot Rule Name:           hourlySnap


  Snapshot Properties
  -------------------
   Snapshots Enabled:            Yes
   Guarantee Consistency:        Disabled
   Retain Count:                 3
   Schedule:                     hourly
   CIFS Directory Name:          ~snapshot
   Directory Display:            All Exports
   Hidden File Attribute:        Not Set
   Restricted Access Configured: Yes
   VSS Mode:                     None
```

```
 Contents:
   Metadata:                   No
   Volume Configuration:       No
   User Snapshots:             Yes
 Archive:
   Total Archive Operations:   0
   Total Successful Operations: 0
   Total Failed Operations:    0
   Total Saved Metadata:       0 B
   Total Saved Volume Config:  0 B
   Average Copy Rate:          0 b/s

Snapshot Summary - hourlySnap_0
-------------------
 Snapshot Name:                hourlySnap_0
 Time Requested:               03/14/2012 00:48:57 -0400
 Time Created:                 03/14/2012 00:48:57 -0400
 Last Time Verified:
 Request:                      Idle
 Snapshot State:               Complete
 Snapshot Origin:              Manual
 Report Name:                  snap_hourly_0_create_20120314004857655.rpt


Included Shares
----------------
     Share Name:               equip (user data)
     Filer:
       Name:                   nas10
       CIFS Share:             equipment
       Volume:                 vol2
       Filer Snapshot:         acopia_5_201203140448_d9bdece8-9866-11d8-91e3-f48e42637d58_vol2

     Share Name:               leased (user data)
     Filer:
       Name:                   nas10
       CIFS Share:             for_lease
       Volume:                 vol1
       Filer Snapshot:         acopia_5_201203140448_d9bdece8-9866-11d8-91e3-f48e42637d58_vol1

     Share Name:               backlots (user data)
     Filer:
       Name:                   fs2
       CIFS Share:             backlot_records
       Volume:                 E:\
       Filer Snapshot:         {8bd8cb9e-409f-4362-a710-b770c85c6cb9}
       Time Created:           03/14/2012 00:48:59 -0400


Excluded Shares
----------------
     Share Name:               scanners
     Filer:
       Name:                   fs5
       CIFS Share:             xraysScanners
     Reason:                   Snapshots were not supported on this type of back-end filer.

Snapshot Summary - hourlySnap_1
-------------------
 Snapshot Name:                hourlySnap_1
 Time Requested:               03/14/2012 00:48:27 -0400
 Time Created:                 03/14/2012 00:48:27 -0400
```

```
   Last Time Verified:
   Request:                      Idle
   Snapshot State:               Complete
   Snapshot Origin:              Manual
   Report Name:                  snap_hourly_0_create_20120314004827354.rpt


  Included Shares
  ---------------
       Share Name:               equip (user data)
       Filer:
         Name:                   nas10
         CIFS Share:             equipment
         Volume:                 vol2
         Filer Snapshot:         acopia_4_201203140448_d9bdece8-9866-11d8-91e3-f48e42637d58_vol2

       Share Name:               leased (user data)
       Filer:
         Name:                   nas10
         CIFS Share:             for_lease
         Volume:                 vol1
         Filer Snapshot:         acopia_4_201203140448_d9bdece8-9866-11d8-91e3-f48e42637d58_vol1

       Share Name:               backlots (user data)
       Filer:
         Name:                   fs2
         CIFS Share:             backlot_records
         Volume:                 E:\
         Filer Snapshot:         {e671c2e5-9217-4b62-903f-78b39e012c96}
         Time Created:           03/14/2012 00:48:28 -0400


  Excluded Shares
  ---------------
       Share Name:               scanners
       Filer:
         Name:                   fs5
         CIFS Share:             xraysScanners
       Reason:                   Snapshots were not supported on this type of back-end filer.
bstnA(gbl)# ...
```

Preparing for Snapshot Reconstitution

There may be situations where back-end snapshots become disassociated with their ARX snapshots. For example, someone could accidentally remove a snapshot rule and want to re-incorporate the back-end snapshots into a new instance of the rule. As another example, a primary site with an ARX may fail, and a disaster-recovery site with another ARX would need to incorporate snapshots in the same way as was done at the primary site. Re-associating filer snapshots with ARX snapshots is called *snapshot reconstitution*.

Snapshot reconstitution is implemented with snapshot reports and a special Perl script that is provided with ARX software. Copy all of these to an external device to prepare for reconstitution at a later date. The snapshot reports contain the names and snapshot times for the back-end snapshots; these names and times can change with each ARX snapshot, so they should be copied after each snapshot operation. You can use the at command, together with some form of the copy command, to regularly send copies of your snapshot reports to an external host. The *ARX® CLI Reference* describes both of these commands in detail.

Copy the reports so that they are converted to XML format; use the format xml option at the end of the copy command to accomplish this.

Choose a destination host that supports Perl scripts, and that has the XML::Simple Perl module installed. You can download the Perl module from the CPAN site:

http://search.cpan.org

For an example of the ARX-CLI syntax, this command sequence sends all reports that begin with “snap” to a host at 172.16.100.183:

```
bstnA(gbl)# end
bstnA# config
bstnA(cfg)# at 17:32:51 every 1 day do "copy reports snap*
ftp://ftpuser:*@172.16.100.183//var/arxSnapRpts/ format xml"
bstnA(cfg)# ...
```

You also require a copy of the snap-recon.pl script, provided in the “software” directory on the ARX. To continue the example, this copies the script to the same host and directory:

```
bstnA(cfg)# end
bstnA# copy software snap-recon.pl ftp://ftpuser:ftpuser@172.16.100.183//var/arxSnapRpts/
bstnA# ...
```

Maintaining the Repository for Snapshot Reports

The above commands add progressively more reports to the above share, so you may need to perform occasional maintenance. All snapshot reports have time stamps in their names, so that they are unique; new reports do not overwrite the previous versions. The reports are small (roughly 3 Kbytes), but the share may accumulate very large numbers of them in the fullness of

time. The snap-recon.pl script only requires the most-recent set of reports. We recommend a periodic script or maintenance process to regularly remove the unnecessary reports.

Reconstituting ARX Snapshots

To reconstitute an ARX snapshot from your snapshot reports, you create a CLI script, copy it to the ARX, and run it in the ARX CLI. This process begins on the filer share with the reports, where you build the CLI script by running the snap-recon.pl script:

```
snap-recon.pl --report-dir path
  [--namespace ns [--volume vol-path [--rule rule-name
    [--target-ns t-ns --target-vol t-vol --target-rule t-rule]]]]
  [--output-script script-name] [--report-prefix prefix] [--verbose]
```

where

path is the path to the snapshot reports. This can be a relative path (such as "../arxReports") or an absolute path (such as "/var/arxReports").

ns (optional) focuses on the snapshots in the given ARX namespace. From the ARX CLI, you can use show namespace for a complete list of namespaces, volumes, and snapshot rules.

vol-path (optional if you entered a namespace) focuses on the snapshots in the given ARX volume. Specify the volume by its path name, starting with a slash (/); for example, "/rcrds" or "/acct."

rule-name (optional if you entered a volume) takes only the snapshots for a given ARX rule.

--target-ns ***t-ns*** **--target-vol** ***t-vol*** **--target-rule** ***t-rule*** (optional if you entered a rule) edits the final output:

- ***t-ns*** replaces all instances of the **--namespace** name with this name. Use this if you want to incorporate the filer snapshots in a new namespace.
- ***t-vol*** replaces all instances of the **--volume** path with the path you enter here.
- ***t-rule*** replaces the rule name selected with the **--rule** option.

script-name (optional) sets a name for the output script. By default, the output script is named "snapRecon.cli."

prefix (optional) changes the report prefixes in the output script. These are the prefixes used in the output script's snapshot manage commands.

verbose (optional) adds the verbose option to all of the output script's snapshot manage commands.

The output of this script is an ARX-CLI script, named "snapRecon.cli" by default.

For example, this command sequence runs the snap-recon.pl test from a Unix host and lists the files in the current directory. The new "snapRecon.cli" file is shown in bold:

```
juser@lnx2:/var/arxSnapRpts$ ls
snap_daily_0_create_20090408013001026.xml  snap_hourly_0_create_20090408013234784.xml
snap_daily_0_create_20090408013036393.xml  snap_hourly_0_create_20090408013305227.xml
snap_daily_0_create_20090408013106820.xml  snap-recon.pl
snap_daily_2_remove_20090408013154370.xml
juser@lnx2:/var/arxSnapRpts$ ./snap-recon.pl --report-dir .
juser@lnx2:/var/arxSnapRpts$ ls
snap_daily_0_create_20090408013001026.xml  snap_hourly_0_create_20090408013234784.xml
snap_daily_0_create_20090408013036393.xml  snap_hourly_0_create_20090408013305227.xml
snap_daily_0_create_20090408013106820.xml  snapRecon.cli
snap_daily_2_remove_20090408013154370.xml  snap-recon.pl
juser@dlnx2:/var/arxSnapRpts$ ...
```

Viewing the ARX-CLI Script

Here is a sample of the CLI file, a commented list of snapshot manage commands that re-incorporate the back-end snapshots into their ARX snapshots:

```
juser@dlnx2:/var/arxSnapRpts$ cat snapRecon.cli
;
;Snapshot Reconstruction Commands
;
;F5 Networks, Inc.
;Copyright (c) 2009
;All Rights Reserved
;


;Report: snap_daily_0_create_20090408013036393.rpt
;Created: 04/08/2009:01:30:36
;Namespace: medarcv
;Volume: /lab_equipment
;Rule: dailySnap
snapshot manage medarcv /lab_equipment equip dailySnap
acopia_2_200904080530_3a5700b6-6bd6-11d8-85d5-a96dcef11607_vol2 created-on 04/08/2009:01:30:36
snapshot manage medarcv /lab_equipment backlots dailySnap {b0263932-0ebf-49ed-ae92-f2f8eaadae64}
created-on 04/08/2009:01:30:36
snapshot manage medarcv /lab_equipment leased dailySnap
acopia_2_200904080530_3a5700b6-6bd6-11d8-85d5-a96dcef11607_vol2 created-on 04/08/2009:01:30:36

;Report: snap_daily_0_create_20090408013106820.rpt
;Created: 04/08/2009:01:31:06
;Namespace: medarcv
;Volume: /lab_equipment
;Rule: dailySnap
snapshot manage medarcv /lab_equipment equip dailySnap
acopia_3_200904080531_3a5700b6-6bd6-11d8-85d5-a96dcef11607_vol2 created-on 04/08/2009:01:31:06
snapshot manage medarcv /lab_equipment backlots dailySnap {0d37ab58-7674-456a-89ac-75af2b1047c2}
created-on 04/08/2009:01:31:06
snapshot manage medarcv /lab_equipment leased dailySnap
acopia_3_200904080531_3a5700b6-6bd6-11d8-85d5-a96dcef11607_vol2 created-on 04/08/2009:01:31:06

;Report: snap_hourly_0_create_20090408013234784.rpt
;Created: 04/08/2009:01:32:34
;Namespace: medarcv
;Volume: /lab_equipment
;Rule: hourlySnap
```

```
snapshot manage medarcv /lab_equipment equip hourlySnap
acopia_4_200904080532_3a5700b6-6bd6-11d8-85d5-a96dcef11607_vol2 created-on 04/08/2009:01:32:34
snapshot manage medarcv /lab_equipment backlots hourlySnap {f351ebf6-2167-4368-92fa-6b83b90079ce}
created-on 04/08/2009:01:32:34
snapshot manage medarcv /lab_equipment leased hourlySnap
acopia_4_200904080532_3a5700b6-6bd6-11d8-85d5-a96dcef11607_vol2 created-on 04/08/2009:01:32:34

;Report: snap_hourly_0_create_20090408013305227.rpt
;Created: 04/08/2009:01:33:05
;Namespace: medarcv
;Volume: /lab_equipment
;Rule: hourlySnap
snapshot manage medarcv /lab_equipment equip hourlySnap
acopia_5_200904080533_3a5700b6-6bd6-11d8-85d5-a96dcef11607_vol2 created-on 04/08/2009:01:33:05
snapshot manage medarcv /lab_equipment backlots hourlySnap {228cc6a0-0577-4afa-95c2-345020d1311f}
created-on 04/08/2009:01:33:05
snapshot manage medarcv /lab_equipment leased hourlySnap
acopia_5_200904080533_3a5700b6-6bd6-11d8-85d5-a96dcef11607_vol2 created-on 04/08/2009:01:33:05

;
;END Snapshot Reconstruction Commands
;
juser@dlnx2:/var/arxSnapRpts$ ...
```

Running the ARX-CLI Script

After you review the CLI script, you copy it to the ARX and run it. From the ARX CLI, use the copy command to download the script from your external host to the ARX's "scripts" directory. For example, this uses FTP to download the snapRecon.cli script that we created above:

```
bstnA(cfg)# end
bstnA# copy ftp://ftpuser:ftpuser@172.16.100.183//var/arxSnapRpts/snapRecon.cli scripts
snapRecon.cli
bstnA# ...
```

Once it is copied to the "scripts" directory, use the priv-exec run command to run it and reconstitute your snapshots. The *ARX® CLI Reference* describes the run command in detail. For example, this command runs the above script:

```
bstnA# run scripts snapRecon.cli
bstnA# ...
```

Correcting Any Snapshot Mismatches

A large CLI script may contain errors that mismatch one or more back-end snapshots with their counterparts on the ARX. To correct this issue, you can use the snapshot clear command to remove the ARX records for one or more back-end snapshots, then you can edit and re-run the ARX-CLI script. For details on the snapshot clear command, recall *Clearing the Snapshot from the ARX Configuration, on page 2-20*.

Replica Snapshot Shares

Replica snapshot shares (referred to from here on as “replica-snap shares”) enable you to create snapshots for data that has been replicated from data on tier-1 filers. Since the replica snapshots do not occupy storage in the managed volume on the tier-1 filer, a larger number of snapshots can be retained, and for longer periods of time, than would be desirable on a tier-1 filer.

Replica-snap shares are supported on Data Domain, EMC, NetApp, and Windows filers. Replica-snap shares are presented via managed volumes, and are not supported for direct volumes. They are managed using policy rules, schedules, and snapshot commands.

Define a replica-snap share by setting the replica-snap attribute for a share and configuring a rule to populate it with snapshot data.

Replica-Snap Share Configuration

Create a replica-snap share by setting the replica-snap attribute for an existing share. This attribute can be set only if the share is not enabled.

In gbl-ns-vol-shr mode, the command syntax to set the attribute for the share is:

```
replica-snap
```

For example:

```
bstnA(gbl-ns-vol-shr[wwmed~/acct~budget])# replica-snap
```

Executing the no replica-snap command removes the replica-snap attribute, disabling the share’s capacity for containing replica snapshots.

Replica-Snap Rule Configuration

Configure a rule for populating a replica-snap share using the snapshot replica-snap-rule CLI command in gbl-ns-vol mode. This command opens gbl-ns-vol-replica-snap mode, which provides commands for defining the rule.

The command syntax is:

```
snapshot replica-snap-rule name
```

where *name* identifies the rule that you are defining (up to 1024 characters).

For example, the following command creates a replica snapshot rule named “acctsnaps” and opens gbl-ns-vol-replica-snap mode:

```
bstnA(gbl-ns-vol[wwmed~/acct])# snapshot replica-snap-rule acctsnaps
```

The “no” form of the command, no snapshot replica-snap-rule *name*, disables the specified snapshot rule.

Replica Snapshot Mode

The gbl-ns-vol-replica-snap CLI command mode provides the following commands for defining a replica-snap rule:

- enable: This activates the current replica-snap rule. The command syntax is:

  ```
  enable
  ```

  For example:

  ```
  bstnA(gbl-ns-vol-replica-snap[wwmed~/acct~acctsnaps])# enable
  bstnA(gbl-ns-vol-replica-snap[wwmed~/acct~acctsnaps])#
  ```

  Executing the no enable command de-activates the replica-snap rule.

- exclude: Specify a share to exclude from the snapshots created by this rule. The command syntax is:

  ```
  exclude sharename
  ```

  where *sharename* identifies a share in the current volume.

  For example, the following command excludes a share named “bills”:

  ```
  bstnA(gbl-ns-vol-replica-snap[wwmed~/acct~acctsnaps])# exclude
  bills
  bstnA(gbl-ns-vol-replica-snap[wwmed~/acct~acctsnaps])#
  ```

  Executing the no exclude *sharename* negates the exclusion of the specified share, enabling it to be included in replica-snap shares.

- report: Enable reporting for a specified report prefix for the current replica-snap rule. The command syntax is:

  ```
  report reportprefix
  ```

  where *reportprefix* is a prefix to include in the names of the resulting reports to help identify them. The optional error-only argument causes a report to be generated only if an error occurs during the operation of the snapshot.

  For example, the following command enables the generation of reports with a prefix of “snapShareError_” when an error occurs during snapshot creation:

  ```
  bstnA(gbl-ns-vol-replica-snap[wwmed~/acct~acctsnaps])# report
  snapShareError_ error-only
  bstnA(gbl-ns-vol-replica-snap[wwmed~/acct~acctsnaps])#
  ```

  Executing the no report command disables reporting for the current replica-snap rule.

- retain: Specify the number of snapshots to retain for a the current replica-snap rule.

  The command syntax is:

  ```
  retain number
  ```

  where *number* is the maximum number of snapshots to keep on the share, up to a maximum of 1024.

  For example, the following command causes 30 snapshots to be kept:

```
bstnA(gbl-ns-vol-replica-snap[wwmed~/acct~acctsnaps])# retain 30
bstnA(gbl-ns-vol-replica-snap[wwmed~/acct~acctsnaps])#
```

Executing no retain reverts the retention of replica snapshots to the default.

- schedule: Specify an existing schedule to associate with the current replica-snap rule.

  The command syntax is:

  ```
  schedule name
  ```

  where *name* identifies a previously-defined schedule.

  For example, the following command associates the schedule named "daily" with the current replica-snap rule:

  ```
  bstnA(gbl-ns-vol-replica-snap[wwmed~/acct~acctsnaps])# schedule
  daily
  bstnA(gbl-ns-vol-replica-snap[wwmed~/acct~acctsnaps])#
  ```

  Executing no schedule dissociates the current replica-snap rule from any schedule.

Displaying Replica Snapshot Information

Execute the show snapshots command to see the current status of one or more replica-snap rules.

Behavioral Considerations For Replica Snapshot Shares

Replica-snap shares exhibit a number of different characteristics which should be kept in mind during their use.

General Considerations

The following considerations pertain generally to replica-snap shares in all circumstances:

- Replica-snap shares are counted toward the ARX's 64 share per volume limit. However, the free space on the replica-snap shares is not counted toward the volume's total free space.
- At least one managed share must be configured and enabled in a volume before that volume can be used.
- From a policy standpoint, a managed volume is considered online once all the managed shares in the volume have completed import, are enabled, and are online. Replica-snap shares are not considered when determining this status, regardless of whether they are online or offline, enabled or disabled.
- Replica-snap shares are marked as such in all reports containing share summaries.

- Replica-snap shares can be removed from the configuration using the no share command in global-namespace-volume mode. For replica-snap shares, the no share command does not require or accept the relocate-dirs, force, or remove-files-entries options. This is the same as executing a no share command on a direct share. A no share command can be executed on a replica-snap share whether it is enabled or not.
- The remove share migrate and remove share no-migrate commands work if given a replica-snap share as the share being removed. They ignore any specified relocate and force options, skip the migration step, and implement the same functionality as the no share command.
- When a replica-snap share is removed from a volume, a shareRemoveComplete trap is sent to provide notification of the removal of the share, and to clear any outstanding notifications in the show health display.
- Some filer-replication applications are volume-level, and copy the entire contents of the back-end share to the replica-snap share, including the filer snapshots. This would overwrite any snapshots that the ARX takes at the replica-snap share, and defeat the snapshot replica-snap-rule. Do not use volume-level replication to copy the source share's contents to the replica-snap share.

CIFS Considerations For Replica-Snap Shares

The following considerations pertain to replica-snap shares in CIFS environments:

- Replica-snap shares are accessed using the same authorization protocols specified for the namespace in which they reside.
- To be used with constrained delegation, all filers hosting replica-snap shares need to be in the same domain as the VIP presenting the containing volume.
- When replica-snap shares are used with constrained delegation, the same domain controller configuration must exist for the replica-snap filers as for filers containing the managed shares.

Subshares

The following considerations pertain to replica-snap shares and subshares:

- Subshares are replicated to the replica-snap share during the snapshot process. If some subshares cannot be replicated to the replica-snap shares, an entry is added to the subshare report noting the failure, but the operation is still considered successful, and the snapshot still executes.
- In order for snapshots that reside on replica-snap shares to be available via subshares, subshares that exist on the volume being replicated must be replicated to the replica-snap shares. Since the target file system is likely to be read-only, the ARX cannot create directories on the replica-snap shares, only the subshares themselves. The ARX relies on the replication process to create the directories on the replica-snap shares that are pointed to by the subshares.

- If a subshare is created via the export command while a volume containing replica-snap shares is enabled, the subshare subsystem attempts to create the subshare on all shares, including the replica-snap share. If the directory does not exist, or the creation fails on any of the replica-snap share, the failure is noted in the associated subshares report, but the operation is still considered a success. This provides access to the snapshot data as soon as possible if the target directory has already been replicated to the replica-snap share. Since subshares are synchronized at each snapshot attempt, any failed subshares synchronizations are resolved during later snapshot attempts.
- The promote subshares and sync subshares functionality also ignores failures on replica-snap shares, and returns success if the failures only occur on replica-snap shares.

Access Based Enumeration (ABE)

The following considerations pertain to replica-snap shares and ABE:

- ABE settings are not captured in snapshots, so any updates to ABE settings on the shares affects the display of the contents of snapshots taken when different ABE settings were in effect.
- The ABE setting on the volume is synchronized to the replica-snap shares as part of the share enable process, the same as with managed shares.
- The cifs access-based-enum <ns> <vol> [force] command also synchronizes the ABE setting to all replica-snap shares in the volume.

Attribute Replication

The following considerations pertain to replica-snap shares and attribute replication:

- Depending on the particular replication mechanism used to replicate the data from the managed shares to the replica-snap shares, security descriptors and other attribute settings may not be preserved on the replica-snap share.

Policy Considerations

The following considerations pertain to replica-snap shares and policy:

- Replica snapshots are controlled by their own snapshot rules, known as “snapshot replica-snap rules.” Snapshot replica-snap rules can be managed using the same set of commands that pertain to ordinary snapshots.
- The maximum retention count for virtual snapshots is 1024.
- A separate type of snapshot rule supports taking snapshots of replica-snap shares. This is the “snapshot replica-snap-rule”. All handling and processing of the snapshots is done in the same manner as for other snapshot rules. Management of the snapshots is performed

using the regular snapshot commands, and a snapshot replica-snap rule can be specified as an argument to these commands, as well as the regular snapshot rule they accept.

- Snapshots on shares excluded from the rule, after snapshots have been created on that share, stay visible in the snapshot presentation until the virtual snapshot retention count is hit, and those snapshots are removed as part of removing the associated virtual snapshots. If the snapshot limit is hit on an individual filer, the creation of the snapshot on that filer fails, and is indicated in the snapshot report.
- Rule names for regular snapshot rules and snapshot replica-snap rules cannot overlap. This is necessary to make snapshot reconstitution work.
- Since snapshot replica-snap rules are targeting shares that are not being changed by the user, it is not necessary to fence the VIP when creating snapshots on replica-snap shares. It is desirable to have the schedules for regular snapshots and replica-snap rules execute at different times to keep the fence up for as little time as possible.
- All replica-snap shares must be able to take and present snapshots, so sparse snapshot support is not an issue for replica-snap shares. Replica-snap shares can be used in a managed volume with managed shares that do not support snapshots.
- Shadow replication ignores snapshot only-shares in the shadow source. None of the data on the replica-snap shares in the source is replicated to the target shadow volumes.
- Share-farms reject replica-snap shares as arguments to the “share” command. Place rules reject replica-snap shares are arguments to the “source” and “dest” commands.

Snapshot Rules

The following considerations pertain to replica-snap shares and snapshot rules:

- Replica-snap shares are not included in regular snapshots, and therefore do not need to be excluded explicitly from those snapshot rules.

Snapshot replica-snap rule

The following considerations pertain to replica-snap shares and replica-snap rules:

- The snapshot replica-snap rule causes the automatic skipping of managed shares during the snapshot process, in the same manner that the snapshot rule skips the replica-snap shares. You can enter exclude commands for replica-snap shares in the volume if they wish to put only a subset of the replica-snap shares into a snapshot replica-snap rule.
- For snapshot replica-snap rules, it is only the user data that ever has a snapshot taken of it. Also, an archive location cannot be specified in a snapshot replica-snap rule, as these shares are not used as part of file tracking, as they have no associated metadata.

Snapshot Reconstitution

The following considerations pertain to replica-snap shares and snapshot reconstitution:

- Snapshot reconstitution is done via the same mechanism used for snapshot rules. Each snapshot must be manually programmed into the ARX via the snapshot manage command. Snapshot reports for replica-snap shares need to be archived by the user, in the same manner as reports for snapshots on managed shares. This is necessary for the snapshot reconstitution script to work correctly.

Snapshot Grouping

The following considerations pertain to replica-snap shares and snapshot grouping:

- When taking snapshots of replica-snap shares, it is not necessary to raise the VIP fence.
- The snapshot grouping code determines when a group of snapshots is only done on replica-snap shares, and does not raise the VIP fence in that case, regardless of the fence setting in the volume. If replica-snap share snapshots are grouped with snaps on managed shares, the fence is raised according to the volume settings.

Subshare Synchronization

The following considerations pertain to replica-snap shares and subshare synchronization:

- The snapshot daemon synchronizes the subshares on replica-snap shares before executing the snapshots. If any of the replica-snap shares fail synchronization, this is noted in the subshare synchronization report, but it is not considered a failure. Even if synchronization of the subshares to the replica-snap shares fails, the snapshots of the replica-snap shares are still taken, but access to snapshots on the replica-snap shares is blocked for the failed subshare.
- A report is generated by the subshares synchronization, and the name of the subshares synchronization report is included in the snapshot report.

File History Archive

The following considerations pertain to replica-snap shares and file history archives:

- Replica-snap shares are not included in the XML configuration output created for the file-history archive functionality.

3

Tracking Files on Your Back-End Storage

Overview

This chapter only applies to managed volumes. Skip this chapter if your site only supports direct volumes.

File tracking is the process of periodically archiving the back-end locations of your files so that you can learn where any file was stored on any given day. Some sites back up and restore their files directly from their back-end filers, behind the ARX and its managed volumes. These sites use a data-protection device, called a *backup server* in this manual, to coordinate NDMP backups and restores between filers and backup tapes. If a managed volume migrates files since the most-recent backup operation, you may not know the filer where the files were backed up. You can use file tracking to keep records of files as they migrate, and to identify the filer that held any given file at any given time.

File tracking uses periodic ARX snapshots of a volume's metadata and configuration. A volume's metadata filer must therefore support ARX snapshots before the volume can support file tracking.

Before You Begin

The managed volume must have its metadata share on a back-end filer that supports snapshots or checkpoints. The Release Notes show the specific filers and filer releases supported by ARX snapshots. This section summarizes the configuration necessary to prepare your metadata share and its back-end filer.

NetApp Configuration

To support ARX snapshots, each NetApp volume must have the nosnapdir option turned off. From the NetApp CLI, use the vol options command on each NetApp volume behind the ARX:

```
vol options vol-name nosnapdir off
```

where ***vol-name*** identifies the NetApp volume.

For example, the following commands access a NetApp filer named "nas1" through SSH, and then permit snapshots in one volume:

```
juser@mgmt17:~$ ssh root@192.168.25.21
root@192.168.25.21's password: password

nas1*> vol options vol1 nosnapdir off
nas1*> ...
```

EMC Celerra Best Practices

For each EMC management station behind your collective ARX volumes, we recommend that you use no more than two EMC volumes (called "file systems" in EMC documentation). For example, if some of your ARX volumes use client shares and/or metadata shares from the EMC management station at 1.2.3.4, those ARX shares should draw from two EMC file systems at most. EMC-Celerra checkpoints typically take 5-10 seconds per file system when the Celerra is under moderate load, where NetApp snapshots rarely take longer than 5 seconds. Further, the EMC management station serializes its checkpoints; if a given management station needs a checkpoint from file-server A and file-server B at the same time, the file-server-A checkpoint must finish before the file-server B checkpoint can begin.

Two or more ARX-snapshot rules with the same schedule perform *snapshot grouping*, described in *Snapshot Grouping*, on page 2-10. That is, the ARX determines the fewest-possible back-end snapshots that it can take to satisfy all of the concurrent rules. This improves snapshot performance. However, any single back-end snapshot may end up in a larger group, and may have to wait for the EMC serialization described in this section.

Creating a Proxy User for Managing the Filer

To invoke snapshots at the volume's metadata filer, the ARX volume requires administrative privileges at the filer's CLI. You enter the proper administrative username and password as a proxy user. If the metadata share is hosted by a NetApp or EMC device, these are UNIX credentials for RSH or SSH logins; they do not require a Windows domain. If the metadata share is hosted by a Windows Server, the volume uses WinRM to access it: in this case, the proxy user must have an FQDN to facilitate Kerberos authentication.

For example, this command sequence creates a proxy user named "nas_admin" for SSH or RSH logins:

```
bstnA(gbl)# proxy-user nas_admin
bstnA(gbl-proxy-user[nas_admin])# user root
Password: rootpasswd
Validate Password: rootpasswd
bstnA(gbl-proxy-user[nas_admin])# exit
bstnA(gbl)# ...
```

For details on these proxy-user commands and others, see *Adding a Proxy User*, on page 3-4 of the *ARX® CLI Storage-Management Guide*.

Preparing an External-Filer Configuration

The ARX logs into the CLI at the metadata share's filer, and requires parameters for doing so. These parameters include the type of filer (identified by vendor) and assignment of the proxy-user credentials from

above. The commands for setting these parameters are described in *Preparing the Filer for ARX-Snapshot Support*, on page 6-12 of the *ARX® CLI Storage-Management Guide*.

For example, the following command sequence sets up snapshot access for the "nas1" filer:

```
bstnA(gbl)# external-filer nas1
bstnA(gbl-filer[nas1])# filer-type network-appliance
bstnA(gbl-filer[nas1])# proxy-user nas_admin
bstnA(gbl-filer[nas1])# manage snapshots
bstnA(gbl-filer[nas1])# exit
bstnA(gbl)# ...
```

Preparing External Filers Behind the Volume's Shares

The ARX can also use CLI access for all of the filers behind the managed volume, the filers that store client data. The managed-volume software uses this access to find the actual file-system path of the filer's shares, as opposed to a CIFS-share name or a virtual NFS-export path that may be used in the ARX-volume configuration. Standard backup servers use only file-system paths to identify client files and directories. Therefore, file-tracking software finds and records real file-system paths. You can use the actual file-system paths to retrieve backed-up files from your backup servers.

For external filers where the ARX tracks files, use the filer-type command and the proxy-user command. It is unnecessary to use the manage snapshots command to support this path-retrieval process:

```
bstnA(gbl)# external-filer fs2
bstnA(gbl-filer[fs2])# filer-type windows
bstnA(gbl-filer[fs2])# proxy-user cifs_admin
bstnA(gbl-filer[fs2])# exit
bstnA(gbl)# ...
```

Best Effort for Path Retrieval

The above instructions are a best practice, not a requirement. They ensure that the file-tracking reports have an actual file-system path for files on these shares. Without CLI access, the file-tracking software uses a "best guess" for paths, based on the following:

- For CIFS shares, the file-tracking software sends a query to its storage filer(s), and the query requires Administrator privileges. Specifically, the proxy user assigned to the namespace must belong to the Administrators group. Without adequate privileges, the path is blank. For details on assigning a proxy user to the namespace, see *Choosing a CIFS Proxy User*, on page 7-15 of the *ARX® CLI Storage-Management Guide*.
- For NFS shares, each path includes the back-end-export path used in the gbl-ns-shr filer command. This may be a virtual path on the filer, not an actual part of the physical file system, so you may need to translate it to use it with your backup server. For information on the filer command, see *Identifying the Filer and Share*, on page 9-34 of the *ARX® CLI Storage-Management Guide*.

Creating a File-History Archive

To track files, an ARX volume periodically records its current state in an archive. The volume's current state is defined here as

- a copy of the volume's configuration, and (typically)
- a snapshot of the volume's metadata from the same time.

You can search this archive later with various file-tracking queries. The archive resides on a filer or a managed volume, which requires sufficient storage space for up to seven years of these file-history records. The filer or volume that stores these records does not require snapshot or checkpoint capabilities. It only keeps a copy of the snapshot/configuration records.

From gbl mode, use the file-history archive command to start the process of configuring one:

file-history archive *name*

where ***name*** (1-64 characters) is a name you choose for the archive configuration.

The CLI prompts for confirmation before creating a new archive object; enter **yes** to continue. This puts you into gbl-archive mode, where you choose a filer or managed volume to hold the archive records.

You can create up to 24 file-history archives on an ARX.

For example, the following command sequence creates an empty archive object, "fileRecordsMed:"

```
bstnA(gbl)# file-history archive fileRecordsMed

This will create a new archive.

Create archive 'fileRecordsMed'? [yes/no] yes
bstnA(gbl-archive[fileRecordsMed])# ...
```

Choosing a Location for the Archive Data

The most-important step in creating a file-history archive is choosing a storage location. As mentioned above, the archive can hold up to 7 years of records. Choose a filer with a large storage capacity. (You can alternatively choose a volume on this ARX as an archive location; skip to the subsection below if this is preferable.)

From gbl-archive mode, use the location filer command to choose an external filer where the archive will reside. The following syntax selects an NFS export to store the archive:

location filer *ext-filer* {nfs3tcp | nfs3} *share-name* [path *path*]

where

ext-filer (1-64 characters) identifies the filer where the archive will reside. This is the external-filer name for the filer; use show external-filer for a complete list of configured filers on the ARX.

nfs3tcp | nfs3 are a required choice. The **nfs3tcp** option is NFSv3 over TCP, and the **nfs3** option is NFSv3 over UDP.

share-name (1-64 characters) is the name of the back-end NFS export where the archive will reside.

path *path* (optional, 1-1024 characters) chooses a subdirectory to store the archive.

This syntax chooses a CIFS share for archive storage:

```
location filer ext-filer cifs share-name proxy-user proxy [path path]
```

where

ext-filer (1-64 characters) identifies a CIFS-supporting filer.

cifs is a required keyword.

share-name (1-64 characters) is the name of the back-end CIFS share where the archive will reside.

proxy (1-32 characters) is the name of a proxy-user configuration (see *Adding a Proxy User*, on page 3-4 of the *ARX® CLI Storage-Management Guide*). The CLI uses the proxy-user credentials to store archive files on the *share-name* share. These credentials require read and write permission at the share; they do not require Backup Operator or Administrator privileges. Use show proxy-user for a full list of all proxy users.

path *path* (optional, 1-1024 characters) chooses a subdirectory to store the archive.

For example, the following command sequence stores the "fileRecordsMed" archive on a CIFS share:

```
bstnA(gbl)# file-history archive fileRecordsMed
bstnA(gbl-archive[fileRecordsMed])# location filer fs4 cifs arx_file_archv proxy-user acoProxy2
bstnA(gbl-archive[fileRecordsMed])# ...
```

Storing the Archive in an ARX Volume

You can store the file-history archive in an ARX volume instead of a particular external filer. If you select a managed volume, the archive can be distributed among the its back-end filers according to your migration policies. This can provide an efficient use of space for your expanding archive.

◆ Note

If you export this archive to clients who do not understand its purpose, the clients may inadvertently delete important records. We recommend using an ARX volume or directory that is accessible to management personnel only. Management access may not even be necessary except on rare occasions. For instructions on exporting volume storage to clients, see **Sharing a Namespace Volume***, on page 11-15 and* **Exporting a Namespace Volume***, on page 11-4 of the* **ARX® CLI Storage-Management Guide**.

Use the location namespace command to store the archive on one of the ARX's volumes. The file-access protocol(s) and the proxy user are part of the namespace configuration, so they are not required in this form of the command:

location namespace *name* volume *vol-path* [path *path*]

where

name (1-30 characters) is a namespace on the current ARX. Use show namespace for a list of all available namespaces. If you choose a multi-protocol (NFS and CIFS) namespace, the archive uses NFS to store its records.

vol-path (1-1024 characters) selects a volume from the above namespace. Type a **?** character for a list of the namespace's volumes.

path *path* (optional, 1-1024 characters) chooses a subdirectory to store the archive.

For example, the following command sequence stores the "testFA" archive on subdirectory of the "wwmed~/acct" volume:

```
bstnA(gbl)# file-history archive testFA
bstnA(gbl-archive[testFA])# location namespace wwmed volume /acct path /fileArchs
bstnA(gbl-archive[testFA])# ...
```

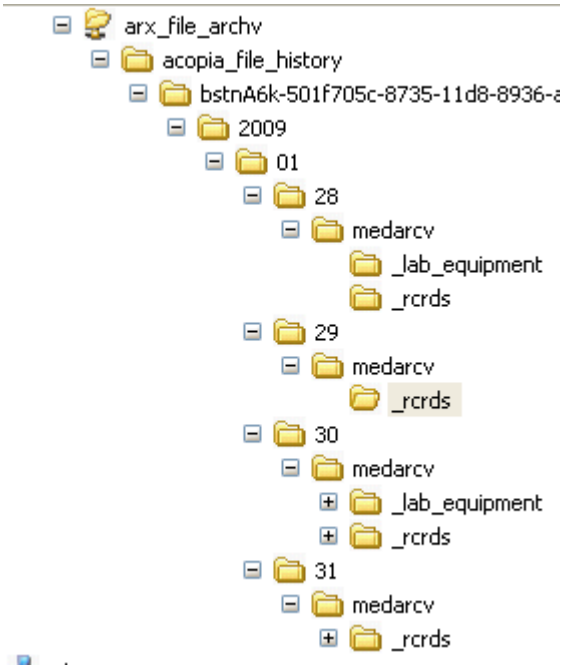
Archive File Structure on the Location Filer

The ARX stores its file-history records with the following directory structure:

```
/acopia_file_history
    /ARX-name-ARX-GUID
        /yyyy
            /mm
                /dd
                    /namespace-name
                        /volume-name
                    /namespace-name
                        /volume-name
                /dd
                ...
            /mm
            ...
        /yyyy
        ...
```

This structure starts at the **path** given by the location command, described above. If there is no path, this starts in the root of the ARX volume or filer share.

For example, the following structure contains 4 days of file records, for January 28 through January 31:



Moving the Location

As time goes on, a location filer may run low on disk space. For this situation, you can move the location to a larger filer or to a managed volume. You begin by using the no location command, which stops the ARX from writing to the current archive location:

```
no location
```

Then manually copy or move the “acopia_file_history” folder to the destination filer or volume. To complete the migration, run the location command with the new filer or volume location.

For example, this command sequence removes the location for the “FHinsur” archive:

```
bstnA(gbl)# file-history archive FHinsur
bstnA(gbl-archive[FHinsur])# no location
bstnA(gbl-archive[FHinsur])# ...
```

To continue the example, assume that the administrator logs onto a management station and moves the “acopia_file_history” directory from the current archive filer to a CIFS share on the ARX. Suppose that the ARX share is backed by a new managed volume on the ARX, “medarcv~/fhStore.” This command moves the location to the root of that volume:

```
bstnA(gbl)# file-history archive FHinsur
bstnA(gbl-archive[FHinsur])# location namespace medarcv volume /fhStore
bstnA(gbl-archive[FHinsur])# ...
```

Setting a Description for the Archive

You can add a description to the archive for use in some of the show commands described below. The description can differentiate the archive from others. From gbl-archive mode, use the description command to add a description string:

description *text*

where ***text*** is 1-255 characters. Quote the text if it contains any spaces.

For example:

```
bstnA(gbl)# file-history archive fileRecordsMed
bstnA(gbl-archive[fileRecordsMed])# description “archive share for ARX file histories”
bstnA(gbl-archive[fileRecordsMed])# ...
```

Removing the Description

From gbl-archive mode, use no description to remove the description string from the current file-history archive:

no description

For example:

```
bstnA(gbl)# file-history archive testFA
bstnA(gbl-archive[testFA])# no description
bstnA(gbl-archive[testFA])# ...
```

Listing All File-History Archives

Use the show file-history archive command to display a list of all the archives on the current ARX. You can enter this command from any CLI mode:

show file-history archive

The output is a table with one row per archive. For example, this shows a single archive:

```
bstnA(gbl)# show file-history archive

Archive Name          Description
--------------------  ---------------------------------------
fileRecordsMed        archive share for ARX file histories
bstnA(gbl)# ...
```

Showing One File-History Archive, with Details

For a detailed view of a single file-history archive, identify the archive at the end of the command:

show file-history archive *name*

where ***name*** (1-64 characters) identifies the desired archive.

The output contains some configuration details followed by a table of the managed volumes that use this archive. (Later sections explain how to configure a managed volume to store its file-history in an archive). For example, this command shows details for the “fileRecordsMed” archive. No managed volumes use this archive yet:

```
bstnA(gbl)# show file-history archive fileRecordsMed

Name:           fileRecordsMed
Location:       \\192.168.25.29\arx_file_archv
Description:    archive share for ARX file histories
Path:           /
Protocol:       CIFS
Proxy User:     acoProxy2
Status:         Online
FreeSpace:      1.1 GB

In use by:

Namespace             Volume                Snapshot Rule
--------------------  --------------------  --------------------
bstnA(gbl)# ...
```

Showing All Archive Configurations

Use the show global-config archive command to show the full configuration for all archives on the ARX:

```
show global-config archive
```

The output is a list of CLI commands that you can use to create each archive. For example, this shows the CLI commands required to recreate the “fileRecordsMed” archive:

```
bstnA(gbl)# show global-config archive
;============================== archive ================================
file-history archive fileRecordsMed
  description "archive share for ARX file histories"
  location filer fs4 cifs arx_file_archive proxy-user acoProxy2 path /
  exit

bstnA(gbl)# ...
```

Showing the Configuration for a Particular Archive

To focus on a single file-history archive (on an ARX with several of them), add the name of the archive to the end of the command:

```
show global-config archive name
```

where ***name*** (1-64 characters) identifies the archive to display.

Removing an Archive

If no volumes currently use this archive, you can remove it from the ARX. From gbl mode, use the no file-history archive command to remove an archive from the ARX configuration:

```
no file-history archive name
```

where ***name*** (1-64 characters) identifies the archive to remove.

For example, the following command removes the "testFA" archive:

```
bstnA(gbl)# no file-history archive testFA
bstnA(gbl)# ...
```

Storing File History in the Archive

The next step in keeping your file history is to create snapshots of your volume's metadata and configuration, and to store those snapshots in your file-history archive. You can set up a snapshot rule to take these types of snapshots on a regular basis. An earlier chapter explained the CLI commands for configuring snapshots; recall Appendix 2, Configuring Volume Snapshots. This section explains the CLI commands for setting up a snapshot rule for file tracking.

You begin by creating a snapshot rule in the volume where you want to track files. If a snapshot rule already exists in the volume (for client data), you can augment that rule to support file tracking. As described above, the volume must be a managed volume, and its metadata share must reside on a filer that supports snapshots.

◆ Note

The volume itself may or may not support snapshots for its clients. There is no requirement for snapshot support on any share except the volume's metadata share.

For example, this command sequence enters the "medarcv~/rcrds" volume and creates a new snapshot rule, "rcrdsArchive:"

```
bstnA(gbl)# namespace medarcv volume /rcrds
bstnA(gbl-ns-vol[medarcv~/rcrds])# snapshot rule rcrdsArchive

This will create a new policy object.

Create object 'rcrdsArchive'? [yes/no] yes
bstnA(gbl-ns-vol-snap[medarcv~/rcrds~rcrdsArchive])# ...
```

Choosing a File-History Archive

From gbl-ns-vol-snap mode, use the archive command to select a file-history archive for the current snapshot rule:

archive ***name***

where ***name*** (1-64 characters) selects an existing archive.

This is the storage depot for up to seven years of file-history data.

For example, the following command sequence selects the "fileRecordsMed" archive for the "rcrdsArchive" rule:

```
bstnA(gbl)# namespace medarcv volume /rcrds
bstnA(gbl-ns-vol[medarcv~/rcrds])# snapshot rule rcrdsArchive
```

```
bstnA(gbl-ns-vol-snap[medarcv~/rcrds~rcrdsArchive])# archive fileRecordsMed
bstnA(gbl-ns-vol-snap[medarcv~/rcrds~rcrdsArchive])# ...
```

Removing the File-History Archive from the Rule

You can disconnect the current snapshot rule from its file-history archive with the no archive command. This prevents future snapshots from tracking any file history. This is the default for a typical snapshot rule, one configured only for client-data snapshots.

```
no archive
```

For example, the following command sequence removes the file-history archive from a snapshot rule in the "/lab_equipment" volume:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# snapshot rule hourlySnap
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~hourlySnap])# no archive
bstnA(gbl-ns-vol-snap[medarcv~/lab_equipment~hourlySnap])# ...
```

Selecting the Contents of the Snapshot

A snapshot can contain one or two forms of data:

- user (client) data and/or
- volume-configuration data, possibly with volume metadata.

A snapshot includes user data by default. This is client-accessible data, and it is not necessary for file-tracking. Volume-configuration data and volume metadata are the file-history records that you can store in a file-history archive.

From gbl-ns-vol-snap mode, use the contents command to determine the contents of this rule's snapshots:

```
contents {user-data | volume-config [metadata]}
```

where

user-data selects client-viewable files and directories. This is the default for a snapshot rule.

volume-config selects the current configuration of the volume. This stores the mappings of front-end shares to back-end paths.

metadata (optional) chooses the volume's metadata, too. The metadata contains the locations of all client files and directories in the volume.

For a typical file-tracking installation, the snapshot rule contains both the configuration and the metadata. For example, the following command sequence selects these contents for the "rcrdsArchive" rule:

```
bstnA(gbl)# namespace medarcv volume /rcrds
bstnA(gbl-ns-vol[medarcv~/rcrds])# snapshot rule rcrdsArchive
bstnA(gbl-ns-vol-snap[medarcv~/rcrds~rcrdsArchive])# contents volume-config metadata
bstnA(gbl-ns-vol-snap[medarcv~/rcrds~rcrdsArchive])# ...
```

Excluding Contents from the Snapshot

You can use no contents to remove either user data or the volume configuration from this rule's snapshots:

```
no contents {user-data | volume-config}
```

where

user-data removes client-viewable files and directories from the rule's snapshots.

volume-config removes the volume configuration and metadata from future snapshots.

A file-tracking application does not require any user data in its snapshots. A snapshot rule never sends user data to the file-history archive.

For example, the following command sequence removes user data from the "rcrdsArchive" rule's snapshots:

```
bstnA(gbl)# namespace medarcv volume /rcrds
bstnA(gbl-ns-vol[medarcv~/rcrds])# snapshot rule rcrdsArchive
bstnA(gbl-ns-vol-snap[medarcv~/rcrds~rcrdsArchive])# no contents user-data
bstnA(gbl-ns-vol-snap[medarcv~/rcrds~rcrdsArchive])# ...
```

Effects on the Metadata Filer

A snapshot rule typically cleans up its metadata snapshot after successfully copying it to the archive. That is, the rule removes the snapshot from the filer behind the metadata share. This conserves space on the metadata filer.

If a snapshot rule includes user-data as well as volume metadata, the metadata filer keeps the same number of snapshots as the user-data filers. The total number of snapshots is controlled by the rule's retain directive, described in *Changing the Number of Retained Snapshots (optional)*, on page 2-8.

Applying a Schedule (optional)

As described in the chapter about snapshots, the ARX volume can create snapshots on a regular schedule. A schedule is not required because you can invoke a snapshot rule manually (recall *Creating a Manual Snapshot*, on page 2-31). However, it is recommended for file tracking.

To create a schedule, use the gbl schedule command (refer to *Appendix 12, Creating a Policy Schedule* in the *ARX® CLI Storage-Management Guide* for details). To apply a schedule to the snapshot rule, use the schedule command in gbl-ns-vol-snap mode.

The best practice for scheduling is to configure daily snapshots; one file-tracking record per day is sufficient for most installations. Another good practice is to choose a non-busy hour for the snapshot. Client access is blocked by a *VIP fence* while the snapshot is taking place, and other volumes in the same namespace may also be blocked. The VIP fence is discussed in more detail below.

For example, the following command sequence applies an early-morning schedule, “daily4am,” to the “rcrdsArchive” rule:

```
bstnA(gbl)# namespace medarcv volume /rcrds
bstnA(gbl-ns-vol[medarcv~/rcrds])# snapshot rule rcrdsArchive
bstnA(gbl-ns-vol-snap[medarcv~/rcrds~rcrdsArchive])# schedule daily4am
bstnA(gbl-ns-vol-snap[medarcv~/rcrds~rcrdsArchive])# ...
```

For detailed instructions on using a schedule in a snapshot rule, refer back to *Applying a Schedule (optional)*, on page 2-9.

Configuring a Progress Report (optional)

A progress reports shows all the milestones and results of a snapshot. We recommend this to verify that all filer snapshots succeeded, or to diagnose problems if one of them failed. From gbl-ns-vol-snap mode, use the report command to generate this report every time the rule creates a snapshot, or only when a snapshot has an error:

report *prefix* [error-only]

where **error-only** (optional) causes the snapshot rule to generate reports if there is an error in the snapshot operation. This is not recommended for snapshots with client data, as it makes snapshot reconstitution impossible for client shares. This option is acceptable for snapshots that are limited to volume configuration and/or metadata, because those snapshots are archived and do not require reconstitution.

Use the show reports command for a list of all reports, or show reports type Snapshot for a list of snapshot reports. Use show reports *report-name* to read the report, show reports status *report-name* to get a one-line summary of the report, grep to search through the full report, or tail reports *report-name* follow to follow a report as it is being written.

For example, the following command sequence enables reports for the “rcrdsArchive” rule, but only for snapshot operations that encounter an error:

```
bstnA(gbl)# namespace medarcv volume /rcrds
bstnA(gbl-ns-vol[medarcv~/rcrds])# snapshot rule rcrdsArchive
bstnA(gbl-ns-vol-snap[medarcv~/rcrds~rcrdsArchive])# report FA_rcrds error-only
bstnA(gbl-ns-vol-snap[medarcv~/rcrds~rcrdsArchive])# ...
```

For details about snapshot reports, including sample reports, refer back to *Configuring a Progress Report (optional)*, on page 2-11.

Enabling the Snapshot Rule

The final step in configuring any snapshot rule is to enable it. By default, the rule is disabled and ignored by policy software. You must enable the rule to run snapshots on a schedule *or* to invoke the snapshots manually. As described in *Enabling the Snapshot Rule*, on page 2-14, you use the enable command to enable the rule. For example, the following command sequence enables the “rcrdsArchive” rule:

```
bstnA(gbl)# namespace medarcv volume /rcrds
bstnA(gbl-ns-vol[medarcv~/rcrds])# snapshot rule rcrdsArchive
```

```
bstnA(gbl-ns-vol-snap[medarcv~/rcrds~rcrdsArchive])# enable
bstnA(gbl-ns-vol-snap[medarcv~/rcrds~rcrdsArchive])# ...
```

Client Access During an Archive Operation: the VIP Fence

When the snapshot rule takes a snapshot of the volume's metadata, the volume's clients are unable to modify the metadata until the snapshot is complete. This raises a *VIP fence*, which is discussed in *Enabling Snapshot Consistency*, on page 2-29. Every VIP that exports this volume puts up a VIP fence, thus blocking client access to any other volumes behind the same VIP or VIPs.

We recommend that you choose a metadata filer that can perform fast snapshot operations. If multiple volumes send their snapshots to a file archive, we recommend that they have all of their metadata shares backed by the same filer volume (or a small set of filer volumes), and that the ARX volumes share the same schedule for their snapshot rules. With the same schedule and the same filer volume(s), you can take advantage of *snapshot grouping*, discussed in *Snapshot Grouping*, on page 2-10.

Canceling an Archive Operation

You may have a situation where you want to cancel an archive operation. For example, the target filer may become unresponsive while the snapshot rule attempts to copy the volume's metadata.

◆ Note

This command cancels only the archive operation, which copies the volume configuration and/or metadata to the file-history archive. This has no effect on the snapshot operation that may be occurring on the volume's metadata share.

From priv-exec mode, you can use the cancel snapshot archive command to cancel an archiving process that is currently underway:

`cancel snapshot archive namespace` ***`ns`*** **`volume`** ***`vol-path`*** **`rule`** ***`rule-name`***

where

ns (1-30 characters) selects the namespace,

vol-path (1-1024 characters) is the managed volume, and

rule-name (1-1024 characters) identifies the snapshot rule that is currently running the archive operation.

The CLI prompts for confirmation before canceling the archive operation. Enter **yes** to proceed.

For example, the following command sequence exits to priv-exec mode and cancels a snapshot-archive that is currently underway:

```
bstnA(gbl)# end
bstnA# cancel snapshot archive namespace medarcv volume /rcrds rule rcrdsArchive
```

```
Confirming this command will cause all archiving operations in namespace ''medarcv'' volume
’’/rcrds''
 associated with rule ''rcrdsArchive'' to be cancelled. Proceed? [yes/no] yes
bstnA# ...
```

Removing all Snapshots Behind a Rule

You can use the snapshot remove command to delete all of the back-end snapshots behind a snapshot rule. If this snapshot rule includes user data as well as metadata, this also removes any metadata snapshots on the volume’s metadata filer. This command only removes filer snapshots, not the rule configuration.

You can skip this section if the snapshot rule only includes volume configuration and/or metadata (recall *Selecting the Contents of the Snapshot*, on page 3-13). If the snapshot rule does not include any user data, the rule deletes the metadata snapshot as soon as it is successfully copied to the archive. This prevents the metadata filer from accumulating unnecessary snapshots.

From priv-exec mode, use the snapshot remove command to remove the filer snapshots and/or checkpoints behind a snapshot rule and the volume’s metadata share. *Removing all Snapshots Behind a Rule*, on page 2-14, describes this command in detail.

For example, this command sequence exits to priv-exec mode and removes all snapshots (including any on the metadata share) behind the “dailyArchive” rule in the “/lab_equipment” volume:

```
bstnA(gbl)# end
bstnA# snapshot remove medarcv /lab_equipment dailyArchive

Confirmation of this command results in the removal all snapshots associated with snapshot rule
'dailyArchive' in namespace 'medarcv' volume '/lab_equipment'.  The snapshot rule is not deleted.

 Proceed? [yes/no] yes

  Starting snapshot operation in volume /lab_equipment, report name:
fileArchive_1_remove_20090106204139315.rpt


  Starting snapshot operation in volume /lab_equipment, report name:
fileArchive_0_remove_20090106204139315.rpt

bstnA# ...
```

Removing a Snapshot Rule

Removing a snapshot rule deletes its configuration from the ARX without affecting any back-end snapshots. If the rule is dedicated to volume configuration and metadata, the volume’s clients detect no change when it is removed.

From gbl-ns-vol mode, use the no snapshot rule command to remove a snapshot rule from the current volume. For details on this command, see *Removing a Snapshot Rule*, on page 2-18.

For example, the following command sequence removes the "test_fha" rule from the "medarcv~/lab_equipment" volume:

```
bstnA(gbl)# namespace medarcv volume /lab_equipment
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# no snapshot rule test_fha

The following snapshots are being managed by the rule 'test_fra':
        test_fha_2(acopia_3_200707101300_501f705c-8735-11d8-8936-a58a9e4556df_datavol4)
        test_fha_1(acopia_14_200707101400_501f705c-8735-11d8-8936-a58a9e4556df_datavol4)

If this command is confirmed, the rule is deleted and all associated data related to the
aforementioned snapshots will be removed from the switch.  The snapshots on the filers will not be
removed.

Proceed? [yes/no] yes
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# ...
```

Showing Historical Configurations

ARX installations add and change front-end services and back-end filers as they grow, so that file-history queries for today's services may not apply to configurations from the past. For example, suppose that the "srv1.family.org" service expands into two services in two new domains, "srv.brother.org" and "srv.sister.org." Assume this service split occurs in March. The "srv.sister.org" service did not exist until March, so a later query about the files "in srv.sister.org as of January 20" cannot succeed. Before you make a query about January's services, you need to know which services existed in January.

For situations where the configuration has remained stable since before the times you want to query, you can skip to the next section.

For situations where you are unsure of the earlier configuration, you can use the show virtual path-history command from any mode. This command only functions after one or more snapshots have archived the volume's configuration:

```
show virtual path-history time-frame [report rpt-prefix] [archive arch-name] [verbose]
```

where

time-frame establishes the date(s) for your query. This takes several forms, discussed in the subsections below.

report *rpt-prefix* (optional, 1-255 characters) sends the output to a report. Without this option, the output appears at the CLI prompt. The report name is in *rpt-prefix_archive-name_date-time*.rpt format; the exact name appears after you enter the command. You can use show reports to list all reports, and show reports *report-name* to view any report.

archive *arch-name* (optional, 1-64 characters) specifies a particular file-history archive.

verbose (optional) expands the output with further details.

The following subsections describe the options for querying various time frames.

For a Particular Day

The most straightforward time-frame is a single date, shown in this syntax:

```
show virtual path-history date {today | date} [report rpt-prefix] [archive arch-name] [verbose]
```

where

today | ***date*** indicates that you want the configuration from a single day. Use *mm/dd/yyyy* format for a *date*.

The remaining options are explained above.

The output contains a block of information for each archived snapshot in the given time-frame. Each block has a heading, "As of *date time*," and contains configuration information followed by two tables. The configuration information identifies the file-history archive, the ARX, the global server that clients accessed to reach their files, and the particular namespace and volume containing the files. The first table contains one row per volume share, with the ARX-share name in one column and the share's back-end location in the other column. The second table has one row per front-end export, and maps the name of the client-visible export to its root directory in the ARX volume.

◆ Note

The ARX volume uses a filer's CLI to find its back-end-share locations. The volume software uses CLI-access parameters to find those paths, which are provided in the external-filer configuration. Recall **Preparing External Filers Behind the Volume's Shares***, on page 3-5**. As explained in that section, the paths are reduced to "best guesses" for any volume shares where the ARX has no CLI access.*

For example, the following command shows the full path history for a single day, 9/14/2009. This shows an archived snapshot for the "medarcv~/lab_equipment" volume, followed by another archived snapshot for the "medarcv~/rcrds" volume. From the Export tables in this output, we find that the "ac1.medarch.org" has a share named "labs" along with two other shares, and that it has several additional shares ("ARCHIVES," Y2005," and more) backed by the "/rcrds" volume. These data points will be useful later, and are shown in **bold** below:

```
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# show virtual path-history date 09/14/2009

Query Parameters
----------------------
Start date: Sep 14 2009
  End date: Sep 14 2009
   Archive: All
   Verbose: No

As of Sep 14 2009 01:02:00

  Archive:           fileRecordsMed
  Hosting Switch:    bstnA (d9bdece8-9866-11d8-91e3-f48e42637d58)
  Global Server:     ac1.medarch.org
  WINS Name:
  WINS Aliases:
  Dynamic DNS Names: ac1, fs1, fs2, fs5
  VIP:               192.168.25.15
  Namespace:         medarcv
  Volume:            /lab_equipment

  Share                      Shared Path -> Physical Path
  -------------------------  ------------------------------------------------
  equip                      \\nas10\equipment -> /vol/vol2/NTFS-QTREE/equipment
  leased                     \\nas10\for_lease -> /vol/vol1/NTFS-QTREE/for_lease
  backlots                   \\fs2\backlot_records -> e:\exports\backlot_records
  scanners                   \\fs5\xraysScanners -> e:\exports\xraysScanners

  Export                     Relative Virtual Path
```

```
  ------------------------  --------------------------------------------------
  labs                      \
  xraysScanners             \
  acopia#lab_equipment$     \


As of Sep 14 2009 01:07:00

  Archive:           fileRecordsMed
  Hosting Switch:    bstnA (d9bdece8-9866-11d8-91e3-f48e42637d58)
  Global Server:     ac1.medarch.org
  WINS Name:
  WINS Aliases:
  Dynamic DNS Names: ac1, fs1, fs2, fs5
  VIP:               192.168.25.15
  Namespace:         medarcv
  Volume:            /rcrds

  Share                     Shared Path -> Physical Path
  ------------------------  --------------------------------------------------
  rx                        \\fs4\prescriptions -> d:\exports\prescriptions
  charts                    \\fs1\histories -> d:\exports\histories
  bulk                      \\fs2\bulkstorage -> e:\exports\bulkstorage

  Export                    Relative Virtual Path
  ------------------------  --------------------------------------------------
  ARCHIVES                  \
  Y2005                     \2005
  bulkstorage               \
  acopia#rcrds$             \
  CELEBS                    \VIP_wing
  Y2004                     \2004

bstnA(gbl-ns-vol[medarcv~/lab_equipment])# ...
```

For a Range of Dates

You can set a start date and/or an end date for the path history's time frame. To set a range of dates in this way, use the start-date clause, the end-date clause, or both. These go in place of the date clause shown above.

This syntax shows all path history from some earlier start date until today:

```
show virtual path-history start-date date [report rpt-prefix] [archive arch-name] [verbose]
```

where ***date*** is some day in the past. Use *mm/dd/yyyy* format. The end date is today, implicitly.

This shows all path history from the first archived snapshot up until a particular end date:

```
show virtual path-history end-date {today | date} [report rpt-prefix] ...
```

where **today** | ***date*** is the last day in the query. Implicitly, the start date is the date of the first snapshot in the archive. Use *mm/dd/yyyy* format for a *date*.

This final option combines the two, so that you see all file-path history between two dates:

```
show virtual path-history start-date date end-date date [report rpt-prefix] ...
```

For example, this command shows all of the path configurations since a particular date in January. The command creates verbose output and sends it to a report:

```
bstnA(gbl)# show virtual path-history start-date 01/07/2009 report pathsSinceJan verbose
Generating report: pathsSinceJan_all_20090116063011.rpt

bstnA(gbl)#
```

As another example, this command shows all path configurations from a particular archive, "fileRecordsMed." This display comes from the earliest snapshot in the archive up to the last one of June 5th:

```
bstnA(gbl)# show virtual path-history end-date 06/05/2009 archive fileRecordsMed

Query Parameters
----------------------
Start date: Jan  1 1970
  End date: Jun  5 2009
   Archive: fileRecordsMed
   Verbose: No

As of Apr 23 2009 01:27:00

  Archive:           fileRecordsMed
  Hosting Switch:    bstnA (d9bdece8-9866-11d8-91e3-f48e42637d58)
  Global Server:     ac1.medarch.org
  WINS Name:
  WINS Aliases:
  Dynamic DNS Names: ac1, fs1, fs2, fs5
  VIP:               192.168.25.15
  Namespace:         medarcv
  Volume:            /lab_equipment

  Share                      Shared Path -> Physical Path
  -------------------------  --------------------------------------------------
  equip                      \\nas10\equipment -> /vol/vol2/NTFS-QTREE/equipment
  leased                     \\nas10\for_lease -> /vol/vol1/NTFS-QTREE/for_lease
  backlots                   \\fs2\backlot_records -> e:\exports\backlot_records
  scanners                   \\fs5\xraysScanners -> e:\exports\xraysScanners

  Export                     Relative Virtual Path
  -------------------------  --------------------------------------------------
  labs                       \
  xraysScanners              \
  acopia#lab_equipment$      \


As of Apr 23 2009 01:32:00

  Archive:           fileRecordsMed
  Hosting Switch:    bstnA (d9bdece8-9866-11d8-91e3-f48e42637d58)
  Global Server:     ac1.medarch.org
  WINS Name:
  WINS Aliases:
  Dynamic DNS Names: ac1, fs1, fs2, fs5
  VIP:               192.168.25.15
  Namespace:         medarcv
  Volume:            /rcrds
...

bstnA(gbl)#
```

For a Time Period Leading Up to an End Date

You can also count back from an end date with the show virtual path-history command. To accomplish this, you specify some number of days, weeks, months, and so on that lead up to the end date:

```
show virtual path-history count {days|weeks|months|quarters|years} before {today | date}  ...
```

where

count (1-100) is the number of days, weeks, or whatever you choose with the next argument.

days|weeks|...|years indicates the time unit.

today | *date* is the last day in the query. Use *mm/dd/yyyy* format for a *date*.

the remaining arguments, not shown above, are the ones described earlier: **report**, **archive**, and **verbose**.

For example, the following command shows all path histories for the past year from the "fileRecordsMed" archive:

```
bstnA(gbl)# show virtual path-history 1 year before today archive fileRecordsMed

Query Parameters
----------------------
Start date: Sep 14 2008
  End date: Sep 14 2009
   Archive: fileRecordsmed
   Verbose: No

As of Sep 14 2009 01:02:00

  Archive:           fileRecordsMed
  Hosting Switch:    bstnA (d9bdece8-9866-11d8-91e3-f48e42637d58)
  Global Server:     ac1.medarch.org
  WINS Name:
  WINS Aliases:
  Dynamic DNS Names: ac1, fs1, fs2, fs5
  VIP:               192.168.25.15
  Namespace:         medarcv
  Volume:            /lab_equipment

  Share                      Shared Path -> Physical Path
  -------------------------  ------------------------------------------------
  equip                      \\nas10\equipment -> /vol/vol2/NTFS-QTREE/equipment
  leased                     \\nas10\for_lease -> /vol/vol1/NTFS-QTREE/for_lease
  backlots                   \\fs2\backlot_records -> e:\exports\backlot_records
  scanners                   \\fs5\xraysScanners -> e:\exports\xraysScanners

  Export                     Relative Virtual Path
  -------------------------  ------------------------------------------------
  labs                       \
  xraysScanners              \
  acopia#lab_equipment$      \


As of Sep 14 2009 01:07:00

  Archive:           fileRecordsMed
  Hosting Switch:    bstnA (d9bdece8-9866-11d8-91e3-f48e42637d58)
```

```
  Global Server:     ac1.medarch.org
  WINS Name:
  WINS Aliases:
  Dynamic DNS Names: ac1, fs1, fs2, fs5
  VIP:               192.168.25.15
  Namespace:         medarcv
  Volume:            /rcrds

  Share                      Shared Path -> Physical Path
  -------------------------  ------------------------------------------------
  rx                         \\fs4\prescriptions -> d:\exports\prescriptions
  charts                     \\fs1\histories -> d:\exports\histories
  bulk                       \\fs2\bulkstorage -> e:\exports\bulkstorage

  Export                     Relative Virtual Path
  -------------------------  ------------------------------------------------
  ARCHIVES                   \
  Y2005                      \2005
  bulkstorage                \
  acopia#rcrds$              \
  CELEBS                     \VIP_wing
  Y2004                      \2004

...

bstnA(gbl)#
```

Showing File History

With the names of the front-end services and front-end shares in a given time frame, you can find the back-end locations of the service's files at that time. You can find file locations at any given day, as long as the service's volume takes and archives more and more of its configuration/metadata snapshots. The archive can hold several-years worth of volume data, and you can query a file's location back to the time of the first snapshot.

From priv-exec mode, use the show file-history virtual-service command to view the location of a particular file on a particular day. This is the general syntax of the command:

```
show file-history virtual-service server-and-share time-frame file-path [options]
```

where

server-and-share specifies the server name and the share name as seen by ARX clients during the ***time-frame***. Server and share names may change over time, so you can use the show virtual path-history command to find the correct names for that date, as described above.

time-frame is a time period when the snapshot rule archived the volume's configuration and metadata at least once.

file-path is the file or directory to seek.

options (all optional) include search parameters, such as recursion into subdirectories.

The subsections below describe specific syntax that you can use with this command.

For a Particular Day

The following syntax searches for one file or directory on a single day:

```
show file-history virtual-service fqdn fe-share date {today | date} {[file filename] [path path]}
```

where

fqdn (1-1024 characters) identifies the front-end service that clients use to access the file. As mentioned above, you can use show virtual path-history to find a service name that is appropriate to your ***date***. This can also be the VIP, WINS name, or WINS alias from the time (all included in the output of show virtual path-history).

fe-share (1-1024 characters) is the front-end share that clients use (or used) to access the file. This is case-sensitive for NFS shares (that is, "thisShare" is different than "THISshare"), but not for CIFS shares.

date {today | *date*} is the specific date for which you want the file's location. To specify a date other than today, use *mm*/*dd*/*yyyy* format. A query for **today** applies to the time of day when the

file-history record was archived; for a file location as of this minute, use the find command or run a nsck ... report metadata-only report. Both of these commands are described in later chapters.

[file *filename*] [path *path*] chooses the file or directory to seek. This is required: you must select the **file**, the **path** (a directory), or both.

- ***filename*** (1-255 characters) is the file itself. This is not case sensitive: for example, an entry of "myfile.txt" would match an actual filename of "myFile.txt" or "MYFILE.TXT." You can also use wildcard characters and sequences supported by the *wildmat* standard: ***** means 0 or more characters, **?** means any single character, **[0-9]** means any single digit, **[^a-z]** means any single character except a lowercase letter, and so on. This searches the root directory of the ***fe-share***, or the ***path*** that you specify in the next argument.
- ***path*** is a directory path, relative to the root of the ***fe-share***. You use forward slashes (/) in an NFS share or a CIFS share: for example, "myDir/yourDir" is a valid path in any share. This also supports wildmat expressions.

The output shows the date you have chosen, the configuration of the archive and ARX service, and a separate table for each snapshot that contains the file record. For example, if the snapshot rule took three snapshots on the date you gave, there would be three sub tables. Each sub table shows the time of the snapshot/archive operation, the file's namespace and volume, the file's back-end share (in "*external-filer-name**share-name*" format), and the file's back-end path.

Note

The ARX volume uses a filer's CLI to find its back-end-share paths. The volume software uses CLI-access parameters to find those paths, which are typically provided in the external-filer configuration for snapshot-supporting filers (recall *Preparing External Filers Behind the Volume's Shares, on page 3-5**). As explained in that section, the paths are reduced to "best guesses" for any volume shares where the ARX has no CLI access.*

Sample: Finding a Missing File

For example, suppose you receive a trouble ticket indicating an accidentally-deleted file in the "ac1.medarch.org" service. Someone accessed the "ARCHIVES" share in that service and accidentally erased the "2005/planA/a_adams.dat" file. The most-recent backup for your filers occurred on September 14. This command sequence exits to priv-exec mode and shows the filer-location of that file on the 14th. The output indicates that

the file was on “fs1” on that date, at “d:\exports\histories\2005\planA.” If you concatenate the final two fields of the output (highlighted below), you get a complete path of “d:\exports\histories\2005\a_adams.dat”.

```
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# end
bstnA# show file-history virtual-service ac1.medarch.org ARCHIVES 09/14/2009 today file
a_adams.dat path /2005/planA

Query Parameters
----------------
Virtual Server: ac1.MEDARCH.ORG
        Export: ARCHIVES
    Start date: Jun 30 2010
      End date: Jun 30 2010
     File Name: a_adams.dat
          Path: /2005/planA
       Archive: All
Case Sensitive: No
       Recurse: No
       Verbose: No

Archive:           fileRecordsMed
Hosting Switch:    bstnA (d9bdece8-9866-11d8-91e3-f48e42637d58)

  ------------------------------------------------------------------------------
  Archive Date/Time: Jun 30 2010 01:11:00
  Global Server:     ac1.MEDARCH.ORG
  WINS Name:
  WINS Aliases:
  Dynamic DNS Names: ac1, fs1, fs2, fs5
  VIP:               192.168.25.15
  Namespace:         medarcv
  Volume Path:       /rcrds/2005/planA
  File Server(s):

      Shared Path: \\fs1\histories
    Physical Path: d:\exports\histories\2005\planA
        File Name: a_adams.dat


bstnA# ...
```

With Input/Output Options

You have several options at the end of the show file-history command to control the search parameters and the output. You can use any of these options at the end of any show file-history virtual-service command:

show file-history ... [recurse] [case-sensitive] [verbose] [report *rpt-prefix*] [archive *arch-name*]

where

recurse (optional) extends the search into subdirectories.

case-sensitive (optional) limits the search to the exact filename or path that you typed. This means that an entry of “INDEX.html” does not match the “index.html” file. By default, “INDEX.html” would match “index.html,” “Index.html,” or any other case combination.

verbose (optional) expands the output with further details. If you specify a **path** with this option, the output is a list of the files in all matching directories.

report *rpt-prefix* (optional, 1-255 characters) sends the output to a report. Without this option, the output appears at the CLI prompt. The report name is in *rpt-prefix_archive-name_date-time*.rpt format; the exact name appears after you enter the command. You can use show reports to list all reports, and show reports *report-name* to view any report.

archive *arch-name* (optional, 1-64 characters) specifies a particular file-history archive.

For example, this command searches for instances of "index.html" with some additional options: this searches recursively, generates verbose output, and sends the output to a report file.

```
bstnA(gbl-ns-vol[medarcv~/lab_equipment])# end
bstnA# show file-history virtual-service ac1.medarch.org labs date today file index.html recurse
verbose report indexPaths
Generating report: indexPaths_ac1.medarch.org_labs_20090118063429.rpt

bstnA# ...
```

For a Range of Dates

You can set a start date and/or an end date for the file's time frame. To set a range of dates in this way, use the start-date clause, the end-date clause, or both. These go in place of the date clause shown above.

This syntax shows file history from some earlier start date until today:

```
show file-history virtual-service fqdn fe-share start-date date {[file filename] [path path]} ...
```

where ***date*** is some day in the past. The end date is today, implicitly. Use *mm/dd/yyyy* format.

The next syntax shows a file's history from the first archived snapshot up until a particular end date:

```
show file-history virtual-service fqdn fe-share end-date {today | date} ...
```

where **today | *date*** is the last day in the query. Implicitly, the start date is the date of the first snapshot in the archive. Use *mm/dd/yyyy* format for a *date*.

This final option combines the two, so that you see the file history between two dates:

```
show file-history virtual-service fqdn fe-share start-date date end-date {today | date} ...
```

For example, this command shows all locations for all "index.html" files since a date in November of 2008:

```
bstnA(gbl)# end
bstnA# show file-history virtual-service ac1.medarch.org labs start-date 11/12/2008 file
index.html recurse

Query Parameters
----------------
```

```
Virtual Server: ac1.MEDARCH.ORG
        Export: labs
    Start date: Nov 12 2008
      End date: Jun 30 2010
     File Name: index.html
          Path:
       Archive: All
Case Sensitive: No
       Recurse: Yes
       Verbose: No

Archive:           fileRecordsMed
Hosting Switch:    bstnA (d9bdece8-9866-11d8-91e3-f48e42637d58)

  ------------------------------------------------------------------------------
  Archive Date/Time: Sep 14 2009 01:02:00
  Global Server:     ac1.medarch.org
  WINS Name:
  WINS Aliases:
  Dynamic DNS Names: ac1, fs1, fs2, fs5
  VIP:               192.168.25.15
  Namespace:         medarcv
  Volume Path:       /lab_equipment
  File Server(s):

      Shared Path: \\nas10\for_lease
    Physical Path: /vol/vol1/NTFS-QTREE/for_lease
        File Name: index.html


  ------------------------------------------------------------------------------
  Archive Date/Time: Sep 14 2009 01:02:00
  Global Server:     ac1.medarch.org
  WINS Name:
  WINS Aliases:
  Dynamic DNS Names: ac1, fs1, fs2, fs5
  VIP:               192.168.25.15
  Namespace:         medarcv
  Volume Path:       /lab_equipment/acme
  File Server(s):

      Shared Path: \\nas10\for_lease
    Physical Path: /vol/vol1/NTFS-QTREE/for_lease/acme
        File Name: index.html


  ------------------------------------------------------------------------------
  Archive Date/Time: Sep 14 2009 01:02:00
  Global Server:     ac1.medarch.org
  WINS Name:
  WINS Aliases:
  Dynamic DNS Names: ac1, fs1, fs2, fs5
  VIP:               192.168.25.15
  Namespace:         medarcv
  Volume Path:       /lab_equipment/miscDiags
  File Server(s):

      Shared Path: \\nas10\equipment
    Physical Path: /vol/vol2/NTFS-QTREE/equipment/miscDiags
        File Name: index.html

...
```

```
bstnA#
```

For a Time Period Leading Up to an End Date

You can also count back from an end date with the show file-history virtual-service command. To accomplish this, you specify some number of days, weeks, months, and so on that lead up to the end date:

show file-history virtual-service *fqdn fe-share count* {days|weeks|months|quarters|years} before {today | *date*} ...

where

count (1-100) is the number of days, weeks, or whatever you choose with the next argument.

days|weeks|...|years indicates the time unit.

today | *date* is the last day in the query. Use *mm/dd/yyyy* format for a *date*.

the remaining arguments, not shown above, are the ones described earlier.

For example, the following command sequence displays the location of a PDF file in the “labs” share over the last three days:

```
bstnA(gbl)# end
bstnA# show file-history virtual-service ac1.medarch.org labs 3 days before today file
reagentlist.pdf recurse

Query Parameters
----------------
Virtual Server: ac1.MEDARCH.ORG
        Export: labs
    Start date: Jun 27 2010
      End date: Jun 30 2010
     File Name: reagentlist.pdf
          Path:
       Archive: All
Case Sensitive: No
       Recurse: Yes
       Verbose: No

Archive:           fileRecordsMed
Hosting Switch:    bstnA (d9bdece8-9866-11d8-91e3-f48e42637d58)

  ------------------------------------------------------------------------------
  Archive Date/Time: Jun 30 2010 01:06:00
  Global Server:     ac1.medarch.org
  WINS Name:
  WINS Aliases:
  Dynamic DNS Names: ac1, fs1, fs2, fs5
  VIP:               192.168.25.15
  Namespace:         medarcv
  Volume Path:       /lab_equipment/acme
  File Server(s):

      Shared Path: \\nas10\for_lease
    Physical Path: /vol/vol1/NTFS-QTREE/for_lease/acme
        File Name: reagentList.pdf
```

```
...
bstnA#
```

Finding a File's Current Location

Using a virtual-file path, visible through a VIP, you can find a file's location on its actual back-end filer. This is the location of the file now, as opposed to an earlier location that you can discover with show file-history commands from above.

From any mode, use the find command to find a file's current back-end location:

find host *hostname-or-vip* {cifs | nfs} *share-name* path *path*

where

hostname-or-vip (1-255 characters) is the DNS hostname or VIP that clients use to access the file (syntax for a WINS name appears later), To use a hostname defined at an external DNS server, you must first identify the server with the ip name-server command. For instructions on using this command, see *Configuring DNS Lookups*, on page 4-31 of the *ARX® CLI Network-Management Guide*.

{cifs | nfs} ***share-name*** (1-4096 characters) is the share that clients use, and

path (1-4096 characters) specifies the client-visible path to the file. Use forward slashes (/), even for paths in CIFS shares.

For example, this command discovers that the "a_adams.dat" file from earlier examples is currently on the filer at 192.168.25.27. The earlier examples showed this file on a different filer when it was last backed up:

```
bstnA(gbl)# find global-server ac1.medarch.org cifs ARCHIVES path /2005/planA/a_adams.dat

Namespace:              medarcv
Logical path:           /rcrds/2005/planA/a_adams.dat
CIFS Physical location: //192.168.25.27/bulkstorage/2005/planA/a_adams.dat

bstnA(gbl)# ...
```

For another example, the following command sequence shows two back-end paths (one through NFS and another through CIFS) to "\\STATS\carrierCrossCheck.html" at the virtual-IP address, "192.168.25.15."

```
bstnA(gbl)# find host 192.168.25.15 cifs STATS path /carrierCrossCheck.html

Namespace:              insur
Logical path:           /claims/stats/carrierCrossCheck.html
NFS Physical location:  192.168.25.21:/vol/vol1/NTFS_QTREE/insurance/stats/carrierCrossCheck.html
CIFS Physical location: //192.168.25.21/insurance/stats/carrierCrossCheck.html
bstnA(gbl)# ...
```

Showing the NFS Filehandles

You can also show the NFS filehandles, both from the client perspective and the server perspective. Use the optional verbose keyword at the end of the command to add the filehandles to the output:

find host *hostname-or-vip* {cifs | nfs} *share-name* path *path* verbose

where **verbose** (optional) adds the NFS filehandles to the output. The other options were explained above.

This shows both the Virtual File Handle, which is the one that the ARX presents to NFS clients, and the Physical File Handle, which the ARX received from the back-end filer.

For example, this shows the location of the carrierCrossCheck.html file along with its NFS filehandles:

```
bstnA(gbl)# find host 192.168.25.15 nfs /claims path /stats/carrierCrossCheck.html verbose

Namespace:              insur
Logical path:           /claims/stats/carrierCrossCheck.html
NFS Physical location:  192.168.25.21:/vol/vol1/NTFS_QTREE/insurance/stats/carrierCrossCheck.html
CIFS Physical location: //192.168.25.21/insurance/stats/carrierCrossCheck.html

  NFSv3
    Virtual File Handle  (32 bytes):
      0x000000010c000000900000000000700000000000000000000000000e00000000
    Physical File Handle (32 bytes):
      0x1e0300000100000084000000fc2293491e0300000100000002000000000000000
bstnA(gbl)#
```

Finding the File with a Global-Server Name

You can also use the global-server name and path to find a file:

find global-server *fqdn* {cifs | nfs} *share-name* path *path* [verbose]

where

fqdn (1-255 characters) identifies the global server that clients use to access the file (for example, "myServer.com"),

{cifs | nfs} ***share-name*** (1-4096 characters) is the share that clients use, and

path (1-4096 characters) specifies the client-visible path to the file. As above, use forward slashes (/) for all paths.

verbose (optional) adds the file's NFS filehandles to the output.

For example, the following command sequence shows two back-end paths to "\\CLAIMS\index.html" on the global server named "ac1.MEDARCH.ORG:"

```
bstnA(gbl)# find global-server ac1.MEDARCH.ORG cifs CLAIMS path /index.html

Namespace:              insur
Logical path:           /claims/index.html
NFS Physical location:  192.168.25.21:/vol/vol1/NTFS_QTREE/insurance/index.html
CIFS Physical location: The file/directory on //192.168.25.21/insurance has an inconsistent name.
```

```
bstnA(gbl)# ...
```

Finding the File with a WINS Name

A CIFS front-end service can advertise its shares using a NetBIOS name registered with a WINS server (see *Setting the NetBIOS Name (optional, CIFS)*, on page 10-7 of the *ARX® CLI Storage-Management Guide*). If you used the optional wins-name and/or wins-alias command to set up a special NetBIOS name, you can use this name to find a file:

`find wins` ***`netbios-name`*** **`{cifs | nfs}`** ***`share-name`*** **`path`** ***`path`*** **`[verbose]`**

where

netbios-name (1-255 characters) identifies a NetBIOS name for a global server, advertised by a WINS server (for example, "CIFSSERVER"),

{cifs | nfs} ***share-name*** (1-4096 characters) is the share that clients use, and

path (1-4096 characters) specifies the client-visible path to the file. As above, use forward slashes (/) for all paths.

verbose (optional) adds the file's NFS filehandles to the output.

For example, this command finds a file in a front-end service with the WINS name, "INSURANCE:"

```
bstnA(gbl)# find wins INSURANCE cifs CLAIMS path /index.html

Namespace:              insur
Logical path:           /claims/index.html
NFS Physical location:  192.168.25.21:/vol/vol1/NTFS_QTREE/insurance/index.html
CIFS Physical location: The file/directory on //192.168.25.21/insurance has an inconsistent name.
bstnA(gbl)# ...
```

Finding the File from a Namespace Perspective

You can also use the namespace name and path to find a file:

`find namespace` ***`namespace`*** **`path`** ***`path`*** **`[verbose]`**

where

namespace (1-30 characters) identifies the namespace of the file,

path (1-4096 characters) specifies the client-visible path to the file, and

verbose (optional) adds the file's NFS filehandles to the output. This output is sometimes incomplete; use one of the other versions of the find command (such as find host) to guarantee complete NFS filehandles.

For example, the following command sequence shows the back-end path to "/acct/index.html" in the wwmed namespace:

```
bstnA(gbl)# find namespace wwmed path /acct/index.html

Namespace:             wwmed
Logical path:          /acct/index.html
NFS Physical location:  192.168.25.19:/exports/budget/index.html
```

As another example, this command shows the NFS filehandles, too:

```
bstnA(gbl)# find namespace wwmed path /acct/index.html verbose

Namespace:             wwmed
Logical path:          /acct/index.html
NFS Physical location:  192.168.25.19:/exports/budget/index.html

 (Note: A value of '-' in the filehandle shows an unknown field.
  Choose a different path query type to fill in this information)

  NFSv3
    Virtual File Handle  (24 bytes):
      0x000080001400000001000000000002000000000000000000-------00000000
    Physical File Handle (24 bytes):
      0x0100000200080009020000098c08c0033e02b839fc08c00
bstnA(gbl)# ...
```

Maintaining a File-History Archive: Listing Records

You may wish to monitor the disk space used by file-history records in the file-history archive. For a list of the records in an archive, you can expand on the show file-history archive command discussed earlier (recall *Listing All File-History Archives*, on page 3-10). Use the name of a particular archive followed by the contents keyword, a time frame, and possibly some options:

```
show file-history archive name contents time-frame [namespace ns volume path] [report rpt-prefix]
```

where

name (1-64 characters) identifies the desired archive.

contents is a required keyword.

time-frame establishes the date(s) for your query. This takes several forms, discussed in the subsections below.

ns (optional, 1-30 characters) selects the records from a particular ARX namespace.

vol-path (optional, 1-1024 characters) focuses on one ARX volume.

report *rpt-prefix* (optional, 1-255 characters) sends the output to a report. Without this option, the output appears at the CLI prompt. The report name is in *rpt-prefix_archive-name_date-time*.rpt format; the exact name appears after you enter the command. You can use show reports to list all reports, and show reports *report-name* to view any report.

The subsections below explain the various time frames that you can specify for this command.

For a Particular Day

The most straightforward time-frame is a single date, shown in this syntax:

```
show file-history archive name contents date {today | date} [namespace ns volume path] ...
```

where

today | ***date*** indicates that you want the archive contents from the given day. This shows any and all snapshots that were archived today or on the date you provide. Use *mm/dd/yyyy* format for a *date*.

The remaining options are explained above.

The output contains three tables. The first table shows the chosen date range, the second lists all the snapshots archived on the given date (one row per snapshot), and the final table summarizes the total space consumed by the snapshots shown.

For example, the following command shows the snapshots added to the "fileRecordsMed" archive on the current day:

```
bstnA(gbl)# show file-history archive fileRecordsMed contents date today

Query Parameters
----------------
   Archive: fileRecordsMed
Start date: Jun 30 2010
  End date: Jun 30 2010
 Namespace:
    Volume:


Switch: bstnA (d9bdece8-9866-11d8-91e3-f48e42637d58)

Archive Date/Time     Namespace:Volume           Snapshot Rule  Config  Metadata
--------------------  -------------------------  -------------  ------  --------
Jun 30 2010 01:06:00  medarcv:/lab_equipment     labArchive     2.1 kB  32 kB
Jun 30 2010 01:11:00  medarcv:/rcrds             rcrdsArchive   2.2 kB  64 kB


Summary:
  Total space consumed by Volume Metadata:       96 kB
  Total space consumed by Volume Configuration:  4.3 kB
  Grand total space consumed:                    100 kB

bstnA(gbl)# ...
```

For another example, this command focuses on the snapshots for a single volume today, “medarcv~/rcrds:”

```
bstnA(gbl)# show file-history archive fileRecordsMed contents date today namespace medarcv volume
/rcrds

Query Parameters
----------------
   Archive: fileRecordsMed
Start date: Jun 30 2010
  End date: Jun 30 2010
 Namespace: medarcv
    Volume: /rcrds


Switch: bstnA (d9bdece8-9866-11d8-91e3-f48e42637d58)

Archive Date/Time     Namespace:Volume           Snapshot Rule  Config  Metadata
--------------------  -------------------------  -------------  ------  --------
Jun 30 2010 01:11:00  medarcv:/rcrds             rcrdsArchive   2.2 kB  64 kB


Summary:
  Total space consumed by Volume Metadata:       64 kB
  Total space consumed by Volume Configuration:  2.2 kB
  Grand total space consumed:                    66 kB

bstnA(gbl)# ...
```

For a Range of Dates

You can set a start date and/or an end date for archive’s contents. To set a range of dates in this way, use the start-date clause, the end-date clause, or both. These go in place of the date clause shown above.

This syntax shows all archive contents from some earlier start date until today:

```
show file-history archive name contents start-date date [namespace ns volume path] ...
```

where ***date*** is some day in the past. The end date is today, implicitly. Use *mm/dd/yyyy* format.

This shows all archived snapshots from the first one up until a particular end date:

```
show file-history archive name contents end-date {today | date} [namespace ns volume path] ...
```

where **today** | ***date*** is the last day in the query. Implicitly, the start date is the date of the first snapshot in the archive. Use *mm/dd/yyyy* format for a *date*.

This final option combines the two, so that you see all file-history snapshots that were archived between two dates:

```
show file-history archive name contents start-date date end-date date ...
```

For example, this command shows all of the path configurations since a particular date in January. The command creates verbose output and sends it to a report:

```
bstnA(gbl)# show file-history archive fileRecordsMed contents start-date 01/07/2009 report
flRcrdsSinceJan
Generating report: flRcrdsSinceJan_20090121093046.rpt

bstnA(gbl)#
```

For a Time Period Leading Up to an End Date

You can also count back from an end date with the show file-history archive command. To accomplish this, you specify some number of days, weeks, months, and so on that lead up to the end date:

```
show file-history archive name contents count {days|weeks|months|quarters|years} before {today |
    date} ...
```

where

count (1-100) is the number of days, weeks, or whatever you choose with the next argument.

days|weeks|...|years indicates the time unit.

today | ***date*** is the last day in the query. Use *mm/dd/yyyy* format for a *date*.

the remaining arguments, not shown above, are the ones described earlier: **namespace**, **volume**, and **report**.

For example, the following command shows all snapshots sent to the "fileRecordsMed" archive over the past 3 months:

```
bstnA(gbl)# show file-history archive fileRecordsMed contents 3 months before today

Query Parameters
----------------
   Archive: fileRecordsMed
Start date: Mar 30 2010
  End date: Jun 30 2010
```

```
 Namespace:
    Volume:


Switch: bstnA (d9bdece8-9866-11d8-91e3-f48e42637d58)

Archive Date/Time     Namespace:Volume           Snapshot Rule  Config  Metadata
--------------------  -------------------------  -------------  ------  --------
Jun 30 2010 01:06:00  medarcv:/lab_equipment     labArchive     2.1 kB  32 kB
Jun 30 2010 01:11:00  medarcv:/rcrds             rcrdsArchive   2.2 kB  64 kB


Summary:
  Total space consumed by Volume Metadata:       96 kB
  Total space consumed by Volume Configuration:  4.3 kB
  Grand total space consumed:                    100 kB...
```

Clearing Records from a File-History Archive

As time goes on, the number of file-history records (volume configurations, and possibly metadata snapshots) on the file-history archive grows. You may want to remove very-old records from the archive to conserve space. You may also want to remove all the file-history records for a removed service; recall *Removing a Full Namespace Service*, on page 10-49. Other occasions may also arise where it is useful to remove file-history records from the archive filer.

From priv-exec mode, use the clear file-history archive command to remove volume configurations (and possibly volume metadata) from an archive:

clear file-history archive *name* [metadata-only] [*time-frame*] [namespace *ns* volume *path*]

where

name (1-64 characters) identifies the desired archive.

metadata-only (optional) limits the clearing operation to volume metadata, and preserves all of the selected volume-configuration data. Without the volume metadata, you can query the front-end-service configurations from a given record (*Showing Historical Configurations*, on page 3-19), but you cannot query the locations of particular files or directories (*Showing File History*, on page 3-25).

time-frame (optional) identifies specific file-history record(s) to clear. If you omit this option, the command clears file-history records from the fullest span of time. A specific time frame takes several forms, discussed in the subsections below.

ns (optional, 1-30 characters) selects the records from a particular ARX namespace.

vol-path (optional, 1-1024 characters) focuses on one ARX volume.

The CLI prompts for confirmation before removing any file-history records from the archive; enter **yes** to proceed.

For example, the following command sequence exits to priv-exec mode and clears all file-history records from the "medarcv~/rcrds" volume. After this command, no file-history queries for this volume are possible in the "fileRecordsMed" archive:

```
bstnA(gbl)# end
bstnA# clear file-history archive fileRecordsMed namespace medarcv volume /rcrds

WARNING: Confirming this command removes all archived metadata and volume configuration data dated
 between '12/31/1969' and '01/22/2009' inclusive from archive 'fileRecordsMed'.

Proceed? [yes/no] yes
bstnA#
```

The subsections below explain the various time frames that you can specify for this command. These are the same time-frame formats used for the show file-history archive command described above.

For a Particular Day

The most straightforward time-frame is a single date, shown in this syntax:

```
clear file-history archive name [metadata-only] date {today | date} [namespace ns volume path]
```

where

today | *date* indicates that you want to clear the archive contents from the given day. This clears any and all configurations/snapshots that were archived today or on the date you provide. Use *mm*/*dd*/*yyyy* format for a *date*.

The remaining options are explained above.

For example, the following command sequence exits to priv-exec mode and clears the file-tracking records archived on a particular date in January. The clear command only removes records from a particular volume, "medarcv~/rcrds:"

```
bstnA(gbl)# end
bstnA# clear file-history archive fileRecordsMed date 01/07/2009 namespace medarcv volume /rcrds

WARNING: Confirming this command removes all archived metadata and volume configuration data dated
 between '01/07/2009' and '01/07/2009' inclusive from archive 'fileRecordsMed'.

Proceed? [yes/no] yes
bstnA#
```

For another example, this clears records for all volumes on the same date:

```
bstnA(gbl)# end
bstnA# clear file-history archive fileRecordsMed date 01/07/2009

WARNING: Confirming this command removes all archived metadata and volume configuration data dated
 between '01/07/2009' and '01/07/2009' inclusive from archive 'fileRecordsMed'.

Proceed? [yes/no] yes
bstnA#
```

For a Range of Dates

You can set a start date and/or an end date for clearing archive records. To set a range of dates in this way, use the start-date clause, the end-date clause, or both. These go in place of the date clause shown above.

This clears records from the first one up until a particular end date:

```
clear file-history archive name [metadata-only] end-date {today | date} ...
```

where **today | *date*** is the last day in the query. Implicitly, the start date is the date of the first snapshot in the archive. Use *mm/dd/yyyy* format for a *date*.

This clears archive records from some earlier start date until today:

```
clear file-history archive name [metadata-only] start-date date [namespace ns volume path]
```

where ***date*** is some day in the past. The end date is today, implicitly. Use *mm/dd/yyyy* format.

This final option combines the two, so that you clear file-history records that were archived between two dates:

```
clear file-history archive name [metadata-only] start-date date end-date date ...
```

For example, this command sequence clears volume metadata from file-history records up to a particular date in January:

```
bstnA(gbl)# end
bstnA#  clear file-history archive fileRecordsMed metadata-only end-date 01/07/2009

WARNING: Confirming this command removes all archived metadata dated
 between '12/31/1969' and '01/07/2009' inclusive from archive 'fileRecordsMed'.

Proceed? [yes/no] yes
bstnA(gbl)#
```

For a Time Period Leading Up to an End Date

You can also count back from an end date with the clear file-history archive command. To accomplish this, you specify some number of days, weeks, months, and so on that lead up to the end date:

```
clear file-history archive name [metadata-only] count {days|weeks|months|quarters|years} before
    {today | date} ...
```

where

count (1-100) is the number of days, weeks, or whatever you choose with the next argument.

days|weeks|...|years indicates the time unit.

today | *date* establishes the last day's records to be cleared. Use *mm/dd/yyyy* format for a *date*.

the remaining arguments, not shown above, are the ones described earlier: **namespace** and **volume**.

For example, the following command sequence clears the file-history records archived in "fileRecordsMed" over the 4 months leading up to January 7:

```
bstnA(gbl)# end
bstnA# clear file-history archive fileRecordsMed 4 months before 01/07/2009

WARNING: Confirming this command removes all archived metadata and volume configuration data dated
 between '09/07/2008' and '01/07/2009' inclusive from archive 'fileRecordsMed'.

Proceed? [yes/no] yes
bstnA# ...
```

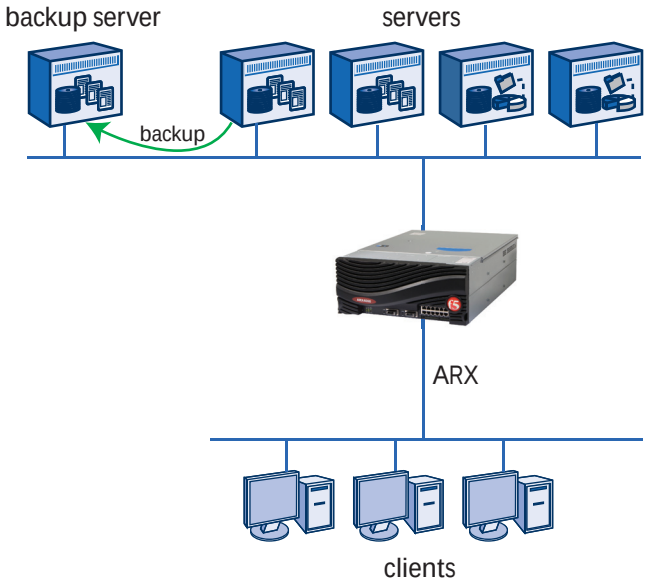
4

Restoring a Volume's Files

Overview

This chapter applies to managed volumes, where policy-based migrations complicate the backup and restore processes. Skip this chapter if your site only supports direct volumes.

You can back up client files by working around a managed volume and connecting directly to its back-end filers. Backups are read operations, so they cannot create inconsistencies in a volume's metadata. This can follow the same workflow that was used before the introduction of the ARX. For example, you might use NDMP on a data-protection device (called a *backup server* in this manual) to keep file-server backups:



Appendix 3, Tracking Files on Your Back-End Storage, explained how to find back-end file locations at any given time. This chapter explains some methods for restoring them to an ARX service.

Moving the Backup File(s) to the Staging Area

You can use the output from show file-history virtual-service or find to retrieve files from your backup server. For example, consider the following command which shows the location of the "a_adams.dat" file as of the last series of backups (September 14 in this example):

```
bstnA# show file-history virtual-service ac1.medarch.org ARCHIVES date 09/14/2009 file a_adams.dat
path /2005/planA

Date range
-----------------------
Start date: Sep 14 2009
  End date: Sep 14 2009

Archive:           fileRecordsMed
Hosting Switch:    bstnA (d9bdece8-9866-11d8-91e3-f48e42637d58)

  ------------------------------------------------------------------------------
  Archive Date/Time: Sep 14 2009 01:07:00
  Global Server:     ac1.medarch.org
  WINS Name:
  WINS Aliases:
  Dynamic DNS Names: ac1, fs1, fs2, fs5
  VIP:               192.168.25.15
  Namespace:         medarcv
  Volume Path:       /rcrds/2005/planA
  File Server(s):

      Shared Path: \\fs1\histories
    Physical Path: d:\exports\histories\2005\planA
        File Name: a_adams.dat

bstnA# ...
```

From this example, we found the following information about the file:

- as of September 14,
- the file was physically on the "fs1" filer, at "d:\exports\histories\2005\planA\a_adams.dat."

You can use the find command to locate the same file as of now. This shows that the file migrated to the filer at "192.168.25.27" since September 14:

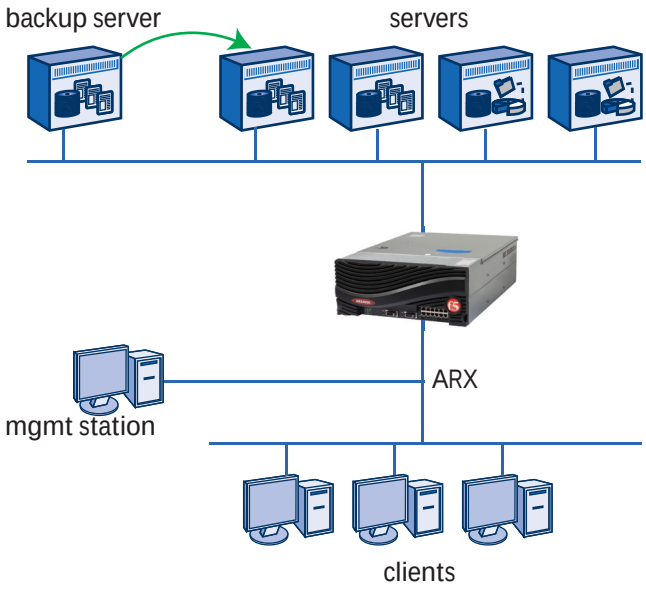
```
bstnA# find global-server ac1.medarch.org cifs ARCHIVES path /2005/planA/a_adams.dat

Namespace:              medarcv
Logical path:           /rcrds/2005/planA/a_adams.dat
CIFS Physical location: //192.168.25.27/bulkstorage/2005/planA/a_adams.dat
bstnA# ...
```

Using the earlier date and filer path, you can recover the file from your backup server. From the examples above, the file was on "fs1" on September 14, when backups occurred. At the backup server, we recover the

file from the September-14 backup of “fs1”. We suggest placing the recovered file back onto the original filer (“fs1”), but *outside any imported directory*. This side directory acts as a *staging area* for restored files.

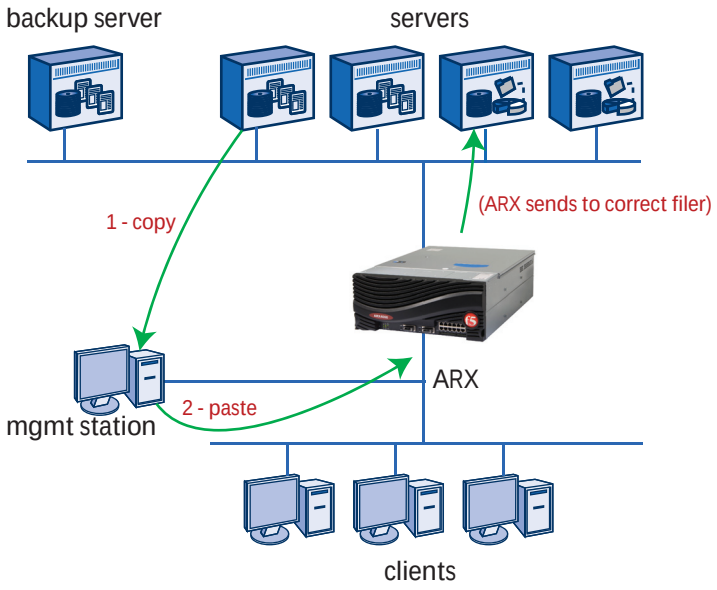
Important

*If the staging-area directory is inside any imported tree, this restore operation causes an inconsistency between the filer state and the metadata in a managed volume. We strongly advise that you choose a staging area that is **not** imported by any managed volume.*



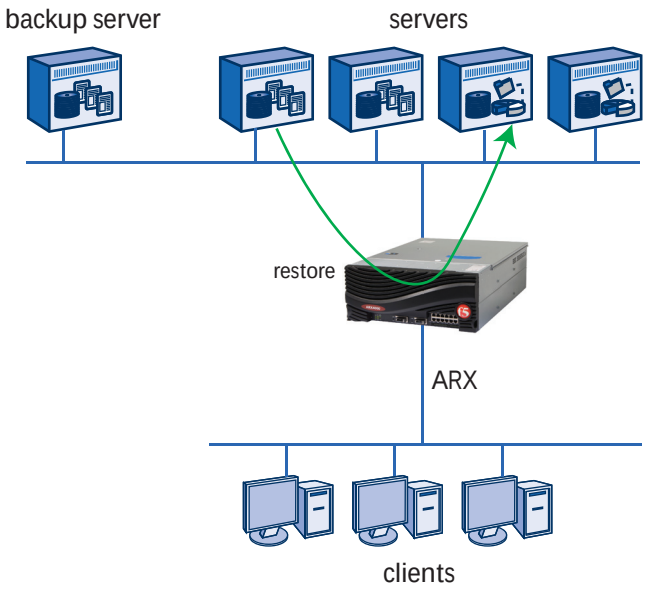
Manually Restoring File(s)

A trouble ticket typically includes the file location from the client's perspective: in this example, "\\ac1.medarch.org\ARCHIVES\2005\planA\a_adams.dat." From your management station, you can connect to the staging area through NFS or CIFS and also connect to the ARX service. For example, you could map your Y drive to "\\fs1\recoveryStage" on the filer and map your Z drive to "\\ac1.medarch.org\ARCHIVES" on the ARX. With both connections established, you can copy the file from the staging area into the front-end service. The ARX immediately places the file onto the filer where it should currently reside, based on your storage policies:



Automatically Restoring File(s)

A *restore* operation automatically copies a restored file from the staging area to its managed volume, which then places the file into the share where it currently resides:



This operation creates a new "restore" directory under the directory where the original file(s) should reside; from there, a client can compare the restored file(s) to the original(s) and choose between the two versions.

You can repeat the restore operation for later restores from the same staging area. This operation automatically distributes all restored files to their respective back-end shares.

Adding an External Filer for the Staging Area (optional)

Before you use the restore data command (described below), you need to identify the file server with the staging-area directory. This is the source for the ARX-restore operation. In most cases, the staging area is on the same filer where the file originally resided, in a directory outside of any imported share. In this case, the external-filer configuration already exists to support the imported share(s), and you can skip to the next section.

For cases where the staging area is on a filer that is currently unused by the ARX, configure it as an external filer. (For detailed instructions on this, refer to *Appendix 6, Adding an External Filer* in the *ARX® CLI Storage-Management Guide*.)

◆ Note

An external-filer configuration, as opposed to a simple IP address, is required to accommodate multi-homed filers.

For example, the following command sequence sets up a file server at 192.168.25.51:

```
bstnA(gbl)# external-filer nasE1
bstnA(gbl-filer[nasE1])# ip address 192.168.25.51
bstnA(gbl-filer[nasE1])# ...
```

Restoring Client Data

The restore data command invokes the restore operation:

restore data ***namespace*** **volume** ***vol*** **path** ***dest-path***
filer ***src-filer*** **{nfs** ***export*** **| cifs** ***share*****}+ source-path** ***src-path***
[recurse] [remove-source]

where

namespace **volume** ***vol*** **path** ***dest-path*** identifies the file or directory to restore:

- ***namespace*** is 1-30 characters. Use the namespace from the show file-history virtual-service or find output, shown earlier.
- ***vol*** is 1-1024 characters. Type **?** for a list of volumes in the above namespace, and choose the volume behind the client's front-end share. If you cannot find the complete path you need (such as "/claims/stats"), choose the volume that is the root of your path ("/claims").
- ***dest-path*** is also 1-1024 characters. This could be the remainder of the path you started with the ***vol***, above (for example, "/stats"). The restore process creates a "/restore" directory below this path, and places the restored file(s) in there.

filer ***src-filer*** **{nfs** ***export*** **| cifs** ***share*****}+ source-path** ***src-path*** points to the filer with the staging area and the backup file(s):

- ***src-filer*** (1-64 characters) is the external-filer name for the filer with the staging area,
- **nfs** ***export*** (1-1024 characters) identifies the NFS export, and
- **cifs** ***share*** (1-1024 characters) identifies the CIFS share. For a multi-protocol share, enter both paths: **nfs** ***export*** **cifs** ***share***.
- ***src-path*** (1-1024 characters) is the path to the backup file(s). This is the staging-area directory.

You can enter the optional **recurse** and **remove** flags in any order:

- **recurse** (optional) causes the restore operation to descend into subdirectories.
- **remove-source** (optional) activates a clean-up routine after the restore is finished; the restore operation removes the backup files after confirming a successful restore to the volume.

Every restore operation produces a report as it runs. The CLI shows the report name after you invoke the command. Use show reports for a full list of restore reports, and use show, tail, or grep to view one report. The report shows the progress of the operation.

For example, the following command sequence restores a full directory, "/claims/stats" in the "insur" namespace, then shows the restore report. The source of the restore is a multi-protocol share on external filer named "nasE1:"

```
bstnA(gbl)# end
bstnA# restore data insur volume /claims path stats filer nasE1 nfs /root_vdm_4/backups cifs
BACKUPS source-path /stats recurse
Scheduling restore operation on switch bstnA, report name: restore.7._claims.rpt
bstnA# ...
bstnA# show reports restore.7._claims.rpt
**** Restore Data Report: Started at Wed Feb 24 02:14:07 2010 ****
**** Software Version: 5.02.000.12538 (Feb 16 2010 20:14:13) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:
**** Namespace:         insur
**** Volume:            /claims
**** Path:              /stats/restore
**** Source Filer:      192.168.25.51
**** Source NFS Export: /root_vdm_4/backups
**** Source CIFS Share: BACKUPS
**** Source Path:       /stats
**** Options:           recurse

**** Legend:
 ****   OK = Object transferred without error.
 ****   SK = Skipped object due to naming problems.
 ****   FL = Encountered error during object transfer.

 ****   SY = Error re-syncing directory attributes after data restore.
 ****   DS = Error deleting source file or directory.
 ****   NR = Did not recurse due to problematic directory contents.
 ****   SA = Failed due to strict attribute consistency requirement.

 ****   FE = File entry.
 ****   DE = Directory entry.
 ****   LE = Hard link.
 ****   SE = Symbolic link.

**** Multi-Protocol Issues:
 ****   CC = Case-blind collision.
 ****   NE = Entry found with NFS that was not found with CIFS.
 ****   CE = Entry found with CIFS that was not found with NFS.
 ****   SL = Symbolic link found with NFS that was not found with CIFS.
 ****   IC = CIFS invalid characters found in NFS name.
 ****   NM = CIFS name has characters that are not mappable to the NFS encoding.
 ****   FN = A portion of the name contains a filer-generated pattern.
 ****   NC = Unable to copy CIFS data due to a filer-generated name.

Entry Type              Size   Object
-------------  -------------   --------------------------------------------------
[         CC]  [           ]   /stats/piechart.ppt
[         CC]  [           ]   /stats/PieChart.ppt
[         IC]  [           ]   /stats/on_the_job:2004.cnv
[         IC]  [           ]   /stats/on_the_job:2003.cnv
[         IC]  [           ]   /stats/in_home:2005
[OK    FE NC]  [  3,575,056]   /stats/PieChart.ppt
```

```
[OK    FE   ]  [      6,562]   /stats/acmeIns.txt
[OK    FE NC]  [    722,589]   /stats/piechart.ppt
[OK    FE NC]  [      1,679]   /stats/on_the_job:2003.cnv
[OK    FE NC]  [      9,509]   /stats/on_the_job:2004.cnv
[         IC]  [           ]   /stats/in_home:2005/age:11-21yrs.csv
[         IC]  [           ]   /stats/in_home:2005/age:>21yrs.csv
[         IC]  [           ]   /stats/in_home:2005/age:<10yrs.csv
[         IC]  [           ]   /stats/in_home:2005/age:>21yrs.csv
[         IC]  [           ]   /stats/in_home:2005/age:<10yrs.csv
[         IC]  [           ]   /stats/in_home:2005/age:11-21yrs.csv
[OK    DE NC]  [           ]   /stats/in_home:2005
[OK    FE NC]  [    335,872]   /stats/in_home:2005/age:11-21yrs.csv
[OK    FE NC]  [    267,264]   /stats/in_home:2005/age:>21yrs.csv
[OK    FE NC]  [     59,712]   /stats/in_home:2005/age:<10yrs.csv
[OK    FE   ]  [      2,432]   /stats/cleanBU.csh
[OK    FE   ]  [      4,545]   /stats/update.csh
[OK    FE   ]  [      1,423]   /stats/index.html
[OK    FE   ]  [      1,985]   /stats/makeCd.pl
[OK    FE   ]  [      4,110]   /stats/carrierCrossCheck.html
[OK    FE   ]  [          0]   /stats/21yrs.csv
[OK    DE   ]  [           ]   /stats/otj_latest
[OK    FE   ]  [      3,438]   /stats/otj_latest/feb.xls
[OK    FE   ]  [     71,462]   /stats/otj_latest/april.xls
[OK    FE   ]  [     17,279]   /stats/otj_latest/july.xls
[OK    FE   ]  [     17,629]   /stats/otj_latest/october.xls
[OK    FE   ]  [      3,321]   /stats/otj_latest/jan.xls

**** Total Found Items:                            21
**** Total Transferred Items:                      21
**** Total Failures:                                0
**** Total Bytes Restored:                  5,105,867

**** Total processed:              21
**** Elapsed time:           00:00:02
**** Restore Data Report: DONE at Wed Feb 24 02:14:09 2010 ****
bstnA# ...
```

Running the Restore from a Remote Host

From any remote host that supports the Secure SHell (SSH) protocol, you can run the restore data command and see the report. Use the following syntax with **ssh**:

ssh *admin-user@mip "restore data..."*

where

admin-user is the username for a valid administrative account at the ARX (use show users to list all of them, as shown in *Listing All Administrative Users*, on page 2-14 of the *ARX® CLI Network-Management Guide*),

mip is a management-IP address for the ARX (use show interface mgmt to show the out-of-band management interface, or show interface vlan to show all in-band management interfaces), and

restore data... is the full restore data command, described above. Surround this with quotation marks ("").

The output of the CLI command appears in the local shell. This also works with all show commands.

For example, the following command sequence re-runs the above restore data command from a remote machine, "mgmt17."

```
juser@mgmt17:~$ ssh admin@10.1.1.7 "restore data insur volume /claims path stats filer nasE1 nfs
/root_vdm_4/backups cifs BACKUPS source-path /stats recurse"
Command>restore data insur volume /claims path stats filer nasE1 nfs /root_vdm_4/backups cifs
BACKUPS source-path /stats recurse
Scheduling restore operation on switch bstnA, report name: restore.9._claims.rpt
juser@mgmt17:~$ ssh admin@10.1.1.7 "show reports restore.9._claims.rpt"
Command>show reports restore.9._claims.rpt
**** Restore Data Report: Started at Wed Feb 25 10:52:11 2010 ****
**** Software Version: 5.02.000.12538 (Feb 16 2010 20:14:13) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:
**** Namespace:         insur
**** Volume:            /claims
**** Path:              /stats/restore
**** Source Filer:      192.168.25.51
**** Source NFS Export: /root_vdm_4/backups
**** Source CIFS Share: BACKUPS
**** Source Path:       /stats
**** Options:           recurse

**** Legend:
 ****   OK = Object transferred without error.
 ****   SK = Skipped object due to naming problems.
 ****   FL = Encountered error during object transfer.

 ****   SY = Error re-syncing directory attributes after data restore.
 ****   DS = Error deleting source file or directory.
 ****   NR = Did not recurse due to problematic directory contents.
 ****   SA = Failed due to strict attribute consistency requirement.

 ****   FE = File entry.
 ****   DE = Directory entry.
 ****   LE = Hard link.
 ****   SE = Symbolic link.

**** Multi-Protocol Issues:
 ****   CC = Case-blind collision.
 ****   NE = Entry found with NFS that was not found with CIFS.
 ****   CE = Entry found with CIFS that was not found with NFS.
 ****   SL = Symbolic link found with NFS that was not found with CIFS.
 ****   IC = CIFS invalid characters found in NFS name.
 ****   NM = CIFS name has characters that are not mappable to the NFS encoding.
 ****   FN = A portion of the name contains a filer-generated pattern.
 ****   NC = Unable to copy CIFS data due to a filer-generated name.

Entry Type              Size   Object
-------------  -------------   -------------------------------------------------
[         CC]  [           ]   /stats/piechart.ppt
[         CC]  [           ]   /stats/PieChart.ppt
[         IC]  [           ]   /stats/on_the_job:2004.cnv
[         IC]  [           ]   /stats/on_the_job:2003.cnv
[         IC]  [           ]   /stats/in_home:2005
[OK    FE NC]  [  3,575,056]   /stats/PieChart.ppt
[OK    FE   ]  [      6,562]   /stats/acmeIns.txt
[OK    FE NC]  [    722,589]   /stats/piechart.ppt
[OK    FE NC]  [      1,679]   /stats/on_the_job:2003.cnv
```

```
[OK    FE NC]  [      9,509]   /stats/on_the_job:2004.cnv
[         IC]  [           ]   /stats/in_home:2005/age:11-21yrs.csv
[         IC]  [           ]   /stats/in_home:2005/age:>21yrs.csv
[         IC]  [           ]   /stats/in_home:2005/age:<10yrs.csv
[         IC]  [           ]   /stats/in_home:2005/age:>21yrs.csv
[         IC]  [           ]   /stats/in_home:2005/age:<10yrs.csv
[         IC]  [           ]   /stats/in_home:2005/age:11-21yrs.csv
[OK    DE NC]  [           ]   /stats/in_home:2005
[OK    FE NC]  [    335,872]   /stats/in_home:2005/age:11-21yrs.csv
[OK    FE NC]  [    267,264]   /stats/in_home:2005/age:>21yrs.csv
[OK    FE NC]  [     59,712]   /stats/in_home:2005/age:<10yrs.csv
[OK    FE   ]  [      2,432]   /stats/cleanBU.csh
[OK    FE   ]  [      4,545]   /stats/update.csh
[OK    FE   ]  [      1,423]   /stats/index.html
[OK    FE   ]  [      1,985]   /stats/makeCd.pl
[OK    FE   ]  [      4,110]   /stats/carrierCrossCheck.html
[OK    FE   ]  [          0]   /stats/21yrs.csv

**** Total Found Items:                           15
**** Total Transferred Items:                     15
**** Total Failures:                               0
**** Total Bytes Restored:                 4,992,738

**** Total processed:              15
**** Elapsed time:           00:00:02
**** Restore Data Report: DONE at Wed Feb 25 10:52:22 2010 ****
juser@mgmt17:~$
```

Showing all Restore Operations

A restore operation occurs in the background while you can run other CLI commands or GUI functions. To see the high-level status of all restore operations, use the show restore data command:

```
show restore data
```

For example, this shows that the current switch has had two successful restores:

```
bstnA(gbl)# show restore data

Namespace: wwmed
    Volume: /acct
        Path: /
            Options:              recurse
            Source Filer:         das1 [192.168.25.19]
            NFS Source:           /exports/backups/acct
            Status:               Success
            Report Name:          restore.6._acct.rpt (bstnA)
            Items Found:          225
            Items Transferred:    225
            Failures:             0
            Total Bytes Restored: 11,163,668
            Completed at:         Mon Sep 14 01:29:11 2009


Namespace: insur
    Volume: /claims
        Path: /stats
            Options:              recurse
            Source Filer:         nasE1 [192.168.25.51]
            NFS Source:           /root_vdm_4/backups/stats
```

```
            CIFS Source:          BACKUPS/stats
            Status:               Success
            Report Name:          restore.7._claims.rpt (bstnA)
            Items Found:          21
            Items Transferred:    21
            Failures:             0
            Total Bytes Restored: 5,105,867
            Completed at:         Mon Sep 14 01:28:01 2009

bstnA(gbl)# ...
```

Focusing on a Single Namespace, Volume, or Path

You can add some options to show a smaller set of restore operations:

`show restore data [`*`namespace`*` [volume `*`vol-path`*` [path `*`path`*`]]]`

where

namespace (1-30 characters) selects a namespace,

vol-path (1-1024 characters) narrows the scope to a specific volume, and.

path (1-1024 characters) narrows the scope further, to a specific virtual path in the volume (for example, '/home/jrandom'). This is relative to the ***vol-path***.

For example, this command shows the restore operation performed on one virtual directory, "insur~/claims/stats:"

```
bstnA(gbl)# show restore data insur volume /claims path /stats

Namespace: insur
    Volume: /claims
        Path: /stats
            Options:              recurse
            Source Filer:         nasE1 [192.168.25.51]
            NFS Source:           /root_vdm_4/backups/stats
            CIFS Source:          BACKUPS/stats
            Status:               Success
            Report Name:          restore.7._claims.rpt (bstnA)
            Items Found:          21
            Items Transferred:    21
            Failures:             0
            Total Bytes Restored: 5,105,867
            Completed at:         Mon Sep 14 01:28:01 2009

bstnA(gbl)# ...
```

Clearing Restore Operations from the Output

The show restore data command displays a running history of all restore operations. The history goes back indefinitely, so the command can display a very large number of records over time. You can clear all of these records at once, or all records for a particular namespace, volume, or path. From priv-exec mode, use the clear restore data command:

`clear restore data [`*`namespace`*` [volume `*`vol-path`*` [path `*`path`*`]]]`

where

namespace (optional, 1-30 characters) identifies one namespace with restore records. If this option is omitted, the command clears all restore-operation records from the history.

vol-path (optional, 1-1024 characters) narrows the scope to one volume, and

path (optional, 1-1024 characters) narrows the scope further to a specific virtual path. This is relative to the ***vol-path***.

Before clearing the records, a prompt requests confirmation; enter **yes** to confirm. After you clear the records, they no longer appear in the show restore data output.

This only affects completed restore operations. It does not affect any restore operation that is in-process when you run the clear command.

For example, this command sequence shows all restore records, clears them, then shows that there are none remaining:

```
bstnA(gbl)# show restore data

Namespace: wwmed
    Volume: /acct
        Path: /
...
Total Bytes Restored: 1,694,056
bstnA(gbl)# end
bstnA# clear restore data
Clear restore data records? [yes/no] yes
bstnA# show restore data
bstnA# ...
```

Canceling a Restore Operation

To cancel an in-progress restore operation, go to priv-exec mode and use the cancel restore data command:

cancel restore data *namespace* volume *vol-path* path *path*

where

namespace (1-30 characters) identifies the namespace,

vol-path (1-1024 characters) is the volume, and

path (1-1024 characters) is the specific virtual path that is being restored. This is relative to the ***vol-path***.

A prompt requests confirmation before the CLI cancels the restore; enter **yes** to confirm that you want to cancel the operation.

For example, this command cancels a restore operation in the "wwmed" namespace:

```
bstnA(gbl)# end
bstnA# cancel restore data wwmed volume /acct path /
Cancel specified restore data operations? [yes/no] yes
bstnA# ...
```

5

Backing Up the Running Configuration

Overview

The switch configuration that you edit with the CLI is called the running configuration, or *running config*. You can save the running config for the next reboot, disaster recovery, or for exporting the configuration from one switch to another.The running config is divided into two major components: the local-running config for the current switch, and global config for parameters that are shared by both switches in a redundant pair. This chapter explains how to save both config types and restore them later.

◆ Note

We recommend saving both the local-running config and the global config after every configuration change. Both configs are vital for switch replacement.

◆ Note

In addition, to prepare for replacing a single switch in the unlikely event of a failure, you should also save the following:

- *Master key (extracted and wrapped). Save it to a remote host along with the configs.*
- *Master key wrapping key password. Save it to a secure location.*

Setting a Default FTP or SCP User

Before you begin backing up the configuration, you have the option to simplify FTP uploads and/or SCP transfers later. The running config exists in one or more local files, which you can copy to an external FTP or SCP server. The default FTP username/password is "anonymous/upgrade-*hostname*," but you can enter a specific username/password for each copy. There is no default for SCP transfers. To avoid retyping FTP or SCP credentials each time, you can establish a default username and password for each transfer protocol FTP.

From cfg mode, use ip ftp-user to set the FTP username:

ip ftp-user *username*

where ***username*** is 1-32 characters.

From the same mode, you can also (or instead) use ip scp-user to set a default username for SCP:

ip scp-user *username*

where, as above, ***username*** is 1-32 characters.

The CLI then prompts twice for the password.

For example, the following command sequence sets up two default users named "juser," one for FTP and another for SCP:

```
bstnA(cfg)# ip ftp-user juser
```

```
Password: jpasswd
Validate Password: jpasswd
bstnA(cfg)# ip scp-user juser
Password: jpasswd
Validate Password: jpasswd
bstnA(cfg)# ...
```

Saving the Local Running Config

The next step in saving the running configuration is to save the local running config. The local *running config* applies only to the current switch: this config includes network and chassis parameters. From priv-exec mode, use the copy running-config command to save the local config as an executable script.

copy running-config scripts *destination-file*

where

scripts is the destination directory, and

destination-file (1-255 characters) is a name you choose for the running-config file.

For example, the following command sequence exits to priv-exec mode, copies the local running configuration to the scripts directory, then shows the new script file:

```
bstnA(cfg)# exit
bstnA# copy running-config scripts running
bstnA# show scripts


  scripts
      running                 Sep 19 05:03  5.7k
```

Saving the Config Off to an FTP Server

To save the running config off to a remote FTP server, use a URL in place of the destination-file name. Follow the URL format below:

copy running-config ftp://[*username*:*password*@]*ftp-site*/*file*

where

***username*:*password*@** (optional) is an FTP username and password (the default is the username/password set by the ip ftp-user command, described above),

ftp-site identifies the FTP server with an IP address or FQDN (for example, “172.16.88.3” or “ftp.myftpsite.com”), and

file is the chosen file name. Lead with an extra “/” if the path starts at the root of the server machine; for example, “aramis//var/cfg/running-config” specifies “/var/config/running-config” on server aramis. Omit the leading slash if the file is going to the home directory for *username*.

The CLI prints a message with the results of the copy operation.

For example, the following command exits from cfg mode to priv-exec mode, then sends the running config to ftp.wwmed.com:

```
bstnA(cfg)# exit
bstnA# copy running-config ftp://juser:jpasswd@ftp.wwmed.com/oct24lcl

% INFO: Copy config file to destination file 'oct24running' completed successfully.

bstnA# ...
```

Saving the Config Off to an SCP Server

For secure sites, you can upload with the Secure Copy (SCP) protocol. The URL has a different syntax for SCP transfers:

copy running-config scp://*username*@*server*:*file* [accept-host-key]

where

***username*@** (optional) is a valid username at the remote host (the default is the username set by the ip scp-user command, described above),

server identifies the SCP server with an IP address or FQDN (for example, “172.16.100.18” or “deb1.mynet.com”), and

file is the chosen file name. Lead with a slash (*scp-server*:*/file*) if the file path is absolute. Without the slash, the path is presumed to start in the home directory for ***username***.

accept-host-key (optional) tells the CLI to accept an unknown host key if offered by the SCP server. The host key authenticates the server; if the key is unknown, it is possible that an attacker has taken the server’s hostname and/or IP address. Note that any SCP server is “unknown” if the switch has not had an SCP exchange with it since the switch’s last reboot.

The CLI prompts for the ***username***’s password, unless you set up a default with the ip scp-user command. If the prompt appears, enter a password that is valid at the remote site.

Another prompt shows the results of the copy operation.

For example, the following command exits from gbl mode to priv-exec mode, then sends the running config to rh1.wwmed.com. This goes to the default user’s home directory, where the default user was set with the ip scp-user command:

```
bstnA(gbl)# exit
bstnA# copy running-config scp://rh1.wwmed.com:oct24running

% INFO: Copy config file to destination file 'oct24running' completed successfully.

bstnA# ...
```

Placing the Config into an ARX Volume

You can also place the config file into an ARX volume. You can use the nfs or cifs clause to send the config file to a given directory in a given volume:

copy running-config {cifs|nfs} ***namespace volume dest-path***

where

cifs | nfs is a required choice. This is the network protocol used to transfer the config file to the ARX volume.

namespace (1-30 characters) identifies the destination namespace.

volume (1-1024 characters) is the destination-volume name.

dest-path (1-255 characters) is the intended path from the volume root (above) to the config file. The directory you specify here must exist on the volume.

The CLI prints a message with the results of the copy operation.

For example, the following command exits from gbl mode to priv-exec mode, then sends the running-config to a directory in the "medarcv~/rcrds" volume:

```
bstnA(cfg)# exit
bstnA# copy running-config cifs medarcv /rcrds admin/oct24running

% INFO: Copy config file to destination file 'oct24running' completed successfully.

bstnA# ...
```

Sending the Config to an E-Mail Recipient

You can alternatively send the running-config file as an E-mail attachment. Before you use E-mail, you must configure the Simple Mail Transfer Protocol (SMTP) on the switch, starting with the smtp command in cfg mode: see the chapter on E-mail and SMTP in the *ARX® CLI Reference*.

Use the following syntax to send the running-config file in an E-mail message:

copy running-config smtp://[*e-mail-address/***]*****file***

where

e-mail-address (optional) identifies the recipient of the E-mail message (for example, "myCoWorker@myco.com"). If you omit this, it defaults to the E-mail recipient set with the cfg-smtp to command. Use a slash (/) to separate this from the file name.

file is the chosen file name.

The CLI prints a message with the results of the copy operation.

For example, the following command sequence sets up SMTP, exits from cfg mode to priv-exec mode, then mails the running-config file to "juser@wwmed.com:"

```
bstnA(cfg)# smtp
bstnA(cfg-smtp)# mail-server email1.wwmed.com
bstnA(cfg-smtp)# from admin@acopia.wwmed.com
bstnA(cfg-smtp)# exit
bstnA(cfg)# exit
bstnA# copy running-config smtp://juser@wwmed.com/oct24running

% INFO: Copy config file to destination file 'oct24running' completed successfully.

bstnA# ...
```

Showing the Local Config

You can send the current local config to the screen without saving it to a file. Use the show running-config command to view all the CLI commands required to re-create the local running-config.

show running-config

For example:

```
bstnA> show running-config
; ARX-4000
; Version 6.01.000.14059 (Aug 12 2011 20:10:50) [nbuilds]
; Database version: 601000.106
; Generated running-config Thu Aug 18 02:22:14 2011
; System UUID  d9bdece8-9866-11d8-91e3-f48e42637d58
; ip private vlan internal 1010 metalog 1011 subnet 169.254.80.0 255.255.255.0
;
terminal character-set unicode-utf-8
;================================ vlan ================================
config
  vlan 25
    description "personnel dept."
    members 2/5 to 2/5
    exit

  vlan 25
    description rtTestVlan
    members 2/6 to 2/6
    exit

  exit

;========================== config-if-vlan ============================
config
  interface vlan 25
    ip address 192.168.25.5 255.255.255.0
    no shutdown
    exit

  interface vlan 25
    ip address 10.46.11.253 255.255.0.0
    no shutdown
    exit

...
```

Saving the Global Config

The next step in saving the running configuration is to save the global-config parameters. The *global config* is the part of the configuration that is shared among both ARXes in a redundant pair: this includes namespace and policy parameters. From priv-exec mode, use the copy global-config command to save the global config to an executable script file.

copy global-config scripts *destination-file*

where

scripts is the destination directory, and

destination-file (1-255 characters) is a name you choose for the global-config file.

For example, the following command sequence exits to priv-exec mode, copies the global running configuration, then shows the new script file:

```
bstnA(cfg)# exit
bstnA# copy global-config scripts global
bstnA# show scripts


  scripts
      global                 Sep 19 05:03  9.0k
      running                Sep 19 05:03  5.7k
```

Saving the Config Off to an FTP Server

To save the global config off to a remote FTP server, use an FTP URL as the destination:

copy global-config ftp://[*username*:*password*@]*ftp-site*/*file*

where

***username*:*password*@** (optional) is an FTP username and password (the default is the username/password set by the ip ftp-user command),

ftp-site identifies the FTP server with an IP address or FQDN (for example, “172.16.88.3” or “ftp.myftpsite.com”), and

file is the chosen file name. As with other FTP copies, use two slashes (*ftp-site*//*file*) if the file path is absolute.

The CLI prints a message to show the results of the copy operation.

For example, the following command exits from gbl mode to priv-exec mode, then sends the global config to /var/oct24gbl on ftp.wwmed.com:

```
bstnA(gbl)# exit
bstnA# copy global-config ftp://juser:jpasswd@ftp.wwmed.com//var/oct24gbl

% INFO: Copy config file to destination file 'oct24gbl' completed successfully.

bstnA# ...
```

Saving the Config Off to an SCP Server

As with the running-config, you can also use the Secure Copy (SCP) protocol to upload the global config:

copy global-config scp://*username@server:file* [accept-host-key]

where

***username*@** (optional) is a valid username at the remote host (the default is the username set by the ip scp-user command, described earlier),

server identifies the SCP server with an IP address or FQDN (for example, “172.16.100.12” or “host.mynet.com”), and

file is the chosen file name. Lead with a slash (*scp-server:/file*) if the file path is absolute. Without the slash, the path is presumed to start in the home directory for ***username***.

accept-host-key (optional) tells the CLI to accept an unknown host key if offered by the SCP server. The host key authenticates the server; if the key is unknown, it is possible that an attacker has taken the server’s hostname and/or IP address. Note that any SCP server is “unknown” if the switch has not had an SCP exchange with it since the switch’s last reboot.

The CLI prompts for the ***username***’s password, unless you set up a default with the ip scp-user command. If the prompt appears, enter a password that is valid at the remote site.

The CLI then shows the results of the copy operation.

For example, the following command exits from gbl mode to priv-exec mode, then sends the global config to the /var directory on rh1.wwmed.com. This defaults to whatever username and password were set earlier with the ip scp-user command:

```
bstnA(gbl)# exit
bstnA# copy global-config scp://rh1.wwmed.com:/var/oct24gbl

% INFO: Copy config file to destination file 'oct24gbl' completed successfully.

bstnA# ...
```

Placing the Config into an ARX Volume

You can also place the config file into an ARX volume. You can use the nfs or cifs clause to send the config file to a given directory in a given volume:

copy global-config {cifs|nfs} *namespace volume dest-path*

where

cifs | nfs is a required choice. This is the network protocol used to transfer the config file to the ARX volume.

namespace (1-30 characters) identifies the destination namespace.

volume (1-1024 characters) is the destination-volume name.

dest-path (1-255 characters) is the intended path from the volume root (above) to the config file. The directory you specify here must exist on the volume.

The CLI prints a message with the results of the copy operation.

For example, the following command exits from gbl mode to priv-exec mode, then sends the global-config to a directory in the "medarcv~/rcrds" volume:

```
bstnA(cfg)# exit
bstnA# copy global-config cifs medarcv /rcrds admin/oct24gbl

% INFO: Copy config file to destination file 'oct24gbl' completed successfully.

bstnA# ...
```

Sending the Config to an E-Mail Recipient

You can alternatively send the global-config file as an E-mail attachment, as shown above for the running-config. Before you use E-mail, you must configure the Simple Mail Transfer Protocol (SMTP) on the switch, starting with the smtp command in cfg mode: see the chapter on E-mail and SMTP in the *ARX® CLI Reference*.

Use the following syntax to send the global-config file in an E-mail message:

copy global-config smtp://[*e-mail-address*/]*file*

where

e-mail-address (optional) identifies the recipient of the E-mail message (for example, "myCoWorker@myco.com"). If you omit this, it defaults to the E-mail recipient set with the cfg-smtp to command. Use a slash (/) to separate this from the file name.

file is the chosen file name.

For example, the following command sequence sets up SMTP (including a destination E-mail address, "juser@wwmed.com"), exits from cfg mode to priv-exec mode, then mails the global-config file:

```
bstnA(cfg)# smtp
bstnA(cfg-smtp)# mail-server email1.wwmed.com
bstnA(cfg-smtp)# from admin@acopia.wwmed.com
bstnA(cfg-smtp)# to juser@wwmed.com
bstnA(cfg-smtp)# exit
bstnA(cfg)# exit
bstnA# copy global-config smtp://oct24gbl

% INFO: Copy config file to destination file 'oct24gbl' completed successfully.

bstnA# ...
```

Showing the Global Config

You can send the current global config to the screen without saving it to a file. Use the show global-config command to view all the CLI commands required to re-create the global config.

```
show global-config
```

The CLI commands appear in the proper order to be run as a CLI script.

◆ Note

Secure information, such as passwords and private SSH keys, is shown in encrypted form. Only the current ARX (or a replacement for it, in case of failure) can decrypt these passwords or keys.

For example:

```
bstnA> show global-config
; ARX-4000
; Version 6.04.000.15155 (Oct 29 2012 09:50:57) [nbuilds]
; Database version: 604000.2
; Generated global-config Fri Nov  9 02:37:18 2012
;
terminal character-set unicode-utf-8
;================================ global ================================
global
  kerberos health-check threshold 3500
  nfs tcp timeout 30
  smb2-allowed

;================================ user ================================
  user adm1 encrypted-password z4nEesIwajAbiyKh40VBO0R5xRbe83huxKCUzrypho0=
    exit

  user adm12 encrypted-password N4b37cIwajAbiyKh40VBO0R5xRbe83humjwfGQX1zu5srcqR4EtQlQ==
    exit

  user admin encrypted-password Ve5i7MIwajAbiyKh40VBO0R5xRbe83huVRIXiJOjaSw=
    exit

  user newadmin encrypted-password 4kQKwMIwajAbiyKh40VBO0R5xRbe83huA5tDm/T7PLsKKzPzxdPKSg==
    exit

;================================ group ================================
  group Administrators
    role backup-operator
    role crypto-officer
    role network-engineer
    role operator
    role storage-engineer
    windows-domain MEDARCH.ORG
    exit

...
bstnA>
```

Focusing On Specific Configuration Sections

You can show an individual group of commands from the show global-config output:

show global-config {filer | namespace | schedule | security | global-server | nfs | cifs}

where

- **filer** shows the “external-filer” section of the report,
- **namespace** shows the “namespace” section,
- **schedule** shows all policy schedules,
- **security** shows all sections related to security (such as “group,” “radius-server,” and “authentication”),
- **global-server** focuses on the “global-server” section,
- **nfs** shows the “nfs” section, and
- **cifs** shows the “cifs” section.

For example, the following command shows the security configuration:

```
bstnA> show global-config security
;================================ group ================================
;============================ radius-server ============================
;=========================== nfs-access-list ===========================
;============================ ntlm-auth-db =============================
;============================ proxy-user ===============================
;============================ win-mgmt-auth ============================
group Administrators
  role backup-operator
  role crypto-officer
  role network-engineer
  role operator
  role storage-engineer
  windows-domain MEDARCH.ORG
  exit

group "Backup Operators"
  role backup-operator
  role operator
  windows-domain MEDARCH.ORG
  exit

group "Domain Admins"
  role backup-operator
  role crypto-officer
  role network-engineer
  role operator
  role storage-engineer
  windows-domain MEDARCH.ORG
  exit

group "Domain Users"
  role operator
  windows-domain MEDARCH.ORG
  exit

group "Enterprise Admins"
  role backup-operator
```

```
  role crypto-officer
  role network-engineer
  role operator
  role storage-engineer
  windows-domain MEDARCH.ORG
  exit

group admins
  role storage-engineer
  user adm1
  user adm12
  exit

group crypto-officer
  user admin
  user newadmin
  exit

group operator
  user admin
  user newadmin
  user adm1
  user adm12
  exit

radius-server 192.168.25.201
  exit

radius-server 192.168.25.207
  auth-port 5555
  retries 4
  timeout 10
  exit

nfs-access-list eastcoast
  anonymous-gid 100
  anonymous-uid 100
  description "allowable subnets in MA, WELLS, & DC"
  nis domain wwmed.com
  permit 172.16.100.0 255.255.255.0 read-write root squash
  permit 172.16.204.0 255.255.255.0 read-only root allow
  permit 172.16.0.0 255.255.0.0 read-write root squash
  permit netgroup surgeons read-write root allow
  permit netgroup medtechs read-only root squash
  deny 192.168.77.0 255.255.255.0
  deny 192.168.202.0 255.255.255.0
  permit 192.168.98.0 255.255.255.0 read-write root allow
  permit 192.168.0.0 255.255.0.0 read-write root squash
  exit

nfs-access-list westcoast
  permit 172.209.3.0 255.255.255.0 read-write root squash
  permit 172.214.1.0 255.255.255.0 read-write root squash
  exit

ntlm-auth-db ntlmMap2
  user lab encrypted-password wjBqMBuLIqHjRUE7RHnFFt7zeG4G3VlRY/6AXKTgb2C2RqXz
  exit

proxy-user acoProxy1
  description "jq's admin account"
  user jqprivate encrypted-password +XuWQsIwajAbiyKh40VBO0R5xRbe83huOLWUc06bywZd/idlcoZRxg==
```

```
  windows-domain WWMEDNET.COM
  exit

proxy-user acoProxy3
  user jqtester encrypted-password mvBVvsIwajAbiyKh40VBO0R5xRbe83huk9fFfFJr4oQd2ncXpjBxEg==
  windows-domain FDTESTNET.COM pre-win2k-name BOSTONCIFS
  exit

proxy-user cifs_admin
  user Administrator encrypted-password Ve5i7MIwajAbiyKh40VBO0R5xRbe83huVRIXiJOjaSw=
  windows-domain MEDARCH.ORG
  exit

proxy-user emc_admin
  user nasadmin encrypted-password se9ur8IwajAbiyKh40VBO0R5xRbe83hueJcCBmZ8b66zbYw4QGIFXQ==
  exit

proxy-user nas_admin
  user root encrypted-password Ve5i7MIwajAbiyKh40VBO0R5xRbe83huVRIXiJOjaSw=
  exit

proxy-user nas_admin2
  user root encrypted-password Ve5i7MIwajAbiyKh40VBO0R5xRbe83huVRIXiJOjaSw=
  exit

proxy-user ny_admin
  user jqpublic encrypted-password ZvUdl8IwajAbiyKh40VBO0R5xRbe83huM/1a8HME+Lk=
  windows-domain WELLS.ME.ORG
  exit

proxy-user acoProxy2
  description "user with backup and admin creds on our servers"
  user jqpublic encrypted-password ZvUdl8IwajAbiyKh40VBO0R5xRbe83huM/1a8HME+Lk=
  windows-domain MEDARCH.ORG
  exit

windows-mgmt-auth fullAccess
  permit all any
  user juser windows-domain MEDARCH.ORG
  user jquser windows-domain MEDARCH.ORG
  exit

windows-mgmt-auth readOnly
  permit session monitor
  permit share monitor
  permit snapshot monitor
  user mhoward_md windows-domain MEDARCH.ORG
  user zmarx_cpa windows-domain MEDARCH.ORG
  user lfine_md windows-domain MEDARCH.ORG
  user choward_md windows-domain MEDARCH.ORG
  exit

windows-mgmt-auth snapViewers
  permit snapshot monitor
  user juser windows-domain MEDARCH.ORG
  user jquser windows-domain MEDARCH.ORG
  exit

bstnA>
```

Focusing On Named Configurations

You can show an individual namespace, volume, or front-end service by specifying the name at the end of the command:

show global-config namespace *name* [*volume*]

where

name (1-30 characters) identifies the namespace,

or

show global-config {nfs | cifs} *name*

where ***name*** (1-255 characters) is the fully-qualified domain name (FQDN) for the front-end service.

For example, the following command shows the configuration for the "medarcv" namespace:

```
bstnA> show global-config namespace medarcv
;====================== namespace managed volumes ======================
namespace medarcv
  protocol cifs
  cifs authentication kerberos
  cifs authentication ntlm
  cifs authentication ntlmv2
  cifs filer-signatures
  proxy-user acoProxy2
  windows-mgmt-auth readOnly
  windows-mgmt-auth fullAccess
  windows-mgmt-auth snapViewers
  sam-reference fs2
  volume /lab_equipment
    cifs access-based-enum auto-enable
    modify
    reimport-modify
    reserve files 4000000
    snapshot directory display all-exports
    snapshot privileged-access
    snapshot vss-mode none
    auto sync files
    metadata share nas1 nfs3 /vol/vol2/meta6
    no compressed-files
    named-streams
    no persistent-acls
    no sparse-files
    unicode-on-disk
    share backlots
      import sync-attributes
      policy freespace percent 3 resume-migrate 5
      filer fs2 cifs backlot_records
      enable
      exit

    share equip
      import priority 1

...

  exit

bstnA>
```

Saving Both Configs

The startup config is a combination of the running config and the global config. You can save the startup config as a single file. From priv-exec mode, use the copy startup-config command to save the startup config to an executable script file.

copy startup-config scripts *destination-file*

where

scripts is the destination directory, and

destination-file (1-255 characters) is a name you choose for the startup-config file.

For example, the following command sequence exits to priv-exec mode, copies the startup configuration, then shows the new script file:

```
bstnA(cfg)# exit
bstnA# copy startup-config scripts start_conf
bstnA# show scripts


  scripts
      global                  Sep 19 05:03  9.0k
      running                 Sep 19 05:03  5.7k
      start_conf              Sep 19 05:03   14k
```

Saving the Config Off to an FTP Server

Use a URL in the copy startup-config command to save the startup config to an FTP site:

copy startup-config ftp://[*username*:*password*@]*ftp-site*/*file*

where

***username*:*password*@** (optional) is an FTP username and password (the default is the username/password set by the ip ftp-user command),

ftp-site identifies the FTP server with an IP address or FQDN (for example, “172.16.88.3” or “ftp.myftpsite.com”), and

file is the chosen file name. As with other FTP copies, use two slashes (*ftp-site*//*file*) if the file path is absolute.

The CLI prints a message with the results of the copy operation.

For example, the following command exits from gbl mode to priv-exec mode, then sends the startup config to ftp.wwmed.com:

```
bstnA(gbl)# exit
bstnA# copy startup-config ftp://juser:jpasswd@ftp.wwmed.com/feb6startup

% INFO: Copy config file to destination file 'feb6startup' completed successfully.

bstnA# ...
```

Saving the Config Off to an SCP Server

Use an SCP-based URL to upload with the Secure Copy (SCP) protocol:

copy startup-config scp://*username@server:file* [accept-host-key]

where the options were defined for uploading the running-config (recall *Saving the Config Off to an SCP Server*, on page 5-6).

The CLI prompts for the ***username***'s password if there is no ip scp-user defined. If the password prompt appears, enter a password that is valid at the remote site. Then a message shows the results of the copy operation.

For example, the following command exits from gbl mode to priv-exec mode, then sends the startup config to the /var directory on rh1.wwmed.com:

```
bstnA(gbl)# exit
bstnA# copy startup-config scp://juser@rh1.wwmed.com:/var/feb6startup
Password: jpasswd

% INFO: Copy config file to destination file 'feb6startup' completed successfully.

bstnA# ...
```

Placing the Config into an ARX Volume

You can also place the config file into an ARX volume. You can use the nfs or cifs clause to send the config file to a given directory in a given volume:

copy startup-config {cifs|nfs} *namespace volume dest-path*

where

cifs | nfs is a required choice. This is the network protocol used to transfer the config file to the ARX volume.

namespace (1-30 characters) identifies the destination namespace.

volume (1-1024 characters) is the destination-volume name.

dest-path (1-255 characters) is the intended path from the volume root (above) to the config file. The directory you specify here must exist on the volume.

The CLI prints a message with the results of the copy operation.

For example, the following command exits from gbl mode to priv-exec mode, then sends the startup-config file to a directory in the "medarcv~/rcrds" volume:

```
bstnA(cfg)# exit
bstnA# copy startup-config cifs medarcv /rcrds admin/feb6startup

% INFO: Copy config file to destination file 'feb6startup' completed successfully.

bstnA# ...
```

Sending the Config to an E-Mail Recipient

You can alternatively send the startup-config file as an E-mail attachment, as shown above for the global-config and running-config. Before you use E-mail, you must configure the Simple Mail Transfer Protocol (SMTP) on the switch, starting with the smtp command in cfg mode: see the chapter on E-mail and SMTP in the *ARX® CLI Reference*.

Use the following syntax to send the startup-config file in an E-mail message:

copy startup-config smtp://[*e-mail-address*/]*file*

where the options were defined for mailing the running-config (recall *Sending the Config to an E-Mail Recipient*, on page 5-7).

For example, the following command sequence sets up SMTP (including a destination E-mail address, "juser@wwmed.com"), exits from cfg mode to priv-exec mode, then mails the startup-config file:

```
bstnA(cfg)# smtp
bstnA(cfg-smtp)# mail-server email1.wwmed.com
bstnA(cfg-smtp)# from admin@acopia.wwmed.com
bstnA(cfg-smtp)# to juser@wwmed.com
bstnA(cfg-smtp)# exit
bstnA(cfg)# exit
bstnA# copy startup-config smtp://feb6startup

% INFO: Copy config file to destination file 'feb6startup' completed successfully.

bstnA# ...
```

Prepared for Disaster Recovery

Once you have saved both the local and global running-configs (or the single startup-config), the ARX is prepared for disaster-recovery.

◆ Note

In addition, to prepare for replacing a single switch in the unlikely event of a failure, you should also save the following:

- *Master key (extracted and wrapped). Save it to a remote host along with the configs.*
- *Master key wrapping key password. Save it to a secure location.*

Restoring the Configuration

The first step to restoring the running config is to place the running-config script(s) onto the chassis. If the scripts are already on the switch (for example, if you saved the config(s) to the current chassis), you can skip this section. If the scripts are on an FTP server, enter priv-exec mode and use copy ftp to retrieve them:

copy ftp://[*username*:*password*@]*ftp-site*/*file* scripts *destination*

where

***username*:*password*@** (optional) is an FTP username and password (the default is the username/password set by the ip ftp-user command),

ftp-site identifies the FTP server with an IP address or FQDN (for example, “172.16.88.3” or “ftp.myftpsite.com”),

file is the script name at the server (lead with an extra “/” if the path is absolute),

scripts specifies the directory for the destination file, and

destination is the script name at the chassis.

For example, the following command sequence copies the startup config from ftp.wwmed.com:

```
bstnA> enable
bstnA# copy ftp://juser:jpasswd@ftp.wwmed.com/feb6startup scripts start_conf

% INFO: The copy command completed successfully.

bstnA# ...
```

Erasing the Current Configuration

You must remove the active “startup-config” file and reboot the switch before you restore the running-config or startup-config. For example:

```
bstnA> enable
bstnA# delete startup-config

Delete file 'startup-config' in directory 'configs'? [yes/no] yes
bstnA# reload
Reload the entire chassis? [yes/no] yes

Broadcast message (Fri Feb  6 15:06:47 2004):

The system is going down for reboot NOW!

...
```

Wait for the machine to reboot, then log back in through the serial CONSOLE port. (With the configuration erased, the MGMT port and all inband (VLAN) management interfaces are unavailable.) For example:

```
User Access Authentication

SWITCH login: admin
Password: password
System authenticating at device scope.
```

Restoring the Local and Global Configs

From priv-exec mode, use the run command to run each running-config script:

run scripts *script-name*

where ***script-name*** (1-255 characters) identifies the running-config script. Use show scripts for a list of available scripts.

Important

In a redundant pair, the global config is shared by both switches. If you restore the global config (either alone or as part of the startup config), you overwrite all global parameters for both switches in the pair. We recommend restoring the running-config first, rejoining the redundant pair, then restoring the global config, as outlined in the next section.

The CLI commands run on the command line until the configuration is finished.

For example, the following command sequence enters priv-exec mode, shows the saved scripts, and runs the “start_conf” script (which contains both the running and global configs):

```
SWITCH> enable
SWITCH# show scripts

scripts
      global                 Mar 16 01:59  6.5k
      running                Mar 16 01:59  3.0k
      start_conf             Mar 16 01:59  9.6k


SWITCH# run scripts start_conf
SWITCH# config
SWITCH(cfg)# hostname bstnA
```

```
bstnA(cfg)# ...
bstnA(gbl)# exit
bstnA# ...
```

Restoring Configs to a Redundant Pair

We recommend a two-phased approach for restoring configs to a redundant pair: restore the running-config to each switch, then restore the global-config on either switch. This synchronizes the pair before starting any namespace imports. Other orders are possible, but this is considered best practice.

Important

Do not restore the same running-config to both peers. The running-config includes the switch's private subnet, which must be unique to both peers. In fact, the private subnet must be unique in the switch's entire RON; see Resolving Conflicting Subnets, on page 6-13 of the ARX® CLI Network-Management Guide.

For instructions on joining a redundant pair, refer to *Enabling Redundancy*, on page 7-19 of the *ARX® CLI Network-Management Guide*.

Restoring the Running Configuration

For example, the following command sequence updates the running-config on both switches:

Peer A

```
SWITCH# run scripts running
SWITCH# config
SWITCH(cfg)# hostname prtlndA
prtlndA(cfg)# ...
prtlndA(cfg)# exit
prtlndA# ...
```

Peer B

```
SWITCH# run scripts running-B
SWITCH# config
SWITCH(cfg)# hostname prtlndB
prtlndB(cfg)# ...
prtlndB(cfg)# exit
prtlndB# ...
```

Restoring the Global Configuration

For example, the following command sequence updates one switch with the global-config:

Peer A

Wait for the peers to join. Use the show redundancy command: when both peers and the quorum disk are "Up," the pair is complete.

```
prtlndA(cfg-redundancy)# show redundancy
```

```
                                          Transitions
Node Switch/Quorum Disk   Status     Role    Total Last (UTC)
---- -------------------- ---------- ------- ----- -------------------
*1   prtlndA              Up         Active  Never -
 2   prtlndB              Up         Backup  1     05:33:19 09/14/2009
 QD  192.168.74.83        Up         Quorum  1     05:33:07 09/14/2009
prtlndA(cfg-redundancy)# ...
```

Exit to priv-exec mode and restore the global config:

```
prtlndA(cfg-redundancy)# end
prtlndA# run scripts global
prtlndA# global
...
prtlndA(gbl)# exit
prtlndA# ...
```

Consideration For Restoring to an ARX-2500 Pair

There is a special consideration you must take into account when restoring a configuration to an ARX-2500 pair, if the configuration changes the setting of the resource-profile command.

After executing the resource-profile command on both devices in the pair, you must execute the dual-reboot command to reboot both devices at once. This is true also if you replay a running-config script with the resource-profile legacy setting. (One method of replaying a running-config is to save the file on the ARX-2500 and use the run command.) After replaying the config script, you must reload the ARX-2500 for resource-profile legacy to take effect.

Restoring to One Peer

If you are only restoring the config on one switch, run its running-config script to set up networking and rejoin the switch with its waiting peer. The switch downloads the global config when it joins the redundant pair. If you want to replace the global-config too, run the global-config script on the peer (Senior) switch after the pair has joined.

For example, the following sequence restores the running-config and rejoins its peer. It does not restore the global config:

```
SWITCH# run scripts running-B
SWITCH# config
SWITCH(cfg)# hostname prtlndB
prtlndB(cfg)# ...
prtlndB(cfg)# exit
prtlndB# ...
```

6

Upgrading Software

Overview

ARX software is distributed in *release files*, typically named with a .rel extension (for example, “initial.rel” or “arx-0.5.0_dev.rel”). To upgrade software on the ARX, you must

1. clear space for a new release file (if necessary),
2. copy a new release file to the switch,
3. arm the switch with the release file, and
4. reboot the ARX.

This chapter describes this process in detail, then describes the process for upgrading software in a redundant pair of switches. The process for downgrading the software in a redundant pair is described as well, in the event that such action must be taken.

Downgrades

Downgrading is the process of changing to an older software release, where the release number is lower. The procedure for downgrading is similar to the procedure for upgrading, except that it deletes the configuration; therefore, you must take great care to preserve the configuration and edit it for the lower release before you begin. At the end of the downgrade, you must restore the configuration.

Important

Downgrades are not recommended. Consult with F5 Support before performing a downgrade.

In case a downgrade is necessary, there are instructions that apply to downgrades in this chapter. At the end of the chapter is a complete procedure for downgrading a redundant pair of ARX devices. Most of the chapter focuses on the upgrade scenario.

Before You Begin

Performance Considerations For ARX-1500 and ARX-2500

The ARX-1500 and ARX-2500 store their metalog data on their internal disks, along with logs, software-release files, and other management data. Managed volumes write their metalog data as clients change the volume state; the metalog is used to restore the volume configuration in the event of a failure. The metalog is also copied to the redundant peer. The speed of many volume operations depends on fast metalog writes.

Some other system operations create a large number of writes to the internal disk, potentially slowing metalog writes. This can slow volume performance, even if it occurs on the backup peer. For example, the process of upgrading the software release is extremely disk intensive, and may cause a noticeable performance degradation. During an upgrade, you use

- the copy command to copy a full release file to the disk, and
- the boot system command to unpack the release file on the disk.

You should perform such disk-intensive operations during off-peak hours on the ARX-1500 and ARX-2500. This is true whether you run the operations on the active peer or the backup.

Coordinating Upgrades Between Shadow-Volume Sites

Software upgrades may require more planning for multiple ARX sites. You can configure a *shadow-copy rule* to copy data from a managed volume on one ARX to a shadow volume on another ARX; for details, see *Chapter 16, Shadowing a Volume* in the *ARX® CLI Storage-Management Guide*. To upgrade both sites, we recommend pausing all policy operations at the source volume (see *Pausing All Rules in a Volume*, on page 14-50 of the same manual), upgrading software at the target site, and then upgrading software at the source site.

If the sites have redundant-ARX pairs, upgrade both peers at the target site (as described later in the chapter) before upgrading both peers at the source site.

Saving the Configuration

Back up your current configuration before initiating an upgrade or downgrade procedure. The configuration is at risk, especially in a downgrade; this makes it possible to reload the configuration later. Save the network-configuration parameters (running-config) in one file and the storage-configuration parameters (global-config) in another file. Use the copy running-config command to back up the running-config, as described in *Saving the Local Running Config*, on page 5-5. The copy global-config command saves the global-config, as described in *Saving the Global Config*, on page 5-9.

For example, this saves both config files to an external FTP server:

```
bstnA(cfg)# exit
bstnA# copy running-config ftp://juser:jpasswd@ftp.wwmed.com/net_config

% INFO: Copy config file to destination file 'net_config' completed successfully.

bstnA# copy global-config ftp://juser:jpasswd@ftp.wwmed.com/storage_config

% INFO: Copy config file to destination file 'storage_config' completed successfully.

bstnA# ...
```

Other Important Configuration Parameters

Record these configuration parameters, too. They are not part of the above config files, but they are crucial to restoring an ARX to service:

- the password for the administrative account (named “admin” by default) and
- the encrypted master key.

You can use the show master-key command to display the master key’s encrypted value. For example:

```
bstnA# show master-key
System Password: %uper$ecretpw
Wrapping Password: an0ther$ecretpw
Validate Wrapping Password: an0ther$ecretpw
Encrypted master key:
2oftVCwAAAAgAAAApwazSRFd2ww/H1pi7R7JMDZ9SoIg4WGA/XsZP+HcXjsIAAAADDRbMCxE/bc=
bstnA#
```

For detailed information about using this CLI command, refer to the *ARX® CLI Reference*.

Checking the Health of the Switch

After you copy your configuration, check the overall health of the ARX. Use the show health command to check for any alarm conditions:

```
bstnA# show health

System Health Information
Date                     ID      Event                       Description
--------------------------------------------------------------------------------
Wed Feb 27 04:12:02 2007 (0)   - No active alarms.

bstnA# ...
```

If any active alarms appear, contact F5 Support before proceeding.

Use the show cores command to confirm that no software processes failed and created a core-memory file for diagnosis:

```
bstnA# show cores


  cores

bstnA# ...
```

No core files appear in the above example. If any appear in your output, address the root cause of all of them before continuing. Contact F5 Support if you need assistance with diagnosis.

Checking the Log Files for Errors

Review the trap and error logs to see if any noteworthy events have occurred. Execute show logs traplog and show logs error.log as appropriate, and execute tail logs traplog and tail logs error.log to view the file contents, if necessary. See *Accessing the Syslog*, on page 8-7 for complete information about accessing and reading these files. Investigate and resolve any issues that come to light.

Review the syslog to see if any noteworthy system events have occurred. Execute show logs syslog and tail logs filename to list and view the syslog files. See *Accessing the Syslog*, on page 8-7 for complete information about accessing and reading these files. Investigate and resolve any issues that come to light.

Changing the Software Release

Clearing Space for a New Release File

Release files consume large amounts of disk space, so the ARX only stores up to three release files (possibly more on some platforms). If your switch has the maximum releases already, you must delete an unused release before you copy a new one onto the chassis. Use the show releases command for a listing of current release files:

```
show releases
```

This presents the files as a directory listing. Some files have single-letter codes (R, A, or B) that signify how the switch is using the release file:

- R is the *running* release.
- A is the *armed* release, the one that will run on the next reboot. By default, this is the same release as the running (R) release.
- B is the *backup* release. F5 personnel can revert the switch back to this release if necessary.

For example, the command below shows three releases on the current switch. The test1.rel file is currently running, and armed to run again on the next reboot. The test2.rel file is the backup, for disaster recovery. The other release file is unused, and is therefore a candidate for replacement:

```
bstnA(gbl-ns[wwmed])# show releases


  releases
  R A test1.rel              Dec  8 00:14  733M
  B   test2.rel              Dec  7 00:06  733M
      test3.rel              Dec  7 03:55  623M

bstnA(gbl-ns[wwmed])# ...
```

Showing the Software Versions

Use the show version command for a more-detailed view of the software versions:

```
show version
```

This shows the specific versions of each release and the module configuration for the chassis. For example:

```
bstnA(gbl-ns[wwmed])# show version
   Copyright (c) 2002-2012 by F5 Networks, Inc. All rights reserved.
Running Release
test1.rel :  Version 6.02.000.14342 (Mar 12 2012 20:07:15) [nbuilds]

Armed Release
test1.rel :  Version 6.02.000.14342 (Mar 12 2012 20:07:15) [nbuilds]

Backup Release
test2.rel :  Version 6.02.000.14334 (Feb 28 2012 17:47:10) [cdoyle]
```

```
System Configuration:  Version 602000.31

  bstnA uptime is 0 weeks, 0 days, 1 hours, 31 minutes.

  Slot  Admin     ModuleType     ModuleState        FW Upgrade
  ----  -------   ------------   ----------------   ----------
    1   Enabled   ACM            Online             Disabled
    2   Enabled   NSM            Online             Disabled


 Resource         State                              Forwarding
 --------         -------------------------------    ----------
  Switch           Up                                Disabled
bstnA(gbl-ns[wwmed])# ...
```

Deleting a Release File

You cannot delete the running, armed, or backup release. To delete any other release file, go to priv-exec mode and use the delete releases command:

delete releases *file*

where ***file*** (1-255 characters) identifies the release file. You can use wildcards (such as * for a string, ? for a single character, or [A-Z] for any uppercase letter).

For example, the following command sequence exits to priv-exec mode, deletes the test3.rel file, then shows the releases directory again to prove that it is gone:

```
bstnA(gbl-ns[wwmed])# end
bstnA# delete releases test1.rel
bstnA# show releases


  releases
  R A test1.rel              Dec  8 00:14  733M
  B   test2.rel              Dec  7 00:06  733M

bstnA# ...
```

Obtaining a New Release File

The next step in upgrading your software is to obtain a new release file from F5. Use a host machine that can access the F5 web site. Ideally, this host machine should be a client for an ARX CIFS or NFS service, or it should serve FTP or SCP.

Start by opening a new web browser and going to this address:

https://downloads.f5.com/esd/productlines.jsp

Unless you were recently logged into an F5 web site, you reach a login page. Enter your e-mail address (as a user name) and your password. If this is your first time accessing the F5 web site, click on “Register for an Account” to get a password.

You reach the Downloads page after the F5 site accepts your credentials. Navigate from this page to the desired ARX release, and follow the instructions to download the release file to the current machine. Release files are very large, so the download can take a long time.

If the current machine is a client for ARX-storage services, you can download the file to any CIFS or NFS service on the ARX. For example, if the machine has its F drive mapped to the "\\ac1.medarch.org\ARCHIVES" share on the ARX, you can download it to "F:\admin\12345.rel." If the current machine is an FTP server or supports SCP, download it to a directory that is accessible from the ARX.

If necessary, transfer the release file from this machine to another host that meets the above criteria.

Copying a Release File to the Switch

The next step in upgrading software is to copy the new release file to the ARX.

For cases where you copied the file into an ARX export or share (as described above), you can find the ARX volume that holds the release file and copy it from there to the "releases" directory. Other copy options, through FTP or SCP, are discussed in the subsections below.

The copy command requires the specific namespace and volume where the release file resides. Use show cifs-service *fqdn* for a map of *fqdn*'s front-end CIFS shares to its ARX volumes; see *Focusing on One CIFS Service*, on page 11-40 of the *ARX® CLI Storage-Management Guide*. For an NFS export, use show nfs-service *fqdn* (*Showing One NFS Service*, on page 11-8 of the same manual).

Then, from priv-exec mode, use copy cifs or copy nfs to copy the release file to the "releases" directory:

```
copy {cifs|nfs} namespace volume file-path releases dest-file
```

where

cifs | nfs is a required choice. This is the network protocol used to copy the release file out of the ARX volume.

namespace (1-30 characters) identifies the namespace. You can find the correct namespace from the output of show cifs-service or show nfs-service.

volume (1-1024 characters) is the volume name. For a CIFS share, use the output of show cifs-service to find the volume name (such as "/rcrds") behind a share name (such as "ARCHIVES"). For an NFS export, use the output of show nfs-service to find the volume name.

file-path (1-255 characters) is the path from the volume root (above) to the release file.

releases is the destination directory.

dest-file (1-255 characters) is the name you choose for the local copy of the release file.

For example, the following command sequence shows the namespace and volume behind "\\ac1.medarch.org\ARCHIVES" (highlighted in bold text below), copies the release file from that volume, then uses show releases to confirm that the file was copied:

```
bstnA# show cifs-service ac1.medarch.org

Service Name:          ac1.MEDARCH.ORG
Domain Join:           Joined to MEDARCH.ORG
Account Name:          ac1$
Delegation:            Constrained, Any Protocol
Delegate To:           cifs/PV770N
               cifs/PV770N.MEDARCH.ORG
               cifs/VM-PV770N-01
               cifs/VM-PV770N-01.MEDARCH.ORG
               cifs/VM-SWP2003S1-5
               cifs/VM-SWP2003S1-5.MEDARCH.ORG
               cifs/VM-SWP2003S2-04
               cifs/VM-SWP2003s2-04.MEDARCH.ORG
               cifs/VM-SWP2008S-01
               cifs/VM-SWP2008S-02
               cifs/VM-SWP2008s-01.MEDARCH.ORG
               cifs/VM-SWP2008s-02.MEDARCH.ORG
               cifs/engdm
               cifs/engdm.MEDARCH.ORG
               cifs/enterprise
               cifs/enterprise.wwmed.com
               cifs/ntap-prov
               cifs/ntap-prov.MEDARCH.ORG
               cifs/ntap820
               cifs/ntap820.MEDARCH.ORG
Description:           insurance-claim records
State:                 Enabled
Signatures:            Enabled
WINS Name Encoding:    ISO-8859-1

Exports for Namespace: insur

  Share Name                       Volume Path                          State
  -------------------------------- - ------------------------------------ -------
  CLAIMS                           /claims                              Online
  SPECS                            /claims/specs                        Online
  STATS                            /claims/stats                        Online

Exports for Namespace: medarcv

  Share Name                       Volume Path                          State
  -------------------------------- - ------------------------------------ -------
  ARCHIVES                         /rcrds                               Online
...

bstnA# copy cifs medarcv /rcrds admin/12345.rel releases test5.rel

% INFO: Copying 1050 megabytes from the specified source . . .

...

% INFO: Transferred 742 of 1050 megabytes; still copying . . .
```

```
% INFO: The copy source file '12345.rel' to destination file 'test5.rel' completed successfully.

bstnA# show releases


  releases
  R A test1.rel              Dec  8 00:14  733M
  B   test2.rel              Dec  7 00:06  733M
      test5.rel              Sep 10 00:09  794M

bstnA# ...
```

Downloading From an FTP Server

If the client machine supports FTP transfers to the ARX, you can use copy ftp instead:

copy ftp://[*username*:*password*@]*ftp-site*/*file* releases *dest-file*

where

***username*:*password*@** (optional) is an FTP username and password (the default is the username/password set by the ip ftp-user command; see *Setting a Default FTP or SCP User*, on page 5-3),

ftp-site identifies the FTP server with an IP address or FQDN (for example, “172.16.88.3” or “ftp.myftpsite.com”),

file is the chosen file name (lead with two slashes (*ftp-site*//*file*) if the file path is absolute),

releases is the destination directory on the ARX, and

dest-file is the name you choose for the local copy of the release file.

For example, the following command sequence copies a release file from mysrv.wwmed.com, then uses show releases to confirm that the file was copied:

```
bstnA# copy ftp://jusr:jpasswd@mysrv.wwmed.com/12345.rel releases test5.rel

% INFO: Copying 1050 megabytes from the specified source . . .

...

% INFO: The copy completed successfully.

bstnA# show releases


  releases
  R A test1.rel              Dec  8 00:14  733M
  B   test2.rel              Dec  7 00:06  733M
      test5.rel              Sep 10 00:09  794M

bstnA# ...
```

Downloading From an SCP Server

For secure sites, you can download the release file with the Secure Copy (SCP) protocol. The URL has a different syntax for SCP transfers:

copy scp://*username@server*:*file* releases *dest-file* [accept-host-key]

where

***username*@** is a valid username at the remote host,

server identifies the SCP server with an IP address or FQDN (for example, “172.16.100.18” or “deb1.mynet.com”), and

file is the release-file name. Lead with a slash (*scp-server:/file*) if the file path is absolute. Without the slash, the path is presumed to start in the home directory for ***username***.

dest-file is the name you choose for the downloaded copy of the release file.

accept-host-key (optional) tells the CLI to accept an unknown host key if offered by the SCP server. The host key authenticates the server; if the key is unknown, it is possible that an attacker has taken the server’s hostname and/or IP address. Note that any SCP server is “unknown” if the ARX has not had an SCP exchange with it since the ARX’s last reboot.

The CLI prompts for the ***username***’s password. Enter a password that is valid at the remote site. For example, the following command exits from gbl mode to priv-exec mode, then downloads a release file through SCP:

```
bstnA(gbl)# exit
bstnA# copy scp://juser@rh1.wwmed.com:/var/rels/12345.rel releases test.rel
Password: jpasswd

% INFO: Copying 1050 megabytes from the specified source . . .

...

bstnA# ...
```

Showing the Version of a Release File

Use the show releases command on a release file to view the release version and build date for the file:

show releases *file*

where ***file*** (1-255 characters) is the release-file name.

This shows the version number, time/date when the release was built, and the user that performed the build (typically, “nbuilds”). For example:

```
bstnA# show releases test5.rel
Version 6.03.000.14767 (Jul 27 2012 20:18:27) [nbuilds]
bstnA# ...
```

Validating the Release File

It is possible for the release file to be incomplete after it downloads. You can verify the integrity of the file by comparing its current *checksum*, a file signature based on its contents, against the checksum it had when it was built. Use the verbose option in the show releases command to run a checksum test on the release file:

```
show releases file verbose
```

A successful checksum shows as "Passed" in the output. For example:

```
bstnA# show releases test5.rel verbose
Version 6.03.000.14767 (Jul 27 2012 20:18:27) [nbuilds]
Checksum: Passed
bstnA# ...
```

If the checksum fails, retry the copy command and/or retry the download from the F5 web site.

Arming the Switch with the Release File

The next step in upgrading software is to *arm* the switch with the new release file. From priv-exec mode, use the boot system command to arm the ARX:

```
boot system release-file
```

where ***release-file*** (1-255 characters) identifies the desired release file.

On a busy system, this can take more than one minute.

◆ Note

If you are downgrading to an earlier release, this command removes all *of your configuration parameters. After the next reboot, all managed volumes will disappear from the configuration; to recover, you must restore the running-config and global-config files (saved earlier) to re-import all managed volumes. The CLI warns of this and prompts for confirmation; confirm that you have saved your config and enter* ***yes*** *to continue.*

As an upgrade example, the following command sequence arms the switch with the "test5.rel" file and then verifies that the switch is properly armed:

```
bstnA# boot system test5.rel

% INFO: The boot system command may take up to 5 minutes to complete.

bstnA# show releases


  releases
  R   test1.rel              Dec  8 00:14  733M
      test2.rel              Dec  7 00:06  733M
  A B test5.rel              Sep 10 00:09  794M


bstnA# ...
```

Rebooting the Switch

The final step in upgrading software is to reboot the switch and activate the armed release file. From priv-exec mode, use the reload command to reboot the switch:

```
reload
```

The CLI prompts for confirmation before rebooting; enter **yes** to proceed. For example:

```
bstnA# reload

Reload the entire chassis? [yes/no/diags] yes

System is resetting.
...
```

Activating the Software License (If Necessary)

If you have ever activated the license for this ARX, you can skip this subsection.

When a user logs in to the ARX for the first time after license-aware software is loaded, the message of the day announces that the ARX is not licensed. The ARX also sends an SNMP trap to this effect.

Contact your Sales representative to get your base-registration key, which is required to activate the license for the first time. The key usually comes in an E-mail message. Then use automatic or manual license activation, as appropriate for your installation. For details on license activation, see *Chapter 5, ARX Feature Licensing* in the *ARX® CLI Network-Management Guide*.

For example:

```
...

User Access Authentication

Username: admin
Password: password

...

bstnA> enable
bstnA# ping license-server base-reg-key CRJGVQP-DYWST-ANKR-GBYYDMT
% INFO: Activation server response: 'Thu Aug 25 04:28:00 UTC 2011'

bstnA# license activate base-reg-key CRJGVQP-DYWST-ANKR-GBYYDMT
% INFO: The license has been successfully activated.

bstnA#
```

If the license activation fails and cannot be resolved, you must arm the ARX with the former release and use reload to go back to it (go back to *Arming the Switch with the Release File*, on page 6-13).

Rebuilding the Configuration (Downgrade Only)

This section applies to a downgrade, where the new release has a lower number than the former release. Skip to the next section if you are upgrading to a higher release.

When the downgraded switch returned from its reboot and you logged back in, a notice informed you that the running-config has been reset to factory defaults. For example:

```
...

User Access Authentication

Username: admin
Password: password

********************** NOTICE ********************
The system detected an incompatible future version
of its configuration-database file(s).
The following file(s) have been automatically backed up.

        complete database:  omDb.complete.copy.Dec03_1534

The current running-config has been reset to factory
defaults, plus the minimal configuration obtained during
system initialization.

If the software on this system has NOT recently been
upgraded or downgraded, we advise you to restore the
running-config and/or global-config from a recent backup.
Otherwise, please use individual CLI commands to upgrade
or downgrade in smaller steps.
********************** NOTICE **********************

SWITCH> ...
```

This indicates that you need to restore the running-config and the global-config, saved off at the beginning of this process (recall *Saving the Configuration*, on page 6-19). First, edit both configuration files to remove all commands that do not exist in the older release. Contact F5 Support if you need assistance.

Once the running-config and global-config files are edited for the downgrade release, use the copy command to copy the files to the ARX. Then use the run configs command to invoke each of them as a script. Run the running-config script first, followed by the global-config script. For example, the following commands replace the previously stored configs:

```
SWITCH> enable
SWITCH# copy ftp://juser:jpasswd@ftp.wwmed.com/net_config configs running-config
SWITCH# run configs running-config
...
bstnA# copy ftp://juser:jpasswd@ftp.wwmed.com/storage_config configs global-config
bstnA# run configs global-config
...
```

Verifying the Installation

The new release is running when the switch reboots. Log back in and use show version to verify that the software upgrade was successful. For example:

```
...

User Access Authentication

Username: admin
Password: acopia
bstnA> show version
 Copyright (c) 2002-2012 by F5 Networks, Inc. All rights reserved.
Running Release
test1.rel :  Version 6.03.000.14767 (Jul 27 2012 20:18:27) [nbuilds]

Armed Release
test1.rel :  Version 6.02.000.14342 (Mar 12 2012 20:07:15) [nbuilds]

Backup Release
test2.rel :  Version 6.02.000.14334 (Feb 28 2012 17:47:10) [cdoyle]

System Configuration:  Version 603000.2

  bstnA uptime is 0 weeks, 0 days, 1 hours, 31 minutes.

  Slot  Admin     ModuleType     ModuleState        FW Upgrade
  ----  -------   ------------   ----------------   ----------
    1   Enabled   ACM            Online             Disabled
    2   Enabled   NSM            Online             Disabled


 Resource         State                            Forwarding
 --------         -------------------------------  ----------
  Switch           Up                              Disabled
bstnA> ...
```

Checking for New Firmware

Firmware upgrade and downgrade is not supported for ARX-VE. Skip to the next section if you are changing the release on that platform.

A software release may contain new firmware for the switch's hardware modules. The *firmware* is low-level software that controls the boot process and various FPGAs.

You can use the show firmware upgrade command to compare the running firmware with the firmware that is available on the new release:

```
show firmware upgrade [verbose]
```

where **verbose** (optional) expands the output with specific version numbers for each firmware component.

The non-verbose output shows a table of slots with an indication of which have a firmware upgrade available. For example, the following chassis has current firmware for all of its slots:

```
bstnA# show firmware upgrade
Show Firmware Update
--------------------

Slot Status Summary
-------------------

1    Up to date
2    Up to date
bstnA# ...
```

Upgrading Firmware

If the show firmware upgrade output indicates that a firmware upgrade is available for any of your slots, you can use the firmware upgrade all command to upgrade all slots at once.

Important

This takes several minutes and causes up to three automatic reboots. Contact F5 Support before upgrading firmware.

Run this command from priv-exec mode:

```
firmware upgrade all
```

The CLI prompts for confirmation before upgrading any firmware. Enter **yes** to proceed.

For example:

```
bstnA# firmware upgrade all

Confirmation of this command commences a firmware upgrade on the
entire chassis. During the upgrade process, the chassis reboots
automatically to complete the upgrade process. If this includes a bios
upgrade, this could take at least 30 minutes.
Proceed? [yes/no] yes
...
```

Sanity Check

When the upgraded switch returns from the reboot, log in and check the basic health of the system:

- use show processors to confirm that the processors are all “Up” or (for network processors) in “Standby” state,
- use show health to verify there are no active alarms, and

If any issues arise, contact F5 Support before proceeding further.

Upgrading a Redundant Pair

Redundancy offers an opportunity to test new software before putting it into service. You can coordinate the software upgrades by upgrading the backup switch first, testing it there, then deciding whether or not to upgrade its peer.

◆ Note

Redundant, high-availability support for the ARX-VE is available via standard hypervisor clustering functionality. These instructions for upgrading a redundant ARX pair do not apply to the ARX-VE.

Saving the Configuration

To prepare for a possible upgrade failure, save the running-config of the Backup switch. These configuration parameters will be invaluable if you decide to revert to the old release after the upgrade.

For example, the following command copies the running-config and the global-config to an FTP server at ftp.wwmed.com:

```
prtlndB# copy running-config ftp://juser:jpasswd@ftp.wwmed.com/runConfPrtlndB

% INFO: The copy command completed successfully.

prtlndB# copy global-config ftp://juser:jpasswd@ftp.wwmed.com/gbl_config_prtlnd

% INFO: The copy command completed successfully.

prtlndB# ...
```

Then record these configuration parameters:

- the password for the administrative account (named “admin” by default) and
- the encrypted master key.

Refer back to *Other Important Configuration Parameters*, on page 6-5 for more details and examples.

Verifying the Active Switch’s Ability to Stand Alone

Before you upgrade one of the switches, confirm that the chosen Active switch (the switch that you will *not* upgrade first) is connected to the quorum disk. The Active switch must be connected to the quorum disk when you upgrade its peer, or both switches reboot simultaneously. Use the show redundancy quorum-disk command to verify the connection to the quorum disk (see *Showing the Quorum Disk*, on page 7-22 of the *ARX® CLI*

Network-Management Guide). Run this command at the *Active* switch, not the switch where you will do the upgrade first. Any "Up" status, even if the switch is not receiving heartbeats, is acceptable.

Important

Verify that the quorum disk is fast and reliable, too. Quorum-disk transitions should be very rare, and latency should be typically low. If the transition count is high, the last transition is recent, or the latency is high, you may want to choose a faster/more-reliable quorum disk before you perform the upgrade. Select another disk by changing it on both peers, one after the other.

For example, the "prtlndA" is ready to take Active service during an upgrade:

```
prtlndA# show redundancy quorum-disk

Path:             192.168.74.83:/exports/quorum-disk/portland1
Protocol:         nfs2
Status:           Up

Heartbeats
  Sent:           744
  Received:       741

Transitions
  Count:          1
  Last:           05:33:07 09/14/2009
  Reason:         Quorum disk 192.168.74.83:/exports/quorum-disk/portland1 is now online.

Heartbeat Latency:
                        Heartbeat Latency Intervals (msec)
  Time Interval    [0-499]   [500-999]   [1000-3999]   [No Response]
  -------------    ------------------------------------------------
  01:00 - 01:45       745           0             0               0
  00:00 - 01:00         0           0             0               0
  23:00 - 24:00         0           0             0               0
  22:00 - 23:00         0           0             0               0
  21:00 - 22:00         0           0             0               0
  20:00 - 21:00         0           0             0               0
  19:00 - 20:00         0           0             0               0
  18:00 - 19:00         0           0             0               0
  17:00 - 18:00         0           0             0               0
  16:00 - 17:00         0           0             0               0
  15:00 - 16:00         0           0             0               0
  14:00 - 15:00         0           0             0               0
  13:00 - 14:00         0           0             0               0
  12:00 - 13:00         0           0             0               0
  11:00 - 12:00         0           0             0               0
  10:00 - 11:00         0           0             0               0
  09:00 - 10:00         0           0             0               0
  08:00 - 09:00         0           0             0               0
  07:00 - 08:00         0           0             0               0
  06:00 - 07:00         0           0             0               0
  05:00 - 06:00         0           0             0               0
  04:00 - 05:00         0           0             0               0
  03:00 - 04:00         0           0             0               0
  02:00 - 03:00         0           0             0               0
```

```
Heartbeat latency summary:
  0-499msec            : 100.00%
  500-999 msec         :   0.00%
  1000-3999 msec       :   0.00%
  No response          :   0.00%
prtlndA# ...
```

Checking the Health of the Backup Switch

From the backup switch, use the show redundancy command to confirm that the redundant pair is formed. Both peers and the quorum disk should all have a status of "Up." For example,

```
prtlndB# show redundancy

                                          Transitions
Node Switch/Quorum Disk   Status     Role    Total Last (UTC)
---- -------------------- ---------- ------- ----- -------------------
 1   prtlndA              Up         Active  1     05:33:19 09/14/2009
*2   prtlndB              Up         Backup  Never -
 QD  192.168.74.83        Up         Quorum  1     05:33:07 09/14/2009
prtlndB# ...
```

If the status is not "Up" for any of the nodes, contact F5 Support before proceeding.

Otherwise, use the show health command to check for any alarm conditions:

```
prtlndB# show health

System Health Information
Date                     ID      Event                       Description
-------------------------------------------------------------------------------
Wed Feb 28 04:45:02 2007 (0)   - No active alarms.

prtlndB# ...
```

If any active alarms appear, contact F5 Support before proceeding.

Use the show cores command to confirm that no software processes failed and created a core-memory file for diagnosis:

```
prtlndB# show cores


  cores

prtlndB# ...
```

No core files appear in the above example. If any appear in your output, address the root cause of all of them before continuing.

Upgrading the Backup Switch

The next step is to upgrade the Backup switch as described earlier. For example, the following command sequence downloads a new software release, new.rel, arms the switch with the new release, and reloads the system:

```
prtlndB# copy ftp://jusr:jpasswd@mysrv.wwmed.com/12345.rel releases new.rel
```

```
% INFO: Copying 1050 megabytes from the specified source . . .

...

% INFO: Transferred 742 of 1050 megabytes; still copying . . .

% INFO: The copy source file '12345.rel' to destination file 'new.rel' completed successfully.

prtlndB# boot system new.rel

% INFO: The boot system command may take up to 5 minutes to complete.

prtlndB# reload

Reload the entire chassis? [yes/no/diags] yes

System is resetting.
...
```

Activating the License (If Necessary)

If you have ever activated the license for this ARX, you can skip this subsection.

When the ARX is running for the first time with license-aware software, the software license must be activated. A message on the system console announces this when you log in, as does an SNMP trap.

Contact your Sales representative to get your base-registration key, which is required to activate the license for the first time. The key usually comes in an E-mail message. Then use automatic or manual license activation, as appropriate for your installation. For details on license activation, see *Chapter 5, ARX Feature Licensing* in the *ARX® CLI Network-Management Guide*.

For example, this command sequence logs in after the reboot, pings the ARX license server, and then performs an automatic activation:

```
...

User Access Authentication

Username: admin
Password: password

...

prtlndB# ping license-server base-reg-key CYBJAAZ-DYWST-ANKR-GBYYDMT
% INFO: Activation server response: 'Thu Aug 25 04:28:00 UTC 2011'

prtlndB# license activate base-reg-key CYBJAAZ-DYWST-ANKR-GBYYDMT
% INFO: The license has been successfully activated.

prtlndB#
```

If the license activation fails and cannot be resolved, you must reload and boot the former release of software (see *Upgrading the Backup Switch*, on page 6-21).

Checking for New Firmware

Firmware upgrade and downgrade is not supported for ARX-VE. Skip to the next section if you are changing the release on that platform.

If this platform supports firmware upgrades, use the show firmware upgrade command to compare the running firmware with the firmware that is available on the new release. For example, the following chassis has new firmware available for its only slot:

```
prtlndB# show firmware upgrade
Show Firmware Update
--------------------

Slot Status Summary
-------------------

1    Upgrade available
prtlndB# ...
```

Upgrading Firmware

If the show firmware upgrade output indicates that a firmware upgrade is available for any of your slots, you can use the firmware upgrade all command to upgrade all slots at once.

Important

This takes several minutes and causes an automatic reboot. Contact F5 Support before upgrading firmware.

For example:

```
prtlndB# firmware upgrade all

Confirmation of this command commences a firmware upgrade on the
entire chassis. During the upgrade process, the chassis reboots
automatically to complete the upgrade process. If this includes a bios
upgrade, this could take at least 30 minutes.

Proceed? [yes/no] yes
...
```

Sanity Check

When the upgraded switch returns from the reboot, log in and check the basic health of the system. This switch is still in the backup role:

- use show processors to confirm that the processors are all “Up” or (for network processors) in “Standby” state,
- use show health to verify there are no active alarms, and
- use show redundancy to ensure that redundancy is up and running.

If any issues arise, contact F5 Support before proceeding further.

Checking the Health of the Active Switch

Repeat the health checks at the active switch, where the previous release is still running. Use the show redundancy command to confirm that the redundant pair is formed, and use show health to verify that there are no active alarms. Both peers and the quorum disk should all have a status of "Up," and there should be no active alarms.

If the status is not "Up" for any of the nodes, or if any active alarms appear, contact F5 Support before proceeding.

Failing Over to the Upgraded Switch

The next step is to fail over and put the new software into active service. Reload the other switch, the switch currently running services on the old software. Use the priv-exec reload command. This forces failover to the switch running the new software. From the now-active switch ("prtlndB" in our examples), take the following steps to confirm that the failover succeeded:

- Repeat show cifs-service all and/or show nfs-service all until all of your configured services have a "State" of "Online."
- From a client machine, test all NFS exports and/or CIFS shares on the ARX.
- Repeat show redundancy until the now-backup peer has re-joined the pair in the "Backup" role. Confirm that the both peers and the quorum disk all have a "Status" of "Up."

The failover should be complete and the above tests should pass after five minutes at most. Contact F5 Support if the failover takes more time or if any of the above tests fail.

Testing the New Release

The new release can process client traffic with the same configuration that was in use before the upgrade. This is an opportunity to verify the integrity of the new release with standard client traffic. However, none of the release's new global-mode features are available until after you upgrade both peers: the CLI does not allow you to enter gbl mode, and the GUI does not offer any storage-management or policy features.

From here, there are two paths to follow: the success path (you want to deploy the new software) or the failure path (you want to continue running the old software).

Success Path

Upgrade the peer switch (now in the Backup role) to the new release. Use the same steps that you used above, from *Saving the Configuration*, on page 6-19 to *Sanity Check*, on page 6-23. Then both switches are running the new software, and all new global-configuration options are now available.

You must invoke one additional failover after both peers have been upgraded. The final failover, with both peers at the new software version, makes it possible to perform the final upgrade of internal databases and enable any new features that are database-dependent. To return the originally-active peer ("prtlndA" in our examples) to its active status, use reload on the currently-active peer ("prtlndB"). This causes a failover.

Failure Path

To return to the old release, you can revert the now-Active switch to the old software. First use reload to fail back to the peer switch, which is still running the previous release. When this switch boots (now in the backup role), use boot system to select the old release, then use reload to install it. Downgrading deletes all configuration parameters, both the running-config and the global-config; a CLI prompt asks for confirmation before doing this. Recall that you saved the global-config and running-config files earlier, in *Saving the Configuration*, on page 6-19.

For example:

```
prtlndB# reload
Reload the entire chassis? [yes/no] yes

Broadcast message (Thu Sep 10 13:27:12 2009):

The system is going down for reboot NOW!
...

prtlndB# boot system oldRelease.rel

WARNING: Arming this release will cause your configuration to be reset to factory default because
it is a lower version than the running release.

Proceed? [yes/no] yes

% INFO: The boot system command may take up to 5 minutes to complete.

prtlndB# reload

Reload the entire chassis? [yes/no/diags] yes

System is resetting.
...
```

When the downgraded switch returns from its reboot and you log back in, a notice informs you that the running-config has been reset to factory defaults. For example:

```
...

User Access Authentication
```

```
Username: admin
Password: password

********************** NOTICE ********************
The system detected an incompatible future version
of its configuration-database file(s).
The following file(s) have been automatically backed up.

        complete database:  omDb.complete.copy.Dec03_1534

The current running-config has been reset to factory
defaults, plus the minimal configuration obtained during
system initialization.

If the software on this system has NOT recently been
upgraded or downgraded, we advise you to restore the
running-config and/or global-config from a recent backup.
Otherwise, please use individual CLI commands to upgrade
or downgrade in smaller steps.
********************** NOTICE **********************

SWITCH> ...
```

Restoring the Old Configuration

The final step in reverting to the old software is to restore the old running-config parameters, saved off at the beginning of this process (recall *Saving the Configuration*, on page 6-19). The global-config is still running on the active peer (prtlndA in the examples), so that does not need to be restored. Use the copy command to copy the file to the ARX. Then use the run configs running-config command to invoke it as a script.

For example, the following commands replace the previously stored running-config:

```
SWITCH> enable
SWITCH# copy ftp://juser:jpasswd@ftp.wwmed.com/runConfPrtlndB configs running-config

Overwrite existing file 'running-config' in directory 'configs'? [yes/no] yes

% INFO: Transferred 0 megabytes; still copying . . .


% INFO: The copy of source file 'runConfPrtlndB' to destination file 'running-config' completed
successfully.

SWITCH# run configs running-config
SWITCH- config
SWITCH(cfg)- vlan 405
...
prtlndB(cfg)#
```

Downgrading a Redundant Pair

Downgrades are against best practices, but may be necessary in some rare cases. This causes a temporary service outage. We recommend that you perform this procedure during off-peak hours, and only under the guidance of F5 Support.

◆ Note

Redundant, high-availability support for the ARX-VE is available via standard hypervisor clustering functionality. These instructions for downgrading a redundant ARX pair do not apply to the ARX-VE.

Saving the Configuration

The system removes the entire configuration database as part of a downgrade. To prepare for this, save the running-config (network parameters) for both peers, and save the global-config (storage parameters, shared between the peers) from either of them. The global-config is shared, so it does not matter which peer you save it from.

For example, the following command copies the running-config and the global-config from the backup peer, "prtlndB," to an FTP server at ftp.wwmed.com:

```
prtlndB# copy running-config ftp://juser:jpasswd@ftp.wwmed.com/runConfPrtlndB

% INFO: The copy command completed successfully.

prtlndB# copy global-config ftp://juser:jpasswd@ftp.wwmed.com/gbl_config_prtlnd

% INFO: The copy command completed successfully.

prtlndB# ...
```

To continue the example, this command copies the running-config for the active peer, "prtlndA," to the same FTP server:

```
prtlndA# copy running-config ftp://juser:jpasswd@ftp.wwmed.com/runA

% INFO: The copy command completed successfully.

prtlndA# ...
```

Then record these configuration parameters:

- the password for the administrative account (named "admin" by default) and
- the encrypted master key.

These parameters are shared between the peers, like the global-config. Refer back to *Other Important Configuration Parameters*, on page 6-5 for more details and examples.

Editing the Config Files

The config files may have commands that are unknown in the downgrade release. Edit them all to remove any such commands. You can use the *CLI Reference* manual from the downgrade release to check for the existence and syntax of any given command in the release.

Additionally, remove the cfg-redundancy enable command from both running-config files. When you run the config files as scripts later, you will want to control the timing for re-enabling redundancy. For example, this shows the end of the running-config file for the "prtlndB" switch. Comment out or remove the enable command, shown here in bold text:

```
...
    exit

;========================== ntp server config ==========================
  ntp server 192.168.74.104 version 4

;========================== cfg-redundancy ============================
  redundancy
    peer 10.1.23.11
    quorum-disk 192.168.74.83:/exports/quorum-disk/portland1 nfs2
    critical route 0.0.0.0 0.0.0.0
    enable
    exit

  exit
```

Remove the same command(s) from the running-config for the active switch, too.

Downgrading the Backup Switch

The next step is to downgrade the Backup switch. This is the same as the upgrade process described earlier, except that you choose a release with a lower release number. Typically, the older release is already on the ARX. Use the boot system command to arm the system with the older release, and then reload with it. The boot system command causes the next reload to erase the entire configuration, so the CLI prompts you for confirmation first; this is necessary for the downgrade, so enter **yes** to proceed.

For example, the following command sequence arms the switch with an older release and reloads the system:

```
prtlndB# boot system old.rel

WARNING: Arming this release will cause your configuration to be reset to factory default because
it is a lower version than the running release.

Proceed? [yes/no] yes

% INFO: The boot system command may take up to 5 minutes to complete.

prtlndB# reload

Reload the entire chassis? [yes/no/diags] yes
```

```
System is resetting.

...
```

Rebuilding the Running-Config

When the downgraded switch returns from its reboot and you log back in, a notice informs you that the running-config has been reset to factory defaults. For example:

```
...

User Access Authentication

Username: admin
Password: password

********************** NOTICE ********************
The system detected an incompatible future version
of its configuration-database file(s).
The following file(s) have been automatically backed up.

        complete database:  omDb.complete.copy.Dec03_1534

The current running-config has been reset to factory
defaults, plus the minimal configuration obtained during
system initialization.

If the software on this system has NOT recently been
upgraded or downgraded, we advise you to restore the
running-config and/or global-config from a recent backup.
Otherwise, please use individual CLI commands to upgrade
or downgrade in smaller steps.
********************** NOTICE **********************

SWITCH> ...
```

This indicates that you need to restore the running-config, saved off and edited at the beginning of this process (recall *Saving the Configuration*, on page 6-27).

Once the running-config file is edited for the downgrade release (recall *Editing the Config Files*, on page 6-28), use the copy command to copy the file to the ARX. The CLI prompts for confirmation before overwriting the current running-config file; enter **yes** to proceed. Then use the run configs running-config command to invoke it as a script.

For example, the following commands replace the previously stored running-config:

```
SWITCH> enable
SWITCH# copy ftp://juser:jpasswd@ftp.wwmed.com/runConfPrtlndB configs running-config

Overwrite existing file 'running-config' in directory 'configs'? [yes/no] yes

% INFO: Transferred 0 megabytes; still copying . . .


% INFO: The copy of source file 'runConfPrtlndB' to destination file 'running-config' completed
successfully.
```

```
SWITCH# run configs running-config
SWITCH- config
SWITCH(cfg)- vlan 405
...
prtlndB#
```

Checking for New Firmware

Firmware downgrade is not supported for ARX-VE. Skip to the next section if you are changing the release on that platform.

If this platform supports firmware downgrades, use the show firmware upgrade command to compare the running firmware with the firmware that is available on the downgrade release. For example, the following chassis has new firmware available for its only slot:

```
prtlndB# show firmware upgrade
Show Firmware Update
--------------------

Slot Status Summary
-------------------
1    Upgrade available
prtlndB# ...
```

Upgrading Firmware

If the show firmware upgrade output indicates that a firmware change is available for any of your slots, you can use the firmware upgrade all command to change all slots at once.

Important

This takes several minutes and causes an automatic reboot. Contact F5 Support before downgrading firmware.

For example:

```
prtlndB# firmware upgrade all

Confirmation of this command commences a firmware upgrade on the
entire chassis. During the upgrade process, the chassis reboots
automatically to complete the upgrade process. If this includes a bios
upgrade, this could take at least 30 minutes.

Proceed? [yes/no] yes
...
```

Sanity Check

After checking the firmware on the downgraded switch, check the basic health of the system:

- use show processors to confirm that the processors are all “Up” or (for network processors) in “Standby” state, and
- use show health to verify there are no active alarms *other than* “licenseNotFoundRaise.” The license is supposed to be disabled at the moment because the peer ARX is still the active peer (and possibly in service), so it has registered all the proxy-IP addresses for this current ARX through ARP. Enabling the license would cause the current switch to gratuitously ARP all of its proxy IPs, creating contention for those addresses between the two ARX devices.

If any issues arise, contact F5 Support before proceeding further.

Downgrading the Active Switch

You must manually erase all configuration parameters from the active peer before you downgrade it. You cannot use the boot system command on an active peer, and the active peer cannot fail over if the redundant peer is running a lower release, so you must erase the device’s configuration before you perform the downgrade. Use the delete startup-config command, followed by a reload, to erase all of the configuration from the active peer. Recall that you already saved the configuration earlier, and edited it for use at the older release level.

Important

This causes a service outage. This outage is unavoidable for downgrades.

The CLI prompts for confirmation at each command, to warn you of the configuration outage and the loss of service; enter **yes** for both commands.

For example, the following command sequence removes the configuration from “prtlndA” and reloads the switch:

```
prtlndA# delete startup-config

Delete file 'startup-config' in directory 'configs'? [yes/no] yes
prtlndA# reload

You had previously deleted your startup-config.
If you would like to restore your configuration,
answer 'no' to cancel this reload and then run the
'restore startup-config' CLI command before
re-running this command. If you answer 'yes'
instead, the ARX returns to its factory defaults.

The redundant peer is unable to take over services at this time.
Reload? [yes/no/diags] yes

System is resetting.

...
```

Downgrading the Release

The next step is to downgrade the release. This is similar to downgrading the release on the backup switch. Typically, the older release is already on the ARX.

For example, the following command sequence arms the active switch with an older release and reloads the system:

```
...

User Access Authentication

Username: admin
Password: password



Running-config last saved 1/7 by J. User




ARX software must be licensed. Please configure DNS for automatic license activation.

SWITCH> enable
SWITCH# boot system old.rel

% INFO: The boot system command may take up to 5 minutes to complete.

SWITCH# reload

Reload the entire chassis? [yes/no/diags] yes

System is resetting.

...
```

Rebuilding the Running-Config

When the downgraded switch returns from its reboot, restore its running-config. Use the running-config that you saved off and edited at the beginning of this process (recall *Saving the Configuration*, on page 6-27).

Use the copy command to copy the edited running-config file to the ARX. The CLI prompts for confirmation before overwriting the current running-config file; enter **yes** to proceed. Then use the run configs running-config command to invoke it as a script.

The running-configs are unique to each switch; be careful to use the correct running-config file.

For example, the following commands replace the previously stored running-config for “prtlndA:”

```
User Access Authentication

Username: admin
Password: password
```

```
Running-config last saved 1/7 by J. User



ARX software must be licensed. Please configure DNS for automatic license activation.

SWITCH> enable
SWITCH# copy ftp://juser:jpasswd@ftp.wwmed.com/runA configs running-config

Overwrite existing file 'running-config' in directory 'configs'? [yes/no] yes

% INFO: Transferred 0 megabytes; still copying . . .


% INFO: The copy of source file 'runA' to destination file 'running-config' completed
successfully.

SWITCH# run configs running-config
SWITCH- config
SWITCH(cfg)- vlan 405
...
prtlndA#
```

Activating the License

The downgrade erases all traces of a former license activation along with the configuration database. If the license was activated previously, you can use the license activate command (without a base-registration key) to re-activate it after the downgrade. (For details on license activation, see *Chapter 5, ARX Feature Licensing* in the *ARX® CLI Network-Management Guide*.)

For example:

```
prtlndA# license activate
% INFO: The license has been successfully activated.

prtlndA#
```

Checking for New Firmware

Firmware downgrade is not supported for ARX-1500, ARX-2500, or ARX-VE. Skip to the next section if you are changing the release on any of these platforms.

If this platform supports firmware downgrades, use the show firmware upgrade command to compare the running firmware with the firmware that is available on the downgrade release. If a change is necessary, use the firmware upgrade all command. For example, the following chassis has all of its firmware up-to-date:

```
prtlndA# show firmware upgrade
Show Firmware Update
--------------------

Slot Status Summary
-------------------

1    Up to date
prtlndA# ...
```

Restoring the Global-Config and Restoring (Standalone) Service

The next step in reverting to the old software is to restore the old global-config parameters, saved off and edited at the beginning of this process (recall *Saving the Configuration*, on page 6-27). Use the copy command to copy the global-config file to the ARX. Then use the run configs global-config command to invoke it as a script.

This re-imports all managed volumes and restores all VIPs to service. Until later in the procedure, storage services are offered on a standalone ARX.

For example, the following commands replace the previously stored and edited global-config:

```
prtlndA# copy ftp://juser:jpasswd@ftp.wwmed.com/gbl_config_prtlnd configs global-config

Overwrite existing file 'global-config' in directory 'configs'? [yes/no] yes

% INFO: The copy of source file 'gbl_config_prtlnd' to destination file 'global-config' completed
successfully.

prtlndA# run configs global-config
prtlndB- global
prtlndB(gbl)- user adm1 encrypted-password wVHZxkyYpVFNLma0Pl/XtoqvYOYPtwhjXt2QkzyLfLc=
prtlndB(gbl-user[adm1])- exit
...
prtlndA#
```

Sanity Check

After re-running the global-config on the downgraded switch, check the basic health of the system:

- use show processors to confirm that the processors are all “Up” or (for network processors) in “Standby” state,
- use show health to verify there are no active alarms, and
- use show redundancy to ensure that redundancy is not currently running.

If any issues arise, contact F5 Support before proceeding further.

Activating the License on the Originally-Backup Peer

The originally-backup peer (prtlndB in the examples) still has a disabled license. It needs its license enabled before it can rejoin the redundant pair. If the license was ever activated previously, you can use the license activate command (without a base-registration key) to re-activate it after the downgrade. For details on license activation, see *Chapter 5, ARX Feature Licensing* in the *ARX® CLI Network-Management Guide*.

For example:

```
prtlndB# license activate
% INFO: The license has been successfully activated.

prtlndB#
```

Restoring Redundancy

The originally-active switch, “prtlndA” in the examples, is now running the downgrade release as a standalone switch. The originally-backup switch (“prtlndB”) is also running the downgrade release. Both have redundancy parameters configured but disabled. Both also have active licenses. They are ready to join as a redundant pair.

Log into each of them, enter cfg-redundancy mode, and use the enable command in that mode. For example:

Peer A

```
prtlndA# config
prtlndA(cfg)# redundancy
prtlndA(cfg-redundancy)# enable
prtlndA(cfg-redundancy)#
```

Peer B

```
prtlndB# config
prtlndB(cfg)# redundancy
prtlndB(cfg-redundancy)# enable

08-25 17:29:56 Peer switch 'prtlndA' is now online



prtlndB(cfg-redundancy)# show redundancy
```

```
                                                    Transitions
Node Switch/Quorum Disk   Status     Role    Total Last (UTC)
---- -------------------- ---------- ------- -----
-------------------
 1   prtlndA              Up         Active  1     14:46:03 08/29/2012
*2   prtlndB              Up         Backup  Never -
 QD  192.168.74.83        Up         Quorum  1     14:44:06 08/29/2012
prtlndB(cfg-redundancy)#
```

7

Metadata Utilities: nsck and sync

Overview

The Namespace-Check (*nsck*) and synchronization (*sync*) utilities manipulate the metadata in managed volumes. *Metadata* is high-level information such as physical-file locations on back-end filers, modification-time stamps, and file sizes. The nsck utility can show a volume's metadata to reveal which back-end shares hold its files and directories. If you suspect that the metadata is inconsistent with the back-end files, you can use nsck to re-examine the filers and compare the results with the metadata. Wherever nsck finds metadata inconsistencies, you have the opportunity to correct them.

The nsck and sync utilities offer several options for working with metadata:

- nsck *report* shows the current state of metadata,
- sync finds and corrects metadata inconsistencies,
- nsck *rebuild* rebuilds an entire namespace by re-importing all of its shares,
- nsck *destage* removes all metadata for a volume, share, or filer so that you can directly access the filer(s), and
- nsck *migrate* moves a volume's metadata from one back-end share to another.

The nsck and sync utilities do not apply to direct volumes, which contain no metadata. This chapter is relevant to managed volumes only.

Showing the Progress of nsck and sync Jobs

Before you begin running nsck and sync jobs, it is helpful to know how to monitor their progress. You can monitor the progress of any nsck or sync job with the show nsck command:

```
show nsck
```

This shows a table with one row per job. The last column is Status. All nsck reports go through two states in the Status column: Pending and Complete. The nsck and sync jobs that verify and/or change metadata have intermediate stages to indicate that they are in progress.

For example, the following jobs have all completed successfully:

```
bstnA(gbl)# show nsck

Op Id  Op Type      Namespace:Path                          Status
------ ------------ -------------------------------------- ------------------
1      report       insur:/claims                          Complete
2      report       wwmed:/acct                            Complete
3      report       wwmed:/acct                            Complete
4      report       wwmed:/acct/payable                    Complete
5      report       wwmed:/acct                            Complete
6      report       wwmed:/acct                            Complete
7      report       wwmed:/acct/payable                    Complete
8      report       wwmed:/acct                            Complete
9      report       wwmed:/acct                            Complete
10     report       wwmed:/acct                            Complete
11     sync         wwmed:/acct/payable                    Complete
12     report       medarcv:/lab_equipment                 Complete
13     report       medarcv:/rcrds                         Complete
14     report       medarcv:/test_results                  Complete
15     report       medarcv:/lab_equipment                 Complete
16     report       medarcv:/rcrds                         Complete
17     report       medarcv:/test_results                  Complete
18     report       medarcv:/lab_equipment                 Complete
19     report       medarcv:/rcrds                         Complete
20     report       medarcv:/test_results                  Complete
21     report       medarcv:/lab_equipment                 Complete
22     report       medarcv:/rcrds                         Complete
23     report       medarcv:/test_results                  Complete
24     report       medarcv:/lab_equipment                 Complete
25     report       medarcv:/rcrds                         Complete
26     sync         medarcv:/rcrds/sprains                 Complete
27     sync         medarcv:/rcrds/                        Complete
28     sync         medarcv:/lab_equipment/acme            Complete
29     sync         medarcv:/lab_equipment/                Complete
30     report       insur:/claims                          Complete
31     report       insur:/claims                          Complete
32     report       insur:/claims/tools                    Complete
33     report       insur:/claims                          Complete
34     report       insur:/claims                          Complete
35     report       insur:/claims/tools                    Complete
36     report       insur:/claims                          Complete
37     report       insur:/claims                          Complete
bstnA(gbl)# ...
```

Showing the nsck/sync Jobs for One Namespace

To view only the nsck and/or sync jobs for one namespace, add the namespace clause to the end of the show nsck command:

show nsck namespace *namespace*

where ***namespace*** (1-30 characters) specifies the namespace.

For example, the following command shows the nsck and sync jobs for the "wwmed" namespace:

```
bstnA(gbl)# show nsck namespace wwmed

Op Id  Op Type      Namespace:Path                          Status
------ ------------ -------------------------------------- ------------------
2      report       wwmed:/acct                            Complete
3      report       wwmed:/acct                            Complete
4      report       wwmed:/acct/payable                    Complete
5      report       wwmed:/acct                            Complete
6      report       wwmed:/acct                            Complete
7      report       wwmed:/acct/payable                    Complete
8      report       wwmed:/acct                            Complete
9      report       wwmed:/acct                            Complete
10     report       wwmed:/acct                            Complete
11     sync         wwmed:/acct/payable                    Complete
bstnA(gbl)# ...
```

Showing One nsck/sync Job

Use the Op Id (from the output of the above commands) to focus the show nsck command on a single nsck or sync job:

show nsck *job-id*

where ***job-id*** specifies the ID for the nsck or sync job.

This shows additional detail for the report, such as the name of the report and the switch where the report is kept (this is useful for redundant pairs).

For example, the following command shows the nsck job 1:

```
bstnA(gbl)# show nsck 1

Op Type:     report
Op Id:       1
Report Name: insur_fgns_1._claims.rpt
Switch:      bstnA
Namespace:   insur
Path:        /claims
Status:      Complete
bstnA(gbl)# ...
```

Showing all nsck/sync Jobs

Use show nsck all to see details for all nsck and sync jobs:

show nsck all

For example:

```
bstnA(gbl)# show nsck all

Op Type:     report
Op Id:       1
Report Name: insur_fgns_1._claims.rpt
Switch:      bstnA
Namespace:   insur
Path:        /claims
Status:      Complete

Op Type:     report
Op Id:       2
Report Name: inconsistencies.2.rpt
Switch:      bstnA
Namespace:   wwmed
...
bstnA(gbl)# ...
```

Showing all nsck/sync Reports

The show nsck command is focused on job status; if all nsck and sync jobs are cleared from the database (as described below), no jobs appear in show nsck. To see an up-to-date list of nsck or sync reports, use show reports type:

show reports type {Dstg | Inc | MdO | MgMd | MdU | Rbld | Sync | SySh}

where

Dstg lists all metadata-inconsistencies reports,

Inc lists all metadata-inconsistencies reports,

MdO lists all metadata-only reports,

MgMd lists all metadata-migration reports,

MdU lists all metadata-upgrade reports, A metadata upgrade may occur in the background after you install a new software release.

Rbld lists all rebuild reports,

Sync lists all sync reports, and

SySh lists all sync-share reports.

These report types are explained later in this chapter.

For example, the following output lists all sync reports:

```
bstnA# show reports type Sync


  reports
    Codes: Sync=Sync Files/Dirs
      sync.1._acct.rpt        Mar 20 01:06  1.4 k        Sync DONE: 31 in 00:00:00
      sync.2._rcrds.rpt       Mar 20 01:07  1.8 k        Sync DONE: 11 in 00:00:00
      sync.3._rcrds.rpt       Mar 20 01:07  1.9 k        Sync DONE: 245 in 00:00:00
      sync.4._lab_equipment.rpt Mar 20 01:07  1.5 k        Sync DONE: 0 in 00:00:01
      sync.5._lab_equipment.rpt Mar 20 01:07  2.0 k        Sync DONE: 0 in 00:00:00

bstnA# ...
```

Showing the Status of One Report

Use show reports status to show the one-line status for a particular report:

show reports status *report-name*

where ***report-name*** (1-255 characters) selects the report.

The output is a single status line for the report; it matches the line from the show reports output.

For example, this command shows the status of the “sync.3.rcrds.rpt” report.

```
bstnA# show reports status sync.3._rcrds.rpt


 reports
    Codes: Sync=Sync Files/Dirs
      sync.3._rcrds.rpt       04/23 02:11   2.0 kB        Sync DONE: 287 in 00:00:00

bstnA# ...
```

Showing Metadata

You can use the nsck report metadata-only command to show the physical locations of all files and directories in a namespace's managed volumes. A file's *physical location* is its directory path including the IP address of the filer where it resides; for example, 192.168.25.21:/usr/local/bin. You can use this command to see the before-and-after effects of namespace policy on the actual location of your files.

From priv-exec mode, use the nsck report metadata-only command to show physical file locations for a namespace:

```
nsck namespace report metadata-only [path] [norecurse]
```

where:

namespace (1-30 characters) specifies a namespace on which to report. Use the show namespace command for a list of configured namespaces.

path (optional, 1-256 characters) narrows the scope of the report to a specific virtual path in the namespace (for example, '/eng/share'). This must be a managed volume or a path within a managed volume.

norecurse (optional) specifies to not recurse into subdirectories during the report.

Note

You can run only one nsck job (report or rebuild) at a time on a given managed volume. Use show nsck *to monitor the progress of nsck jobs; see* ***Showing the Progress of nsck and sync Jobs**, on page 7-4.*

The report goes to a file, "metadata_only.*id*.rpt." The *id* is a unique number for every nsck report. The CLI shows the report name after you invoke the command. Use show reports to see the file listing; use show, tail, or grep to read the file. To save the report off to an external FTP site, use the copy ... ftp command from priv-exec mode.

This report has five major sections:

1. The start time, namespace, volume, and path for the report.
2. A table of namespace shares that were examined, along with the IP of the back-end filer and the share name at the filer.
3. A legend of keys for the next table.
4. A table of files and directories in the metadata. For each file/directory, the table shows three fields:
   a) Type is a list of file types (shown as keys from the legend above),
   b) Share is the filer-share name with the file, and
   c) Path is the exact path of the file on the share, relative to the share's root.

5. A summary list with various totals.

For example, the following command sequence exits to priv-exec mode, dumps all physical file locations for the 'wwmed' namespace, and reviews the report:

```
bstnA(gbl)# end
bstnA# nsck wwmed report metadata-only
Scheduling report: metadata_only.5.rpt on switch bstnA
bstnA# show reports metadata_only.5.rpt
**** Metadata-Only Report: Started at 04/23/2012 02:08:59 -0400 ****
**** Software Version: 6.02.000.14358 (Apr 18 2012 20:09:15) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:
**** Namespace: wwmed
**** Volume: /acct
**** Path: /acct

Share                 Physical Filer
--------------------  ------------------------------------------------------------
[budget            ]  192.168.25.19:/exports/budget
[bills             ]  192.168.25.25:/work1/accting
[bills2            ]  192.168.25.23:/exports/acct2
[it5               ]  192.168.25.24:/lhome/it5

**** Legend:
****   FL = File: The reported entry is a file.
****   DR = Directory: The reported entry is a directory.
****   SL = Symlink: The reported entry is a symbolic link.
****   LN = Link: The reported entry has a link count greater than one.
****   NL = No Lock: Was unable to lock parent directory during report.
****   CC = NFS case-blind name collision.
****   IC = Name contains invalid CIFS characters.
****   FN = Name may conflict with a filer-generated name.
****   SP = A persistent split is registered in the metadata, due to a FGN.
****   NF = Name is only accessible to NFS clients.
****   IA = Inconsistent attributes between this directory's master and stripes (recorded).
****   IS = Inconsistent attributes on this specific directory stripe (recorded).


Type                   Share                 Path
---------------------  --------------------  -------------------------------------
[   DR              ]  [budget            ]  /
[FL                 ]  [budget            ]  /layer3.fm.lck
[   DR              ]  [bills2            ]  /planner
[FL                 ]  [bills2            ]  /layer2.fm.lck
[FL                 ]  [bills2            ]  /cliOperator.book
[   DR              ]  [budget            ]  /production
[FL                 ]  [bills2            ]  /finance_sites.html
[FL                 ]  [bills             ]  /shellCmds_winXP.fm
[FL                 ]  [budget            ]  /sampleNet.html

...

[FL                 ]  [budget            ]  /receipts/giftShop/sep2006.csv
[FL                 ]  [bills2            ]  /receipts/giftShop/xmas2009.csv
[FL                 ]  [bills2            ]  /receipts/giftShop/aug2009.csv

**** Total Files:                      3,994
**** Total Directories:                  439
**** Total Hard Links (nlink>1):           0
**** Total Symlinks:                       9
```

```
**** Total Locking Errors:                 0

**** Total items:              4,433
**** Elapsed time:           00:00:00
**** Metadata-Only Report: DONE at 04/23/2012 02:08:59 -0400 ****

bstnA# ...
```

Focusing on One Share

You can generate a report that focuses on one managed-volume share. Use the optional share clause to report the metadata for a single share:

nsck *namespace* report metadata-only [*path*] share *share-name* [norecurse]

where

namespace, **report metadata-only**, ***path***, and **norecurse** are all explained above,

share *share-name* (optional; 1-64 characters) specifies the namespace share.

If the share does not hold any of the files in the ***path***, the report is empty.

For example, the following command sequence dumps the metadata for the "bills" share in "wwmed:"

```
bstnA(gbl)# end
bstnA# nsck wwmed report metadata-only share bills
Scheduling report: metadata_only.20.rpt on switch bstnA
bstnA# ...
```

Sending the Output to a Different File

Use the optional outputfile clause to send the nsck results to a file other than the default, "metadata_only.*n*.rpt:"

nsck *namespace* report metadata-only [*path*] [share *share-name*] [norecurse] outputfile *file-name*

where

name, **report metadata-only**, ***path***, **share *share-name***, and **norecurse** are all explained above, and

outputfile *file-name* (optional; 1-255 characters) specifies a prefix for the file name. Use show reports to see the file listing; use show, tail, or grep to read the file. If the ***file-name*** already exists, the CLI prompts for confirmation.

For example, the following command sequence dumps a share's file locations to a file named "wwmed_bills.8.rpt:"

```
bstnA(gbl)# end
bstnA# nsck wwmed report metadata-only share bills outputfile wwmed_bills
Scheduling report: wwmed_bills.8.rpt on switch bstnA
bstnA# ...
```

Canceling the Report

This report operation can require a long time and heavy CPU resources if run on a very-large namespace. You can use the cancel nsck report command to cancel a long-running nsck report:

cancel nsck report *job-id*

where ***job-id*** (1-2,147,483,647) identifies the nsck-report job to cancel. This ID appears in the output of show nsck.

The CLI prompts for confirmation before canceling the report job. Enter **yes** to proceed.

For example, the following command sequence stops job 19, which was still in progress:

```
bstnA(gbl)# show nsck


  show nsck

Op Id  Op Type      Namespace:Path                         Status
------ ------------ -------------------------------------- ------------------
1      report       insur:/claims                          Complete
2      report       wwmed:/acct                            Complete
3      report       wwmed:/acct                            Complete
4      report       wwmed:/acct/payable                    Complete
5      report       wwmed:/acct                            Complete
6      report       wwmed:/acct                            Complete
7      report       wwmed:/acct/payable                    Complete
8      report       wwmed:/acct                            Complete
9      report       wwmed:/acct                            Complete
10     report       wwmed:/acct                            Complete
11     sync         wwmed:/acct/payable                    Complete
12     report       medarcv:/lab_equipment                 Complete
13     report       medarcv:/rcrds                         Complete
14     report       medarcv:/lab_equipment                 Complete
15     report       medarcv:/test_results                  Complete
16     report       medarcv:/rcrds                         Complete
17     report       medarcv:/lab_equipment                 Complete
18     report       medarcv:/test_results                  Complete
19     report       medarcv:/rcrds                         Report in progress
bstnA(gbl)# end
bstnA# cancel nsck report 19

Cancel the nsck metadata-only report operation 'metadata_only.18.rpt'? [yes/no] yes
bstnA# ...
```

Showing Directory Structure

You can display a summary of a managed volume's directory structure without having to run an actual metadata report, using the CLI command nsck report dir-structure. This generates a directory structure report that you can view later.

The command syntax is:

nsck *name* report dir-structure [*path*] [norecurse] [summary] [outputfile *outputfilename*]

where:

name specifies the namespace for which to run the directory structure report.

path specifies a volume or volume/path on which to report.

norecurse indicates that the report will not explore subdirectories.

summary indicates that the report will not list the details of the individual subdirectories it encounters.

outputfile specifies the name of the file to which the report output is written. The "outputfile" option is a report prefix that has the ".rpt" extension appended to it at a minimum. If no path is specified, the file name is outputfile.volume.rpt.

For example:

nsck wwmed report dir-structure path outputfile wwmed_dirstruct

generates a directory structure report for the namespace "wwmed" that is written to the file named "wwmed_dirstruct". The report recurses into subdirectories that it encounters, and is a full report, not a summary.

Use the show reports type ds CLI command to list the directory structure reports that have been generated already.

If this command is executed for a namespace without specifying a path, one report is generated per volume in the namespace. (This includes direct volumes.)

Example

This example shows the results of the command, nsck insur report dir-structure:

```
bstnA# show reports dir_structure.43.rpt
**** Directory Structure Report: Started at 04/25/2012 12:01:25 -0400
****
**** Software Version: 6.02.000.14360 (Apr 23 2012 20:09:34)
[nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:
**** Namespace: insur
**** Volume: /claims
**** Path: /claims
```

```
Share                 Physical Filer
--------------------
------------------------------------------------------------
[shr1-old          ]  192.168.25.21 NFS:/vol/vol2/insurance,
CIFS:insurance
[shr1-next         ]  192.168.25.51NFS:/root_vdm_4/patient_records,
CIFS:patient_records

**** Legend:
****   NL = No Lock: Was unable to lock parent directory during
report.

Flag  Dirs      Files     Links    Symlinks  Path
----  --------- --------- -------- --------
------------------------------------------------
[  ]         15         8        0        0  /claims/
[  ]          0         0        0        0  /claims/Claims:2001/
[  ]          0         0        0        0  /claims/claims:2005/
[  ]          1        24        0        2  /claims/common/
[  ]          0         4        0        0  /claims/draft_proposals./
[  ]          0        50        0        0  /claims/images/
[  ]          0         0        0        0  /claims/POLICI~2/
[  ]          2         7        0        1  /claims/specs/
[  ]          2         9        0        0  /claims/stats/
[  ]          0         8        0        0  /claims/tools/
[  ]          0         0        0        0  /claims/Tools/
[  ]          0         0        0        0  /claims/y2kclaims/
[  ]          0         0        0        0
/claims/dir0134(U+0134)_shr1-old-12/
[  ]          0         1        0        0  /claims/common/eastLab/
[  ]          0        22        0        0  /claims/specs/common/
[  ]          0         3        0        0
/claims/stats/in_home:2005/
[  ]          1         1        0        1  /claims/specs/protos/
[  ]          0         5        0        0  /claims/stats/otj_latest/
[  ]          0         0        0        0
/claims/specs/protos/accepted/
[  ]          0         9        0        0  /claims/Y2KCLA~1/
[  ]          0         0        0        0  /claims/Y2KCLA~4/

Directory Structure Summary:
  Maximum Entries in a directory:                    50
  Maximum sub-directories in a directory:            15
  Maximum Files in a directory:                      50
  Maximum Links in a directory:                       0
  Maximum Symlinks in a directory:                    2
  Average Entries in a directory:                     8
  Average Files in a directory:                       7

**** Total Files:                          151
**** Total Directories:                      21
**** Total Leaf Directories:                 16
**** Total Interior Directories:              5
**** Total Hard Links (nlink>1):              0
**** Total Symlinks:                          4

**** Total items:                 172
**** Elapsed time:           00:00:00
**** Directory Structure Report: DONE at 04/25/2012 12:01:25 -0400
****
```

Cancelling the Report

Directory structure reports can be cancelled by executing the command cancel nsck report *id*, where *id* specifies the nsck job to be cancelled. nsck jobs have IDs in the range of 1 to 2,147,483,647.

Warning Signs for Metadata Inconsistencies

If clients get front-end-service (NFS or CIFS) failures for a particular managed volume, the volume's metadata might be inconsistent. For example, clients might see files disappearing ("file not found" and/or "stale NFS filehandle" errors) or new files that appear unexpectedly. Inconsistent metadata is the result of changes happening directly on a filer, not through its managed volume. Best practices dictate that you should avoid this situation at all times. However, some situations make this impossible to prevent:

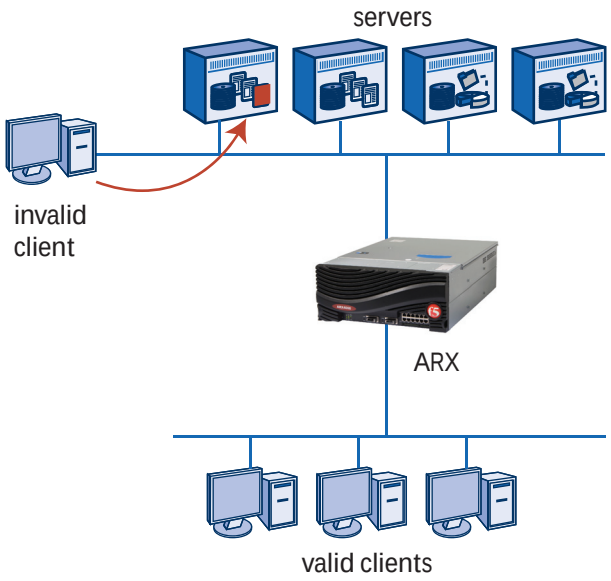
- On reboot recovery, some filers replace files that were deleted shortly before the reboot.
- A filer-integrated Anti-Virus application may rename, remove, or create files on the filer itself.
- Some customers prefer to perform restore operations directly on their back-end filers.

Whenever a managed-volume client tries to access a file that has been removed by one of the above operations, the client system reports an internal error. For CIFS operations, messages appear in the syslog (labeled UNXFINDNONE, UNXNOPATH, UNXOPENTYPE and/or UNXCRETYPE.), and the namespace software sends an SNMP trap (vcifsNotfound or vcifsTypeMismatch). Use grep to search the syslog for a string. (The syslog is rotated regularly; use show logs for a complete list of all log files, including backups.)

To find the exact scope of metadata inconsistency, use the nsck report inconsistencies command described below.

Finding Metadata Inconsistencies

The nsck report inconsistencies utility detects metadata inconsistencies and produces a report. Metadata inconsistencies occur when a client is connected directly to a back-end share and performs writes or deletes through that direct connection. Without the ARX in the data path, the switch has no knowledge of the changes and cannot update its metadata.



A local process on the filer, such as anti-virus software, can cause an inconsistency by removing or renaming a file.

You can use the nsck report inconsistencies to determine whether to run sync files or a full nsck rebuild on a namespace, or to confirm that there are no inconsistencies at all.

From priv-exec mode, use the nsck report inconsistencies command to check the current namespace's managed volumes for metadata inconsistency:

```
nsck name report inconsistencies [path] [share share-name]
[norecurse] [nofilehandles] [outputfile outputfile]
```

where:

name (1-30 characters) is the namespace on which to report. Use the show namespace command for a list of configured namespaces.

path (optional, 1-256 characters) narrows the scope of the report to a specific virtual path in the namespace (for example, '/home/jrandom'). This must be a managed volume, or a path within a managed volume.

share *share-name* (optional, 1-64 characters) narrows the scope further, to a single namespace share. This is the share's name in the namespace, not on the filer.

norecurse (optional) specifies to not recurse into subdirectories during the report.

nofilehandles (optional) specifies to not validate filehandles during a report.

outputfile *outputfile* (optional, 1-255 characters) specifies a prefix for a customized file name (for example, 'jrandom_inconsistencies'), as opposed to the default. Use the show reports command to display the file in the maintenance directory.

You can run one nsck job at a time on a given volume. Use show nsck to monitor the progress of nsck jobs; see *Showing the Progress of nsck and sync Jobs*, on page 7-4.

The CLI shows the report name after you invoke the command. Use show reports for a full list of nsck reports, and use show, tail, or grep to view one report. The report contains the following information:

1. The start time, namespace, volume, and path for the report.
2. A table of shares that were examined, along with the IP of the back-end filer and the share name at the filer.
3. A legend of keys for various types of metadata inconsistency. These keys are used in the next table.
4. A table of files with metadata inconsistencies. For each file, the table shows
   a) Type - a list of the file's inconsistencies (shown as keys from the legend above),
   b) Share - the name of the ARX share with the inconsistent file, and
   c) Path - the exact path of the file on the share, relative to the share's root.
5. A summary list with various totals.

For example, the following command sequence finds a missing file in the medarcv namespace (highlighted in bold text):

```
bstnA(gbl)# end
bstnA# nsck medarcv report inconsistencies norecurse
Scheduling report: inconsistencies.15.rpt on switch bstnA
bstnA# show reports inconsistencies.15.rpt
**** Inconsistencies Report: Started at 04/25/2012 11:59:39 -0400 ****
**** Software Version: 6.02.000.14360 (Apr 23 2012 20:09:34) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:
**** Namespace: medarcv
**** Volume: /rcrds
**** Path: /rcrds

Share                Physical Filer
-------------------  ------------------------------------------------------------
[rx                ]  192.168.25.29:prescriptions
[charts            ]  192.168.25.20:histories
```

```
[bulk              ]  192.168.25.27:bulkstorage

**** Legend:
****   LF = File exists in the metadata, but is missing from the physical filer.
****   LD = Directory exists in the metadata, but is missing from the physical filer.
****   FF = File exists on the physical filer, but is missing from the metadata.
****   FD = Directory exists on the physical filer, but is missing from the metadata.
****   LL = File is a symlink in the metadata, but is a regular file on the filer.
****   FL = File is a symlink on the filer, but is a regular file in the metadata.
****   IF = Filehandles in the metadata do not match the filehandles on the physical filer.
****   MF = The file is currently being migrated.
****   NL = Unable to lock parent directory during report.
****   FE = Error contacting filer during report.
****   FO = Filer Offline: The filer is offline or disabled.
****   F8 = A file name matches a CIFS alternate "8.3" name on another share.
****   D8 = A directory name matches a CIFS alternate "8.3" name; its contents will be skipped.
****   DC = A client has the file or directory open for delete-on-close, but the filer has already
deleted it.
****   SD = Striped leaf directory found on filer, expected on other shares.
****   SL = File is a symlink.
****   UT = Name contains characters that are invalid UTF-8; must solve issue directly on the filer
****   IS = Inconsistent attributes on one of this directory's stripes (discovered)
****   MI = Attributes are consistent, metadata marked as inconsistent
****   SI = Attributes are inconsistent, metadata not marked as inconsistent


Type                       Share                Path
-------------------------  -------------------  ------------------------------
[LF                    ]  [charts            ]  /copyRandomx.exe
[   F8                 ]  [charts            ]  /KMO_ME~1.DAT -> kmo_medical_record.dat
[   D8                 ]  [charts            ]  /RECORD~1/ -> records_predating_y2k

**** Total Found Items:                   0
**** Total Lost Items:                    1
**** Total Invalid Filehandles:           0
**** Total Migrating Files:               0
**** Total Deleted Before Close:          0
**** Total Locking Errors:                0
**** Total Filer Errors:                  0
**** Total 8.3 Errors:                    2
**** Total Found Stripes:                 0
**** Total Inconsistent Attrs:            0

**** Total processed:             218
**** Elapsed time:           00:00:00
**** Inconsistencies Report: DONE at 04/25/2012 11:59:39 -0400 ****

bstnA# ...
```

Focusing on Multi-Protocol Issues

A volume in a multi-protocol (CIFS and NFS) namespace may have files whose names are inconsistent between CIFS and NFS. Some characters are legal in NFS file names but unsupported in CIFS names, just as CIFS has some naming conventions that NFS cannot always follow. (These issues are discussed extensively in a later chapter about troubleshooting managed

volumes.) To create a report that shows all multi-protocol naming issues in a volume, add the multi-protocol flag to the nsck ... report inconsistencies command:

nsck *name* report inconsistencies [*path*] [share *share-name*] [norecurse] [nofilehandles] [multi-protocol] [outputfile *outputfile*]

where:

multi-protocol (optional) focuses the report on NFS/CIFS naming issues, and

the remaining options were described above.

For example, the following command sequence generates an inconsistencies report for the "insur~/claims" volume that focuses on multi-protocol issues, then shows the report:

```
bstnA(gbl)# end
bstnA# nsck insur report inconsistencies multi-protocol outputfile insur_fgns
Scheduling report: insur_fgns._claims.rpt on switch bstnA
bstnA# show reports insur_fgns._claims.rpt
**** Inconsistencies Report: Started at 04/25/2012 12:01:43 -0400 ****
**** Software Version: 6.02.000.14360 (Apr 23 2012 20:09:34) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:
**** Namespace: insur
**** Volume: /claims
**** Path: /claims
**** multiprotocol flag specified

Share                 Physical Filer
--------------------  ------------------------------------------------------------
[shr1-old          ]  192.168.25.21 NFS:/vol/vol2/insurance, CIFS:insurance
[shr1-next         ]  192.168.25.51NFS:/root_vdm_4/patient_records, CIFS:patient_records

**** Legend:
****   LF = File exists in the metadata, but is missing from the physical filer.
****   LD = Directory exists in the metadata, but is missing from the physical filer.
****   FF = File exists on the physical filer, but is missing from the metadata.
****   FD = Directory exists on the physical filer, but is missing from the metadata.
****   LL = File is a symlink in the metadata, but is a regular file on the filer.
****   FL = File is a symlink on the filer, but is a regular file in the metadata.
****   IF = Filehandles in the metadata do not match the filehandles on the physical filer.
****   MF = The file is currently being migrated.
****   NL = Unable to lock parent directory during report.
****   FE = Error contacting filer during report.
****   FO = Filer Offline: The filer is offline or disabled.
****   F8 = A file name matches a CIFS alternate "8.3" name on another share.
****   D8 = A directory name matches a CIFS alternate "8.3" name; its contents will be skipped.
****   DC = A client has the file or directory open for delete-on-close, but the filer has already
deleted it.
****   SD = Striped leaf directory found on filer, expected on other shares.
****   SL = File is a symlink.
****   UT = Name contains characters that are invalid UTF-8; must solve issue directly on the filer
****   IS = Inconsistent attributes on one of this directory's stripes (discovered)
****   MI = Attributes are consistent, metadata marked as inconsistent
****   SI = Attributes are inconsistent, metadata not marked as inconsistent

****
****  Multi-Protocol Only:
****   NF = Name is accessible only to NFS clients.
****   NI = Name does not exist via both NFS and CIFS.
```

```
****   CC = Case-blind name collision.
****   IC = CIFS invalid characters found in NFS name.
****   FN = A portion of the name contains a filer-generated pattern.
****   NM = CIFS name has characters that are not mappable to the NFS encoding.
****   SP = A NFS name in the path's ancestry will prevent CIFS access.

Type                        Share                Path
--------------------------  -------------------  ------------------------------
[               NF CC   ]  [shr1-old          ]  /INDEX.html
[               NF CC   ]  [shr1-old          ]  /index.html
[               NF CC   ]  [shr1-old          ]  /Tools
[               NF CC   ]  [shr1-old          ]  /tools
[               NF IC   ]  [shr1-old          ]  /Claims:2001
[               NF IC   ]  [shr1-old          ]  /claims:2005
[               NF IC   ]  [shr1-old          ]  /draft_proposals.
[   F8                  ]  [shr1-old          ]  /OVERDU~1.DOC -> overdueclaimsmemo.doc
[                     SP]  [shr1-old          ]  /draft_proposals./FY06challenges.ppt
[                     SP]  [shr1-old          ]  /draft_proposals./claim2389_rebut2.doc
[                     SP]  [shr1-old          ]  /draft_proposals./FY2007propB.ppt
[                     SP]  [shr1-old          ]  /draft_proposals./claim2389_rebut1.doc
[               NF IC   ]  [shr1-old          ]  /images/FRU replace wrongway.tif
[               NF IC   ]  [shr1-old          ]  /images/:KJUK000
[               NI NM   ]  [shr1-old          ]  /images/file012bÄ«/ (Characters: U+012b)
[               NF CC   ]  [shr1-old          ]  /stats/piechart.ppt
[               NF CC   ]  [shr1-old          ]  /stats/PieChart.ppt
[               NF IC   ]  [shr1-old          ]  /stats/on_the_job:2004.cnv
[               NF IC   ]  [shr1-old          ]  /stats/on_the_job:2003.cnv
[               NF IC   ]  [shr1-old          ]  /stats/in_home:2005
[                     SP]  [shr1-old          ]  /tools/extractCdDocs.pl
[                     SP]  [shr1-old          ]  /tools/cleanBU.csh
[                     SP]  [shr1-old          ]  /tools/checkinAll.pl
[                     SP]  [shr1-old          ]  /tools/docSet.pl
[                     SP]  [shr1-old          ]  /tools/updateCvs.csh
[                     SP]  [shr1-old          ]  /tools/books.xml
[                     SP]  [shr1-old          ]  /tools/blowAway.pl
[                     SP]  [shr1-old          ]  /tools/makeCd.pl
[               NF IC   ]  [shr1-old          ]  /stats/in_home:2005/age:11-21yrs.csv
[               NF IC   ]  [shr1-old          ]  /stats/in_home:2005/age:>21yrs.csv
[               NF IC   ]  [shr1-old          ]  /stats/in_home:2005/age:<10yrs.csv
[               NF IC   ]  [shr1-old          ]  /stats/in_home:2005/age:>21yrs.csv
[               NF IC   ]  [shr1-old          ]  /stats/in_home:2005/age:<10yrs.csv
[               NF IC   ]  [shr1-old          ]  /stats/in_home:2005/age:11-21yrs.csv
[                     SP]  [shr1-old          ]  /stats/in_home:2005/age:<10yrs.csv
[                     SP]  [shr1-old          ]  /stats/in_home:2005/age:>21yrs.csv
[                     SP]  [shr1-old          ]  /stats/in_home:2005/age:11-21yrs.csv
[   D8                  ]  [shr1-old          ]  /Y2KCLA~1/ -> y2kclaims

**** Total Found Items:                   0
**** Total Lost Items:                    0
**** Total Invalid Filehandles:           0
**** Total Migrating Files:               0
**** Total Deleted Before Close:          0
**** Total Locking Errors:                0
**** Total Filer Errors:                  0
**** Total 8.3 Errors:                    2
**** Total Found Stripes:                 0
**** Total Inconsistent Attrs:            0

**** Total processed:             171
**** Elapsed time:           00:00:00
**** Inconsistencies Report: DONE at 04/25/2012 12:01:43 -0400 ****
bstnA# ...
```

Canceling the Report

This report operation can require a long time and heavy CPU resources if run on a very-large namespace. As with a metadata-only report, you can use the cancel nsck report command to cancel a long-running inconsistencies report:

cancel nsck report *job-id*

where ***job-id*** (1-2,147,483,647) identifies the nsck-report job to cancel. This ID appears in the output of show nsck.

The CLI prompts for confirmation before canceling the report job. Enter **yes** to proceed.

For example, the following command sequence stops job 30, an inconsistencies report that was still in progress:

```
bstnA(gbl)# show nsck


  show nsck

Op Id  Op Type      Namespace:Path                          Status
------ ------------ --------------------------------------- ------------------
1      report       insur:/claims                           Complete
2      report       wwmed:/acct                             Complete
3      report       wwmed:/acct                             Complete
4      report       wwmed:/acct/payable                     Complete
5      report       wwmed:/acct                             Complete
6      report       wwmed:/acct                             Complete

...

26     sync         medarcv:/rcrds/sprains                  Complete
27     sync         medarcv:/rcrds/                         Complete
28     sync         medarcv:/lab_equipment/acme             Complete
29     sync         medarcv:/lab_equipment/                 Complete
30     report       insur:/claims                           Report in progress
bstnA(gbl)# end
bstnA# cancel nsck report 30

Cancel the nsck inconsistencies report operation 'inconsistencies.25.rpt'? [yes/no] yes
bstnA# ...
```

Synchronizing Metadata with Actual Files

An earlier example showed a file, copyRandom.exe, that is recorded in volume metadata but is missing from its back-end filer. You can use the *sync* utility to re-synchronize the metadata with the actual contents of the filers. You can configure a CIFS volume to automatically sync itself whenever a client encounters a metadata error (see *Automatically Synchronizing Metadata (CIFS)*, on page 9-10 of the *ARX® CLI Storage-Management Guide*), or you can run the sync operation manually as described here. From priv-exec mode, use the sync files command to synchronize metadata for files:

```
sync files namespace volume vol-path path path [recurse]
[rename-files]
```

where:

namespace (1-30 characters) is the namespace to sync. Use the show namespace command for a list of configured namespaces (see *Listing All Namespaces*, on page 7-4 of the *ARX® CLI Storage-Management Guide*).

vol-path (1-1024 characters) chooses a specific managed volume in the namespace. Use forward slashes (/) in the path, even for CIFS volumes.

path (1-1024 characters) selects a virtual path in the volume (for example, '/' or '/home/jrandom'). You can shorten the time required for the sync with this option; use the metadata-inconsistencies report to find the best path.

You can enter the optional **recurse** and **rename-files** flags in any order:

- **recurse** (optional) causes the sync operation to descend into subdirectories.
- **rename-files** (optional) allows the sync process to rename any newly-discovered file that has the same name as a previously-imported file. (This command generates a report, described below, that identifies renamed files.)

In multi-protocol (NFS and CIFS) volumes, this also synchronizes the metadata with the current state of filer-generated names (FGNs, such as "FILE~2.DOC") on its back-end filers. This is discussed in detail in a later chapter about troubleshooting managed volumes.

As with nsck jobs, every sync operation produces a report as it runs. The CLI shows the report name after you invoke the command. Use show reports for a full list of sync reports, and use show, tail, or grep to view one report. The report is similar to the metadata-inconsistencies report, described above. This report adds some additional Type keys to show which files were synchronized, were renamed, or encountered synchronization issues.

For example, this command sequence re-synchronizes the medarcv~/rcrds volume and then shows the report:

```
bstnA(gbl)# end
bstnA# sync files medarcv volume /rcrds path / recurse rename-files
Scheduling sync files operation on switch bstnA, report name: sync.3._rcrds.rpt
bstnA# show reports sync.3._rcrds.rpt
**** Sync Report: Started at 04/25/2012 12:00:38 -0400 ****
**** Software Version: 6.02.000.14360 (Apr 23 2012 20:09:34) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:

**** Namespace: medarcv
**** Volume:    /rcrds
**** Path:      /
**** Options:   recurse rename-files

Share                Physical Filer
-------------------  ------------------------------------------------
[rx               ]  192.168.25.29:prescriptions
[charts           ]  192.168.25.20:histories
[bulk             ]  192.168.25.27:bulkstorage

**** Legend:
****
****  Actions:
****   SY = File metadata has been synchronized.
****   LF = Lost File, File exists in the metadata, but not on filer.
****   FF = Found File, File exists on filer, but not in the metadata.
****   LL = Lost Symlink, File is a symlink in the metadata, but not on filer.
****   FL = Found Symlink, File is a symlink on the filer, but not in the metadata.
****   IF = Invalid File Handle, Metadata/Filer file handle mismatch.
****   CR = File collided with existing filename, and was renamed.
****   NC = File or directory is now accessible to both NFS and CIFS.
****   SD = Striped directory found on filer; registered in metadata.
****   SC = Directory stripe inconsistency has been cleared.
****   SI = Directory stripe is now inconsistent due to attempted synchronization.
****
****  Warnings:
****   CC = NFS case-blind name collision, FGN may result under CIFS.
****   IC = Name contains invalid characters for CIFS, FGN will result under CIFS.
****   NI = Name does not exist via both NFS and CIFS.
****   UN = Name contains unmappable unicode characters, FGN will result under NFS.
****   U8 = Name contains invalid UTF-8 characters, not imported.
****   CD = CIFS case blind name collision, directory ignored.
****   RV = Reserved name encountered on filer, not imported.
****   NF = Name is only accessible using NFS.
****   SL = File is a symlink.
****
****  Errors:
****   TO = Operation timed out: An operation to the filer timed out.
****   LK = Failed to acquire Lock.
****   FE = Encountered filer error during sync.
****   CF = File collided with existing filename.
****   CN = CIFS case blind name collision, file renamed.
****   CZ = CIFS case blind name collision, no rename, use rename-files.
****   FD = Found Directory, Directory exists on filer, but not in the metadata.
****   NE = Directory not empty


Synchronization Results:
================================================================================
```

```
Type                 Share               Object
-------------------  ------------------  ------------------------------------
[SY LF            ]  [charts          ]  /copyRandomx.exe

================================================================================

Total Found Items:                              0
Total Lost Items:                               1
Total Invalid Filehandles:                      0
File Name Collisions:                           0
File Collision Renames:                         0
Total Migrations Aborted:                       0
Total Items Synchronized:                       1
Directories completely processed:              33
Total Directories:                             33
CIFS Case Blind Collision Renames:              0

**** Total processed:             284
**** Elapsed time:           00:00:01
**** Sync Report: DONE at 04/25/2012 12:00:39 -0400 ****

bstnA# ...
```

Synchronizing with Actual Directories

Some installations require new directories that cannot be configured through the client interfaces. These directories have features that can only be added through an administrative interface on the back-end filer itself, such as disk quotas. If such a directory is created behind a managed volume, you can use the sync utility to add it to the metadata. This can only function for empty directories, and for directories that are only on a single share behind the managed volume. From priv-exec mode, use the sync directories command to search a managed-volume directory for any new, unrecorded subdirectories:

sync directories *namespace* volume *vol-path* path *path*

where:

namespace (1-30 characters) is the namespace to sync. Use the show namespace command for a list of configured namespaces (see *Listing All Namespaces*, on page 7-4 of the *ARX® CLI Storage-Management Guide*).

vol-path (1-1024 characters) chooses a specific managed volume in the namespace. Use forward slashes (/) in the path, even for CIFS volumes.

path (1-1024 characters) selects a virtual path in the volume (for example, '/' or '/home/jrandom'). The sync utility searches for new directories under this path; it only searches this directory, and does not descend into any known or discovered subdirectories.

Like the nsck and sync-files operations, this generates a report as it runs. The report name appears after you issue the command.

For example, this command sequence synchronizes the root directory in the medarcv~/lab_equipment volume and then shows the report:

```
bstnA(gbl)# end
bstnA# sync directories medarcv volume /lab_equipment path /
Scheduling sync directories operation on switch bstnA, report name: sync.5._lab_equipment.rpt
bstnA# show reports sync.5._lab_equipment.rpt
**** Sync Report: Started at Wed Feb 24 01:58:07 2010 ****
**** Software Version: 5.02.000.12538 (Feb 16 2010 20:14:13) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:

**** Namespace: medarcv
**** Volume:    /lab_equipment
**** Path:      /

Share                 Physical Filer
--------------------  ------------------------------------------------
[equip             ]  192.168.25.49:equipment
[leased            ]  192.168.25.49:for_lease
[backlots          ]  192.168.25.27:backlot_records
[scanners          ]  192.168.25.71:xraysScanners

**** Legend:
****
****  Actions:
****   SY = File metadata has been synchronized.
****   LF = Lost File, File exists in the metadata, but not on filer.
****   FF = Found File, File exists on filer, but not in the metadata.
****   LL = Lost Symlink, File is a symlink in the metadata, but not on filer.
****   FL = Found Symlink, File is a symlink on the filer, but not in the metadata.
****   IF = Invalid File Handle, Metadata/Filer file handle mismatch.
****   CR = File collided with existing filename, and was renamed.
****   NC = File or directory is now accessible to both NFS and CIFS.
****   SD = Striped directory found on filer; registered in metadata.
****
****  Warnings:
****   CC = NFS case-blind name collision, FGN may result under CIFS.
****   IC = Name contains invalid characters for CIFS, FGN will result under CIFS.
****   NI = Name does not exist via both NFS and CIFS.
****   UN = Name contains unmappable unicode characters, FGN will result under NFS.
****   U8 = Name contains invalid UTF-8 characters, not imported.
****   CD = CIFS case blind name collision, directory ignored.
****   RV = Reserved name encountered on filer, not imported.
****   NF = Name is only accessible using NFS.
****   SL = File is a symlink.
****
****  Errors:
****   TO = Operation timed out: An operation to the filer timed out.
****   LK = Failed to acquire Lock.
****   FE = Encountered filer error during sync.
****   CF = File collided with existing filename.
****   CN = CIFS case blind name collision, file renamed.
****   CZ = CIFS case blind name collision, no rename, use rename-files.
****   FD = Found Directory, Directory exists on filer, but not in the metadata.
****   NE = Directory not empty


Synchronization Results:
================================================================================

Type                Share              Object
```

```
------------------  -----------------  ------------------------------------
[SY    FD         ]  [leased          ]  /siemans

================================================================================

Total Found Items:                          1
Total Lost Items:                           0
Total Invalid Filehandles:                  0
File Name Collisions:                       0
File Collision Renames:                     0
Total Migrations Aborted:                   0
Total Items Synchronized:                   1
Directories completely processed:           1
Total Directories:                          1
CIFS Case Blind Collisions:                 0
Non-Empty Directories:                      0

**** Total processed:              0
**** Elapsed time:           00:00:00
**** Sync Report: DONE at Wed Feb 24 01:58:07 2010 ****
bstnA# ...
```

Synchronizing Metadata in a Single Share

If the metadata inconsistencies are limited to a single, known share, you can focus the sync process on the affected share. A narrow focus leads to shorter sync times; you can use the metadata-inconsistencies report (described above) to determine the share(s) with metadata inconsistencies.

To focus on one share, add the share option after the *path*:

sync files *namespace* volume *vol-path* path *path* share *share-name* [recurse] [rename-files]

or, for directories,

sync directories *namespace* volume *vol-path* path *path* share *share-name*

where:

share-name (1-64 characters) identifies the share to sync. This is the share's name in the namespace, not on the filer.

The remaining arguments and flags are explained above.

For example, this command sequence syncs files in a single share, "wwmed~/acct~bills." It calls for a recursive sync, and it allows renaming of newly-discovered files that introduce collisions:

```
bstnA(gbl)# end
bstnA# sync files wwmed volume /acct path /payable share bills recurse rename-files
Scheduling sync files operation on switch bstnA, report name: sync.1._acct.rpt
bstnA# ...
```

Synchronization Occurs in Share-Priority Order

Whenever multiple shares are involved in a synchronization, the sync process scans the shares according to their import priority. This is important for any discovered files that collide with each other; the share with the

higher priority wins the file conflict, and the other share's file must be renamed. It is also important for colliding directories; the higher priority share gets the *master* instance of the directory and the other share gets a *stripe*. The master instance controls the attributes of the directory (such as permissions and named streams), and gets all newly-created files by default. Tiered volumes use import priority to differentiate their Tier-1 shares from the rest.

Refer to *Setting the Share's Priority (for Tiering)*, on page 9-36 of the *ARX® CLI Storage-Management Guide* for more information on setting the import priority for a volume's shares.

Showing the Progress of All Sync Operations

A sync operation occurs in the background, similar to an nsck run. To see the high-level status of all sync operations, use the show sync command:

```
show sync {files | directories}
```

For example, this shows all sync-files operations:

```
bstnA(gbl)# show sync files

Namespace: wwmed
    Volume: /acct
        Path: /payable
            Options:            recurse rename-files
            Status:             Success
            Report Name:        sync.1._acct.rpt (bstnA)
            Share:              bills
            Lost Items:         0
            Found Items:        0
            Synchronized Items: 0
            Processed Items:    21


Namespace: medarcv
    Volume: /rcrds
        Path: /sprains
            Options:            recurse rename-files
            Status:             Success
            Report Name:        sync.2._rcrds.rpt (bstnA)
            Lost Items:         0
            Found Items:        0
            Synchronized Items: 0
            Processed Items:    11

        Path: /
            Options:            recurse rename-files
            Status:             Success
            Report Name:        sync.3._rcrds.rpt (bstnA)
            Lost Items:         1
            Found Items:        0
            Synchronized Items: 1
            Processed Items:    245

bstnA(gbl)# ...
```

Focusing on a Single Namespace, Volume, or Path

You can add some options to show a smaller set of sync operations:

show sync {files | directories} [*namespace* [volume *vol-path* [path *path*]]]

where

files | directories chooses the type of sync operation,

namespace (1-30 characters) selects a namespace,

vol-path (1-1024 characters) narrows the scope to a specific volume, and.

path (1-1024 characters) narrows the scope further, to a specific virtual path in the volume (for example, '/home/jrandom'). This is relative to the ***vol-path***.

For example, this command shows all the sync-file operations performed on the medarcv~/rcrds volume:

```
bstnA(gbl)# show sync files medarcv volume /rcrds

Namespace: medarcv
    Volume: /rcrds
        Path: /sprains
            Options:            recurse rename-files
            Status:             Success
            Report Name:        sync.2._rcrds.rpt (bstnA)
            Lost Items:         0
            Found Items:        0
            Synchronized Items: 0
            Processed Items:    11

        Path: /
            Options:            recurse rename-files
            Status:             Success
            Report Name:        sync.3._rcrds.rpt (bstnA)
            Lost Items:         1
            Found Items:        0
            Synchronized Items: 1
            Processed Items:    245

bstnA(gbl)# ...
```

Clearing Sync Operations from the Output

The show sync command displays a running history of all sync operations. The history goes back indefinitely, so the command can display a very large number of records over time. You can clear all of these records at once, or all records for a particular namespace, volume, or path. From priv-exec mode, use the clear sync command:

clear sync {files | directories} [*namespace* [volume *vol-path* [path *path*]]]

where

files | directories chooses the type of sync record to clear,

namespace (optional, 1-30 characters) identifies one namespace from which to clear sync records. If this option is omitted, the command clears all sync-files or sync-directories records from the history.

vol-path (optional, 1-1024 characters) narrows the scope to one managed volume, and

path (optional, 1-1024 characters) narrows the scope further to a specific virtual path. This is relative to the ***vol-path***.

Before clearing the records, a prompt requests confirmation; enter **yes** to confirm. After you clear the records, they no longer appear in the show sync output.

For example, this command sequence shows all sync-file records, clears them, then shows that there are none remaining:

```
bstnA(gbl)# show sync files

Namespace: wwmed
    Volume: /acct

...

            Found Items:        0
            Synchronized Items: 1
            Processed Items:    245

bstnA(gbl)# end
bstnA# clear sync files
Clear sync records? [yes/no] yes
bstnA# show sync files
bstnA# ...
```

Canceling a Sync Operation

To cancel an in-progress sync operation, go to priv-exec mode and use the cancel sync command:

cancel sync {files | directories} ***namespace*** **volume** ***vol-path*** **path** ***path***

where

files | directories chooses the type of sync operation to cancel,

namespace (1-30 characters) identifies the namespace,

vol-path (1-1024 characters) is the volume, and

path (1-1024 characters) is the specific virtual path that is being synchronized. This is relative to the ***vol-path***.

A prompt requests confirmation before the CLI cancels the sync; enter **yes** to confirm that you want to cancel the operation.

For example, this command cancels a sync-files operation in the “wwmed” namespace:

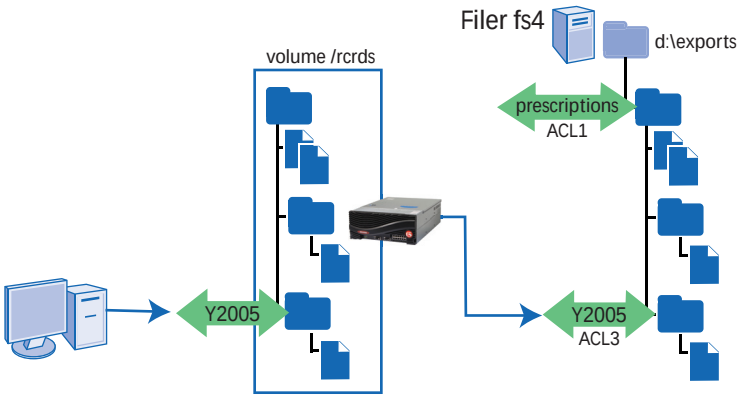
```
bstnA(gbl)# end
bstnA# cancel sync files wwmed volume /acct path /
```

```
Cancel specified sync operations? [yes/no] yes
bstnA# ...
```

Adding and Synchronizing Filer Subshares (CIFS)

This section only pertains to volumes that support CIFS. Skip to the next section for NFS-only volumes.

A *filer subshare* is a shared directory that is under another shared directory. A CIFS service and its managed volume can pass clients from a front-end subshare through to the corresponding subshare on a back-end filer. When clients access subshares this way, the filer can enforce the subshare's share-level ACL. For example, clients that connect to the "Y2005" share (below) use ACL3; by default, when subshares are not supported, the volume connects to the filer's top-level share (through ACL1 in this illustration) and descends to the directory from there.



Two sections of the *ARX® CLI Storage-Management Guide* have instructions for providing this CIFS-subshare support: *Supporting Subshares and their ACLs*, on page 9-20 explains how to set up managed volumes, and *Exporting a Filer Subshare (and Using its ACL)*, on page 11-16 describes the commands to set up a CIFS service. These instructions explain how to work with subshares that exist on the filers *before* the managed volume imports them.

The supported method for adding *new* subshares is to directly access a back-end filer, add the subshare there, then synchronize the metadata with the new subshare.

◆ Note

This is contrary to the standard best practice. Share definitions and ACLs are not managed by the volume, so this does not cause any metadata inconsistencies. You can also create the underlying directory for the new subshare.

Ensure the consistency of back-end subshares that correspond to front-end CIFS service exports by using subshare synchronization. Refer to *Synchronizing Subshares*, on page 9-25 of the *ARX® CLI Storage-Management Guide* for complete details.

Important

If the volume is backed by any NetApp filers or EMC servers and you plan to support free-space quotas on them (using freespace cifs-quota*), prepare the subshares directly on the filer before you synchronize subshares. The subshare-replication process does not create NetApp qtrees or EMC filesystems, and those special directories are required to support free-space quotas. Before you run subshare replication, access the filers directly and create one qtree or filesystem per subshare. Then, synchronize their share-level ACLs and other attributes.*

Rebuilding a Namespace

The *nsck rebuild* utility completely rebuilds all managed volumes in a namespace: it re-imports the back-end shares and rebuilds its metadata databases. This command causes a brief interruption in service, and it forces all NFS clients to unmount and then re-mount their managed volumes; use this only in situations where sync files fails. This has no effect on direct volumes.

◆ Note

By default, this stops all modifications (client writes and file renames on import) in all managed volumes after the rebuild. Essentially, the nsck utility issues a no modify *for every managed volume in the namespace.*

To guarantee that managed-volume modifications are enabled after the rebuild, run the modify *command in each volume and raise the* reimport-modify *flag before you rebuild the namespace. The* reimport-modify *flag stops the nsck utility from running* no modify.

If you do not do this but the rebuild/import succeeds without any collisions, you can use the modify *command on each volume after the rebuild.*

For details on the reimport-modify *command, see Allowing the Volume to Modify on Re-Import, on page 9-8 of the ARX® CLI Storage-Management Guide.*

From priv-exec mode, use the nsck rebuild command to rebuild all managed volumes in a namespace:

```
nsck namespace rebuild
```

where ***namespace*** (1-30 characters) is the name of the namespace to rebuild. Use the show namespace command to see a list of configured namespaces.

The CLI prompts for confirmation before re-importing all managed volumes in the namespace: enter **yes** to continue. Enter **no** to cancel the rebuild.

You can run one nsck job at a time on a given volume. Use show nsck to monitor the progress of nsck jobs; see *Showing the Progress of nsck and sync Jobs*, on page 7-4.

For example, the following command sequence exits to priv-exec mode and rebuilds the ‘wwmed’ namespace.

```
bstnA(gbl)# end
bstnA# nsck wwmed rebuild
Reimport all shares in the namespace wwmed? [yes/no] yes
bstnA# ...
```

Directory Mastership After a Rebuild

A managed volume sometimes duplicates (or stripes) its directories on multiple shares. Each directory has a single *master*, and may have one or more *stripes*. The master directory is on the share where it was first found, and each stripe is a copy on another share. If a filer share has the master instance of a directory, the directory tends to grow on that share; by default, a new file goes to the same share as its parent's master directory. For example, if a client creates a "newfile.doc" file in "\mydir," and "\mydir" has its master on the nas20 filer, the volume sends "newfile.doc" to nas20 by default.

A file-placement rule can change the placement of master directories during import, normal operation, and re-import. The *ARX® CLI Storage-Management Guide* describes how to use these rules to set master-directory placement: refer to *Matching Directories Only*, on page 14-7 and *Matching and Promoting Directories*, on page 14-9. Additionally, you can set the priority of the volume's shares so that the highest-priority shares have mastership from the beginning of the import; see *Setting the Share's Priority (for Tiering)*, on page 9-36 of the same storage manual. If no such rules or priorities are configured for master-directory placement, mastership of directories is not deterministic after the rebuild. After the rebuild is finished, you can correct this by adding a new file-placement rule.

Rebuilding a Volume

You can narrow the focus of an nsck rebuild to a single volume. To rebuild one volume, add the optional volume clause to the end of the nsck rebuild command:

```
nsck namespace rebuild volume volname
```

where:

name (1-30 characters) is the name of the namespace to rebuild. Use the show namespace command to see a list of configured namespaces.

volume *volname* (optional, 1-256 characters) limits the rebuild to the chosen volume (for example, '/home').

The CLI prompts for confirmation before re-importing the volume's shares: enter **yes** to continue. Enter **no** to cancel the rebuild.

◆ Note

As mentioned above, this stops all managed-volume modifications (client writes and file renames on import) after the rebuild. Essentially, the nsck utility issues a no modify *for the volume.*

To guarantee that modifications are enabled after the rebuild, run the modify *command and raise the* reimport-modify *flag before you rebuild the volume. The reimport-modify flag stops the nsck utility from running* no

modify *on the volume.*

If you do not do this but the rebuild/import succeeds without any collisions, you can use the modify *command after the rebuild.*

For example, the following command sequence exits to priv-exec mode and rebuilds the '/etc' volume in the 'archives' namespace:

```
bstnA(gbl)# end
bstnA# nsck archives rebuild volume /etc
Reimport all shares from volume /etc in namespace archives? [yes/no] yes
bstnA# show nsck

Op Id  Op Type      Namespace:Path                          Status
------ ------------ --------------------------------------- ------------------
...
47     report       archives:/etc                           Rebuild in progress
bstnA# ...
```

Forcing the Rebuild

It is possible for an nsck-rebuild job to freeze in an unfinished state: for example, the job freezes if one of the back-end filers goes down during the re-import. In this situation, you should resolve the problem and then restart the rebuild. To override the unfinished rebuild job and start a new one, add the optional force flag to the nsck rebuild command.

nsck *namespace* rebuild [volume *volname*] force

For example, the following command sequence exits to priv-exec mode and forces a rebuild for the 'archives' namespace:

```
bstnA(gbl)# end
bstnA# nsck archives rebuild force
Reimport all shares in the namespace archives? [yes/no] yes
bstnA# ...
```

De-Staging a Namespace's Managed Volumes

You may want to perform filer recoveries directly on the filer. To directly access the shares behind a namespace's managed volumes, you first release (or *de-stage*) the shares from the volumes. This removes all metadata from all managed volumes and then disables all of the volumes' shares. Once the volumes are de-staged, you can access all of their shares directly (that is, not through a VIP) without any risk of metadata inconsistencies. After you restore all files to the shares, you can re-enable all of the namespace's managed volumes.

As with a rebuild, NFS clients must unmount and remount the volumes after the shares are re-enabled.

From priv-exec mode, use the nsck destage command to release all shares from a namespace's managed volumes.

nsck *namespace* destage

where:

namespace (1-30 characters) is the name of the namespace to be destaged. Use the show namespace command to see a list of configured namespaces.

destage specifies to destage the namespace metadata and prepare it for reimport.

The CLI prompts for confirmation before releasing the shares: enter **yes** to continue. Enter **no** to cancel the destage operation.

Important

While the namespace is de-staged, its managed volumes are inaccessible to clients.

Note

As with nsck ... rebuild*, this disables all modifications (client writes and file renames on import) in the namespace's managed volumes. That is, no modifications are possible after the managed volumes are later re-enabled. To prevent this, someone must first run the* modify *command and raise the* reimport-modify *flag.*

You can run one nsck job at a time on a given volume. Use show nsck to monitor the progress of nsck jobs; see *Showing the Progress of nsck and sync Jobs*, on page 7-4.

For example, the following command sequence exits to priv-exec mode and destages the "wwmed" namespace:

```
bstnA(gbl)# end
bstnA# nsck wwmed destage

The namespace is in use by NFS global service 'ac1.medarch.org'.

This operation will remove entries for all shares
from the namespace metadata.  To reimport a destaged share,
```

```
enable the share.

% WARNING: Namespace wwmed is in use by global services.

Destaging the namespace will disrupt all clients using these services.

Destage the namespace anyway? [yes/no] yes
bstnA# ...
```

De-Staging the Volumes in a Volume Group

Some metadata failures require a destage of all volumes in the same volume group. Volume groups are a means of isolating namespaces in the ARX's memory so that the failure of one or more namespaces in one volume group does not affect the performance of namespaces in other volume groups. The *ARX® CLI Storage-Management Guide* describes volume groups in detail: refer to *Assigning The Volume To A Volume Group (optional)*, on page 9-48.

For example, suppose a volume uses an NFS metadata share, and the mount to that metadata share hangs. The volume (which could support NFS, CIFS, or both for its clients) gradually goes offline. Other volumes in the same volume group will fail eventually, also. The solution for this problem is to restart all of the volumes in the volume group, which results in a re-import of the volumes. You can destage all of them without affecting any volumes in other volume groups. Then, you can use the priv-exec reload command to fail over to the peer ARX and restart all the volumes. Only the volumes in the affected volume group must re-import.

To destage the volumes in one volume group, add the optional volume-group argument to the end of the nsck destage command:

nsck *namespace* destage volume-group *volumegroupid*

where:

namespace (1-30 characters) is the name of the namespace where the destage occurs. Use the show namespace command to see a list of configured namespaces.

destage specifies to destage the namespace metadata and prepare it for reimport.

volume-group *volumegroupid* (optional, 1-8) identifies a volume group used by this namespace. You can use the show volume-group command for a full list of the volume groups on this ARX, and to find which volume groups are assigned to this namespace.

The CLI prompts for confirmation before releasing the volumes: enter **yes** to continue. Enter **no** to cancel the destage operation. If any of the volumes' metadata shares support NFS and are unavailable, the prompt recommends a reload to initiate a failover; this is required to correct a hung NFS mount at a metadata share, as described above.

You can run one nsck job at a time on a given volume. Use show nsck to monitor the progress of nsck jobs; see *Showing the Progress of nsck and sync Jobs*, on page 7-4.

For example, the following command sequence exits to priv-exec mode, shows all volume groups, and releases all volumes running on volume group 2:

```
bstnA(gbl)# end
bstnA# nsck wwmed destage volume /it
bstnA# show volume-group

Switch: bstnA
----------------------------------------------------------------------
Volume Group 1
-----
Physical Processor: 1.1
State: Normal; maximum instances
Share credits: 4 shares used (124 credits remain of total 128)
Direct share credits: 3 direct shares used (2045 credits remain of total 2048)
Volume credits: 2 volumes used (30 credits remain of total 32)
File credits: 4.0 M files reserved (252 M credits remain of total 256 M)

Namespace         Domain  Volume              State
---------         ------  ------              -----
medco              2      /vol                Enabled
wwmed              1      /acct               Enabled

2 Namespaces            2 Volumes

Volume Group 2
-----
Physical Processor: 1.1
State: Normal; maximum instances
Share credits: 9 shares used (119 credits remain of total 128)
Direct share credits: 6 direct shares used (2042 credits remain of total 2048)
Volume credits: 6 volumes used (26 credits remain of total 32)
File credits: 72 M files reserved (184 M credits remain of total 256 M)

Namespace         Domain  Volume              State
---------         ------  ------              -----
medarcv            1      /rcrds              Enabled
medarcv            1      /lab_equipment      Enabled
medarcv            1      /test_results       Enabled
insur              2      /claims             Enabled

2 Namespaces            4 Volumes

...

bstnA# nsck medarcv destage volume-group 2

This operation will remove entries for all shares in the volume group
from the namespace metadata.  To reimport a destaged share,
enable the share.  The following volumes will be affected:

/rcrds
/lab_equipment
/test_results


Destage volumes hosted by volume group 2 in the namespace "medarcv"? [yes/no] yes
```

```
bstnA# ...
```

De-Staging a Single Volume

You can narrow the focus of an nsck destage even further, to a single volume. To destage one volume, add the optional volume clause to the end of the nsck destage command:

nsck *namespace* destage volume *volname*

where:

namespace (1-30 characters) is the name of the namespace where the destage occurs. Use the show namespace command to see a list of configured namespaces.

destage specifies to destage the namespace metadata and prepare it for reimport.

volume *volname* (optional, 1-256 characters) is the volume to destage (for example, '/var').

The CLI prompts for confirmation before releasing the volume: enter **yes** to continue. Enter **no** to cancel the destage operation.

You can run one nsck job at a time on a given volume. Use show nsck to monitor the progress of nsck jobs; see *Showing the Progress of nsck and sync Jobs*, on page 7-4.

For example, the following command sequence exits to priv-exec mode and releases a volume from the "wwmed" namespace:

```
bstnA(gbl)# end
bstnA# nsck wwmed destage volume /it

Volume /it is in use by NFS global service 'ac1.medarch.org'.

This operation will remove entries for all shares in the volume
from the namespace metadata.  To reimport a destaged share,
enable the share.

% WARNING: Volume /it in namespace wwmed is in use by global services.

Destaging the volume will disrupt all clients using this volume.

Destage the volume anyway? [yes/no] yes
bstnA# ...
```

Forcing the De-Stage

As mentioned in the section about rebuilding a namespace, it is possible for filer error or network glitch to freeze an nsck job in an unfinished state. Once the error is resolved, you may want to de-stage the volume or namespace; however, nsck only allows a single job to run on a given volume. To override the unfinished nsck job and start a new one, add the optional force flag to the nsck destage command.

nsck *name* destage [volume-group *volumegroupid*| volume *volname*] force

For example, the following command sequence exits to priv-exec mode and forces a de-stage for the 'wwmed~/it' volume:

```
bstnA(gbl)# end
bstnA# nsck wwmed destage volume /it force
Volume /it is in use by NFS global service 'ac1.medarch.org'.

This operation will remove entries for all shares in the volume
from the namespace metadata.  To reimport a destaged share,
enable the share.

% WARNING: Volume /it in namespace wwmed is in use by global services.

Destaging the volume will disrupt all clients using this volume.

Destage the volume anyway? [yes/no] yes
bstnA# ...
```

Re-Enabling the Shares

With the share released from its namespace, you can access the share directly at its filer. Once you are finished, you must re-enable all shares in the volume(s) to restart the import process.

When the shares are ready, re-enable each of them to import them back into the namespace. From gbl-ns-vol mode, you can use enable shares to enable all of the volume's shares at once; see *Enabling All Shares in the Volume*, on page 9-54 of the *ARX® CLI Storage-Management Guide*.

Recall that the modify command is disabled by nsck ... destage unless the reimport-modify flag was also raised. If the reimport-modify flag was down during the destage, you have the opportunity to enable modify before enabling shares (see *Allowing the Volume to Modify on Import*, on page 9-6 of the *ARX® CLI Storage-Management Guide*).

To continue the example:

```
bstnA# global
bstnA(gbl)# namespace wwmed
bstnA(gbl-ns[wwmed])# volume /it
bstnA(gbl-ns-vol[wwmed~/it])# modify
Automatically re-enable volume modify mode during nsck rebuild? [yes/no] yes
bstnA(gbl-ns-vol[wwmed~/it])# enable shares
bstnA(gbl-ns-vol[wwmed~/it])# ...
```

Migrating Metadata to a New Back-End Share

Managed volumes require fast and extremely-reliable back-end shares to store their metadata. A slow metadata share can affect client service as well as import speed. To migrate a volume's metadata from a slow back-end share to one on a faster filer, use the nsck migrate-metadata command:

nsck *namespace* migrate-metadata volume *vol-path* target *ext-filer* {nfs3 | nfs3tcp | cifs} *path*

where:

namespace (1-30 characters) identifies the namespace.

vol-path (1-1024 characters) is the managed volume that requires new metadata storage. Use forward slashes (/) in the path, even for CIFS volumes.

ext-filer (1-64 characters) identifies the target share for the volume's metadata. This is the external-filer name for the filer; use show external-filer for a complete list of configured filers.

nfs3 | nfs3tcp | cifs is a required choice. This is the protocol used to access the metadata share; it can be different from any of the protocols used for the managed volume. The choices are NFSv3 over UDP (nfs3), NFSv3 over TCP (nfs3tcp), or CIFS.

path (1-1024 characters) selects the metadata share and path (for example, '/' or '/usr/acoMD'). For a CIFS share, this is the share name (and possibly a path to a subshare); use forward slashes (/), even for CIFS shares.

Important

This stops all client access to the volume during the metadata migration. Whether or not the migration succeeds, client access resumes as soon as the operation ends.

The CLI prompts for confirmation before it starts the migration; enter **yes** to proceed. The nsck utility copies all of the metadata to the target share, verifies its integrity at the target share, then switches the managed volume over to the new metadata share. If the migration fails in any way before the last step, the volume comes back online with its metadata still on the original share. You must manually restart the migration after a cancellation or unexpected failure.

As with all nsck jobs, this operation produces a report to show its progress. The CLI shows the report name after you invoke the command. Use show reports for a full list of nsck reports, and use show, tail, or grep to view one report. The report shows the source and target shares, the size of the metadata, and the time required to migrate it.

For example, this command sequence migrates metadata for the wwmed~/acct volume and then shows the report:

```
bstnA(gbl)# end
bstnA# nsck wwmed migrate-metadata volume /acct target nas3 nfs3 /vol/vol2/meta7
```

```
Volume /acct is in use by NFS global service 'ac1.medarch.org'.
Volume /acct in namespace wwmed is in use or being browsed by global services.
  Migrating metadata for this volume will disrupt clients using these services.
  No other volume will be affected by this procedure.

Proceed with metadata migration? [yes/no] yes
Starting metadata migration for wwmed:/acct. 1835008 bytes to be transferred.
Detailed status contained in report: migrate_metadata.34.rpt on switch bstnA
bstnA# show reports migrate_metadata.34.rpt
**** Metadata Migration Report: Started at Wed Feb 24 02:15:36 2010 ****
**** Software Version: 5.02.000.12538 (Feb 16 2010 20:14:13) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:

Namespace:                  wwmed
Volume:                     /acct

Original Location:          nas1(192.168.25.21): /vol/vol1/meta1 (nfs3)
Target Location:            nas3(192.168.25.47): /vol/vol2/meta7 (nfs3)

Metadata Size:              2,232,320 (2.1 MB)

**** Total processed:              0
**** Elapsed time:          00:00:04
**** Metadata Migration Report: DONE at Wed Feb 24 02:15:40 2010 ****
```

Canceling a Metadata Migration

You can cancel a migration any time before it has restarted the volume. The restart occurs after the metadata is successfully copied to the new share. To cancel a metadata migration in an earlier phase, go to priv-exec mode and use the cancel migrate-metadata command:

cancel migrate-metadata *namespace* volume *vol-path*

where

namespace (1-30 characters) identifies the namespace,

vol-path (1-1024 characters) is the volume, and

A prompt requests confirmation before the CLI cancels the migration; enter **yes** to confirm.

For example, this command cancels a metadata migration for the "medarcv~/rcrds" volume:

```
bstnA(gbl)# end
bstnA# cancel migrate-metadata medarcv volume /rcrds
Are you sure you want to cancel this migration? [yes/no] yes
Metadata migration cancelled successfully.
bstnA# ...
```

Clearing All nsck Jobs

From priv-exec mode, use the clear nsck command to clear all the nsck jobs that are currently pending, and to clear all finished jobs from the job history in show nsck:

clear nsck

The CLI prompts for confirmation before clearing all jobs; enter yes to clear all nsck jobs. This does not delete the nsck reports; use delete reports *nsck-report-name* to remove an nsck report, or show reports to see all reports.

For example:

```
bstnA(gbl)# end
bstnA# clear nsck
Clear all nsck records? [yes/no] yes
bstnA# show nsck

Op Id  Op Type      Namespace:Path                          Status
------ ------------ --------------------------------------- ------------------
bstnA# ...
```

Clearing One nsck Job

Use the nsck-job ID with the clear nsck command to clear one pending (or finished) nsck job:

clear nsck *job-id*

where ***job-id*** (1-2,147,483,647) identifies the nsck job to clear. This ID appears in the name of the nsck report.

As above, the CLI prompts for confirmation before clearing the job. Enter **yes** to proceed.

For example, the following command sequence stops nsck-job 1, which was still pending:

```
bstnA(gbl)# show nsck

Op Id  Op Type      Namespace:Path                          Status
------ ------------ --------------------------------------- ------------------
1      report       wwmed:/acct                             Pending
bstnA(gbl)# end
bstnA# clear nsck 1
Clear NSCK ID 1? [yes/no] yes
bstnA# ...
```

Truncating a Report

To conserve CPU cycles and/or internal-disk space, you may want to stop a report before it is finished. A oversized, CPU-intensive report could possibly have an effect on namespace performance. You can stop the report without stopping the nsck job. From priv-exec mode, use the truncate-report command to stop all report processing and truncate the report file:

truncate-report ***report-name***

where ***report-name*** (1-255 characters) specifies report to truncate.

The CLI prompts for confirmation before truncating the report. Enter **yes** to continue.

For example, the following command sequence truncates an inconsistencies report:

```
bstnA(gbl)# end
bstnA# truncate report inconsistencies.29.rpt
Truncate report ''inconsistencies.29.rpt''? [yes/no] yes
bstnA# ...
```

8

Troubleshooting Tools

Overview

This chapter describes the basic tools and procedures for troubleshooting the ARX.

- You can get a list of all alarm conditions with the show health command.
- The show health time-skew command shows the clocking skew between the ARX and its network peers.
- The ARX keeps its logs in a syslog file. You can access the syslog file by using show, grep, and tail.
- You can forward all syslog messages (as they are logged) to one or more external hosts.
- You can show the current tasks running on the system.
- You can collect all diagnostic information from a troubled switch and send it to a remote site. This is for sending information to F5 for thorough diagnosis.
- Using SSH, you can run show commands from a remote host.

These are basic tools for troubleshooting the ARX. Later chapters describe procedures for troubleshooting network connections and managed volumes.

Showing All Active Alarms

Several of the ARX's Enterprise SNMP traps signal that an alarm condition is now active. This indicates a system failure, or that the system has crossed a resource threshold of some kind. If the condition clears or the system drops back below the threshold, another trap clears the alarm condition. From any mode, you can use the show health command to show all currently-active alarms on the system:

```
show health
```

Refer to the *ARX SNMP Reference* for detailed information about any trap, including correction procedures.

For example, this shows current alarm conditions on the ARX named "bstnA:"

```
bstnA(cfg)# show health

System Health Information
Date                     ID      Event                       Description
-------------------------------------------------------------------------------
Mon Jan 24 00:50:27 2011 (540) - powerFail                   Slot power 1/2 absent
Mon Jan 24 00:50:27 2011 (540) - powerFail                   Slot power 1/1 absent
Mon Jan 24 00:52:40 2011 (185) - gatewayOffline              Client gateway 10.1.11.218 is
offline.
Mon Jan 24 00:54:52 2011 (411) - nsmStandby                  Processor 2.2 is in standby.
Mon Jan 24 00:55:04 2011 (411) - nsmStandby                  Processor 2.4 is in standby.
Mon Jan 24 00:55:04 2011 (411) - nsmStandby                  Processor 2.3 is in standby.
Mon Jan 24 00:55:04 2011 (411) - nsmStandby                  Processor 2.1 is in standby.
Mon Jan 24 01:00:51 2011 (826) - virtualServiceAclUpdateFail 192.168.25.15:/acct:NIS information
has not been fully resolved by the NIS daemon.
Mon Jan 24 01:02:17 2011 (148) - downRevAdForestLevelRaise   The forest functional level of domain
MEDARCH.ORG is not capable of supporting constrained delegation
Mon Jan 24 01:09:43 2011 (19)  - archiveFreeSpaceThresholdRaise fileRecordsMed,
fs4:arx_file_archv:is running low on freespace (<10 GB Free).
Mon Jan 24 01:20:05 2011 (826) - virtualServiceAclUpdateFail 192.168.25.15:/claims:NIS information
has not been fully resolved by the NIS daemon.
bstnA(cfg)# ...
```

Clearing SNMP Traps

After you have investigated an SNMP trap and resolved the underlying issue, you can dismiss the trap using the privileged-exec mode CLI command clear health. The command syntax is:

```
clear health trap
```

where *trap* is the trap or event name.

For example, the following command dismisses the filerAccessDenied trap:

```
bstnA# clear health filerAccessDenied
```

Showing Time Skews Between the ARX and Other Servers

Several ARX processes depend on time synchronization with other servers in the network. For example, some migration policies are based on last-modified or last-accessed times, and Kerberos authentication uses tickets with expiration times. If the ARX time is drastically different from DC time or filer time, these features behave unpredictably.

We recommend using NTP to synchronize time between the ARX and other network devices. For information on connecting the ARX to your NTP server(s), refer to *Configuring NTP*, on page 4-15 of the *ARX® CLI Network-Management Guide*.

From any mode, you can use the show health time-skew command to see the time skew between the ARX, its DCs, and its back-end filers:

show health time-skew [timeout *seconds*]

where timeout *seconds* (optional, 1-3600) specifies the maximum time to wait for responses from each server. The default is 3 seconds. The ARX often makes two or more connection attempts to each server, and this timeout applies to each attempt. Therefore, the command could take twice as long (or longer, in some cases) than the timeout you set here. Also, this timeout applies to each individual server, so the overall time for multiple servers may be longer. You can always use **<Ctrl-C>** to return to the CLI prompt before the command finishes.

The output is a table with one row per external device, showing the device's IP address, its role for the ARX ("External Filer" or "Domain Controller"), and its time skew with respect to ARX time.

For example, this command shows the time skews between the "bstnA" switch and the network devices that it uses:

```
bstnA(cfg)# show health time-skew
IP Address            Role                 Observed Time Skew vs. ARX
--------------------  -------------------  ----------------------------
192.168.25.22         External Filer       00:18:41  ahead
192.168.25.24         External Filer       00:00:00
192.168.25.25         External Filer       04:16:19  behind
192.168.25.48         External Filer       3 d 03:59:15 ahead
192.168.25.22         External Filer       00:00:00
192.168.25.44         External Filer       00:00:00
192.168.25.19         External Filer       00:00:00  ahead
192.168.25.71         External Filer       00:00:00
192.168.25.27         External Filer       00:01:29  behind
192.168.25.20         External Filer       00:01:29  behind
192.168.25.28         External Filer       00:01:29  behind
192.168.25.29         External Filer       00:01:29  behind
192.168.25.21         External Filer       00:00:00
192.168.25.47         External Filer       00:00:00
192.168.25.51         External Filer       00:00:00
192.168.25.102        Domain Controller    00:01:29  behind
192.168.25.104        Domain Controller    00:01:29  behind
192.168.202.9         Domain Controller    00:01:30  behind
192.168.202.10        Domain Controller    00:01:30  behind
192.168.202.11        Domain Controller    00:01:29  behind
```

```
172.16.124.73         Domain Controller     00:01:29  behind
192.168.25.103        Domain Controller     00:01:29  behind
172.16.168.21         Domain Controller     00:01:29  behind
10.19.230.94          Domain Controller     00:01:30  behind
192.168.25.109        Domain Controller     00:01:30  behind
192.168.25.110        Domain Controller     00:01:30  behind
172.16.108.139        Domain Controller     00:00:41  behind
172.16.110.5          Domain Controller     00:00:41  behind
172.16.120.5          Domain Controller     00:00:41  behind
172.16.213.9          Domain Controller     00:01:06  behind
172.16.240.88         Domain Controller     00:01:07  behind
172.16.210.7          Domain Controller     00:01:06  behind
172.16.74.88          Domain Controller     00:01:30  behind
bstnA(cfg)# ...
```

Accessing the Syslog

The *syslog* (short for System Log) contains all log messages from the system's internal software. Processes add informational messages to this log as they run, as well as error messages of varying severities. For an alphabetical list of syslog messages, see the *ARX Log Catalog*.

The syslog is rotated by the system whenever it exceeds 75 Megabytes. The rotation process renames the current syslog file "syslog.1.log," the current syslog.1.log becomes syslog.2.log, and so on. The oldest log kept is syslog.100.log; older logs are deleted. To see a list of all syslogs on the current system, use the show logs command:

```
show logs
```

For example, this command is run on a system with an active syslog and no backups:

```
bstnA(cfg)# show logs


  logs
      catalina.out            Nov 21 03:27     0
      dfpd.dump.0             Nov 20 14:31   419
      dmesg                   Nov 21 03:23   24k
      error.log               Nov 21 03:59  897k
      fastpath                Nov 21 03:51   363
      firebird.log            Nov 21 03:23   26k
      gziplog                 Nov 20 14:32    96
      kernel.log              Nov 21 03:51  583k
      megaserv.log            Nov 21 03:54  9.5k
      nisd_dump.0             Nov 21 03:28   283
      nlmd.dump.0             Nov 20 14:31  2.7k
      nlmd.dump.1             Nov 20 14:31  2.7k
      om.install.log          Nov 21 02:39    53
      om.upgrade.log          Nov 21 02:40  176k
      originalOidList.txt     Nov 21 03:24   23k
      pxlog.log               Nov 21 03:27  109k
      ramecc.log              Nov 21 03:24   10k
      rtmd.dump.0             Nov 20 14:31   698
      scm-top.dump            Nov 21 03:43  5.7M
      scm-vmstat.dump         Nov 21 03:59   18k
      statsd.dump.0           Nov 20 14:31   20k
      syslog                  Nov 21 03:59   13M
      traplog                 Nov 21 03:56  7.8k
      traplog.1               Nov 21 03:38   26k
      traplog.10              Nov 20 13:52   25k
      traplog.2               Nov 20 17:05   25k
      traplog.3               Nov 20 16:28   25k
      traplog.4               Nov 20 15:59   25k
      traplog.5               Nov 20 15:30   25k
      traplog.6               Nov 20 15:01   25k
      traplog.7               Nov 20 14:32   25k
      traplog.8               Nov 20 14:12   25k
      traplog.9               Nov 20 13:57   28k
      txnmond.dump.0          Nov 20 14:28     0
```

Tailing the Syslog

You can use the tail command from any mode to view syslog messages as they are added to the log:

tail logs *file-name* follow

where

logs is required to specify the "logs" directory,

file-name (1-255 characters) identifies the file to tail, and

follow is required to follow the syslog as it grows.

Use tail logs syslog follow to view the syslog file. Use <Ctrl-C> to stop the tail output and return to the CLI prompt.

For example:

```
prtlndA> tail logs syslog follow
2004-06-18T18:26:33.038-0400:bstnA:1-1-SCM-0:LOG-0-6-FILTER:: sshd[5691]: Accepted
keyboard-interactive for admin from 172.16.22.183 port 58849 ssh2
2004-06-18T18:26:33.040-0400:bstnA:1-1-SCM-0:LOG-0-6-FILTER:: sshd[5691]: lastlog_get_entry: Error
reading from /var/log/lastlog: Bad file descriptor
2004-06-18T18:26:48.294-0400:bstnA:1-1-SCM-960:L2SW_DRV-0-5-MSG5:: Disable Mirror Port:
2004-06-18T18:28:39.195-0400:bstnA:1-1-SCM-0:LOG-0-4-FILTER:: sntpc: resetting on error 0.117 >
0.102
2004-06-18T18:28:39.222-0400:bstnA:1-1-SCM-0:LOG-0-4-FILTER:: sntpc: resetting on error 0.117 >
0.102
2004-06-18T18:31:22.008-0400:bstnA:5-2-ASM_FC-602:LIBCIFS-0-7-MSG7:: connection to
\\10.51.1.23\histories deleted
2004-06-18T18:37:13.257-0400:bstnA:1-1-SCM-0:LOG-0-6-FILTER:: sshd[9204]: WARNING:
/usr/local/etc/moduli does not exist, using old modulus
2004-06-18T18:37:13.371-0400:bstnA:1-1-SCM-0:LOG-0-6-FILTER:: sshd[9204]: Could not reverse map
address 172.16.22.183.
2004-06-18T18:37:14.028-0400:bstnA:1-1-SCM-0:LOG-0-6-FILTER:: sshd[9204]: Accepted
keyboard-interactive for admin from 172.16.22.183 port 60054 ssh2
2004-06-18T18:37:14.032-0400:bstnA:1-1-SCM-0:LOG-0-6-FILTER:: sshd[9204]: lastlog_get_entry: Error
reading from /var/log/lastlog: Bad file descriptor
...
```

Showing the End of the Syslog

You can show the last few lines in the syslog file, without following any new syslog entries, by omitting the follow keyword:

tail logs *file-name* [*number-lines*]

where

file-name (1-255 characters) identifies the syslog file to tail, and

number-lines (optional, 1-4096) is the number of lines to show. By default, the command shows the last 24 lines in the file.

For example, the following command shows the last five lines in the current syslog file:

```
bstnA> tail logs syslog 5
2004-06-18T18:31:22.008-0400:bstnA:5-2-ASM_FC-602:LIBCIFS-0-7-MSG7:: connection to
\\10.51.1.23\histories deleted
2004-06-18T18:37:13.257-0400:bstnA:1-1-SCM-0:LOG-0-6-FILTER:: sshd[9204]: WARNING:
/usr/local/etc/moduli does not exist, using old modulus
```

```
2004-06-18T18:37:13.371-0400:bstnA:1-1-SCM-0:LOG-0-6-FILTER:: sshd[9204]: Could not reverse map
address 172.16.22.183.
2004-06-18T18:37:14.028-0400:bstnA:1-1-SCM-0:LOG-0-6-FILTER:: sshd[9204]: Accepted
keyboard-interactive for admin from 172.16.22.183 port 60054 ssh2
2004-06-18T18:37:14.032-0400:bstnA:1-1-SCM-0:LOG-0-6-FILTER:: sshd[9204]: lastlog_get_entry: Error
reading from /var/log/lastlog: Bad file descriptor
bstnA>
```

Log Components

Every log message originates from a log component (such as SCM_CLI in the sample above). A *log component* is a source of syslog messages, typically an internal process or group of processes. The table below is an alphabetical list of all log components, with a brief description and an indication of the board(s) where the software runs.

Log Component	Description
AFFINITY	Affimgrd and libaffin, which manage the assignment of internal processes to CPU cores.
AFN_NET	IPC Abstraction Layer, for communication between internal ARX processes.
API	API Services.
APP_RON_RTMD	RON Route Table Maintenance Daemon.
AT	AT Library. This is deprecated, so it is unlikely to yield any log messages.
AUTH_CONSOLE	Console Authentication. This component performs authentication for administrators who log into the Console (serial) port.
AUTH_HTTP	HTTP Authentication. This component performs authentication for administrators who access the GUI through HTTP.
AUTH_HTTPS	HTTPS Authentication. This component performs authentication for administrators who access the GUI through HTTP over SSL, or HTTPS.
AUTH_SSH	SSH Authentication. This component performs authentication for administrators who access the CLI through Secure SHell (SSH).
AUTH_TELNET	Telnet Authentication. This component performs authentication for administrators who access the CLI through Telnet.
AUTO_DIAG_CLI	Auto Diagnostics, which gathers input/output statistics and monitors for service degradation.
BESPD	Back-end Share Probing daemon, which periodically probes the shares behind ARX volumes and checks their health.
BOOTPD	BOOTP Daemon, or the boot-protocol server, which manages ARX boots and reboots.
CATALOG_INTERNAL	Message Catalog Internal Errors.
CHASS	Chassis Manager, which implements chassis operations such as 'reload,' 'firmware upgrade,' and hardware monitoring.

Log Component	Description
CHASS_LIB	Chassis Manager Library. The chassis manager implements chassis operations such as 'reload,' 'firmware upgrade,' and hardware monitoring.
CIFS_CLI	CIFS CLI.
CIFS_SHIM_LIB	Interface to metadata storage on CIFS filers. Metadata is data about managed-volume files, such as their locations on back-end filers.
CIFSCONFD	CIFS Configuration Daemon, which manages one front-end CIFS service.
CLI	Command Line Interface.
CLI_NFS_COPY	Copy to/from NFS namespace.
CLI_RUN_EXIM	Run exim4 from the CLI..
CLI_WIZARD	Initial Interview, which prompts with questions and performs initial configuration the first time the ARX boots.
CLUSTER	Not supported.
COMMON_DNAS	Common DNAS Messages.
COMMON_NIS	Common Network Information System (NIS) messages. NIS servers provide a lookup service for NFS servers, to manage large NFS access lists.
COMMONLIB	Not supported.
CORE	Core collector.
CRMD	Critical Resource Monitor Daemon, which makes HA failover decisions based on external filers, routers, etc.
DDBD	Distributed Database Library. This is for the internal database, which stores the ARX configuration. You can see the full configuration with 'show running-config' and 'show global-config'.
DEBUG	Internal debugging - used for development only.
DFPD	DNAS Fast Path Manager Daemon, which configures managed volumes in the NSM data plane.
DME	Policy Data Mover Engine (DME). This implements file migrations from one back-end filer to another.
DMOUNTD	Distributed Mount Daemon for NFS mounts.
DMSG_LIB	DNAS Message Library.
DNAS_IMPORT	DNAS Import Messages. An import occurs when a filer share is enabled in an enabled managed volume. This is the process of walking the directory tree of the share and recording all of its files and directories in managed-volume metadata.
DNAS_MD_MIGRATE	DNAS Metadata Migration, which moves managed-volume metadata from one external filer to another.
DNAS_PGRP	DNAS Process Group and NFS. DNAS, or Distributed NAS, is the internal name for namespace software.
DNAS_POLICY	Inline policy events that occur during normal volume processing, such as file creates, directory renames, and other client actions. This is off by default because it generates a great deal of log messages, enough to potentially affect client traffic.

Log Component	Description
DNAS_SINIT	DNAS Storage Initialization.
DNAS_SREMOVE	DNAS Storage Removal.
DNC_LIB	DNAS Configuration Library for NFS. DNAS is the internal name for namespace software.
DOMAIN_JOIN	Active Directory domain joining components, which join an ARX-CIFS service to a Windows domain.
DR	Disaster Recovery, which involves loading a configuration on a backup-ARX site and activating the configuration on command.
DT	Direct Transport processes. This is a high-speed inter-process communication (IPC) library.
EMAILHOME	Email home daemon and CLI commands.
ENVLIB	Environmental Library, for monitoring the hardware platform. This only applies to the ARX-500.
EVAL_NET_RESULT	The results of joining an Active Directory (AD) domain.
EXIMD	Exim Daemon, which sends RPC requests to the exim message-transfer agent. That agent sends SMTP (email) messages from the ARX to external devices.
FCN	File Change Notification, for API access to snapshotted configuration and metadata.
FCND	File Change Notification daemon which provides API access to snapshotted configuration and metadata.
FFP	Fast Filter Processor.
FFP_UTIL	Fast Filter Processor Library.
FFPD	Fast Filter Processor Daemon.
FILER_MGMT_LIB	Snapshot daemon filer operations library.
FREESPACEMONITOR	Free Space Monitor.
FT_CIFS_SHIM_LIB	Interface to metadata storage on CIFS filers. Metadata is information about files and directories in managed volumes, such as their locations on back-end filers.
FT_CLI	File Tracking CLI. These are the CLI operations for setting up a file-history archive and using it to track the location of managed-volume files over time.
FTD	File Tracking daemon. This is the process of sending volume metadata and configuration to an external file-history archive, and using that archive to track the location of managed-volume files over time.
GBL_LIB	Global Services Library, for NFS and CIFS service management operations.
GBL_SVCMGR	Global Service Manager, which starts and stops NFS and CIFS front-end services.
GBLSVC_CLI	Global Service CLI.
GUI	Graphical User Interface (GUI), also known as the ARX Manager.
HA_CLI	High Availability CLI, such as the 'redundancy' command.

Log Component	Description
HWAGENT	Hardware Agent, for startup, shutdown, and monitoring of hardware components.
IPC_EVENT_LIB	IPC Event Library, for communication between internal ARX processes. This is deprecated, so it is unlikely to yield any log messages.
IPC_EVENT_MGR	IPC Event Manager, for communication between internal ARX processes.
IPC_LIB	Internal software inter-process messaging library.
IPCMOND	IPC service monitoring daemon, for monitoring the communication between internal ARX processes.
IPMIUTIL	ipmiutil event collector. All log levels for this logging component go to the kernel.log file, not syslog.
KRBCONFD	Active Directory (AD) Configuration Daemon, for periodically monitoring DCs and selecting the active DCs for each Windows domain.
KRBDDNS	Dynamic DNS registration helper. This registers dynamic-DNS aliases for an ARX-CIFS service with a remote DNS server.
L2SW_DRV	Layer2 - Internal Switch Hardware. This only applies to ARX-500, ARX-2000, and ARX-4000 systems.
L2SW_LVL7	Layer2 - Internal Switch Software. This controls packet traffic over network interfaces, channel traffic and LACP, spanning-tree, and other low-level networking. This only applies to ARX-500, ARX-2000, and ARX-4000 systems.
LIBCIFS	CIFS client library, used for non-client access to CIFS file servers. For example, this library is used for policy-invoked migrations.
LIBCIFSAUTH	CIFS authentication library, for authenticating front-end CIFS clients and internal ARX processes that access back-end filers.
LIBCIFSEXT	CIFS utility library, used in conjunction with LIBCIFS.
LIBEXIM	Exim Utilities messages.
LIBKDCLBSTATS	Kerberos load-balancing statistics.
LIBSDDL	CIFS Security Descriptor Definition Language translation library, which transforms binary Security IDs (SIDs) and descriptors into text strings. Callers to this library include the SID_CACHE, POLICY_SHADOW, and SUBSHRMGT processes.
LIBXSD	CIFS security-descriptor translation library, which processes can use to access the SID_CACHE daemon. The SID_CACHE daemon translates between Security IDs (SIDs) on various file servers and the group/user names in a Windows domain.
LICENSE	License monitoring daemon and CLI.
LIP_LIB	Logical IP Address Library. A logical IP address is an internal address that processes use to communicate to one another.
LOCBROKERD	IPC Location Broker for internal (process-to-process) RPC requests.
LOGMGRD	Logging Manager Daemon.
MANUF	Manufacturing Purposes only - Internal Switch Software.
MERGED	Layer2 - Merge FPGA Daemon.
MGMT_CLI	Management SLM - not supported.

Log Component	Description
MGT_DNAS_CONFIG	DNAS Configuration. DNAS, or Distributed NAS, is the internal name for namespace software.
MGT_RON_RTMD	Route Table Control Daemon. This checks gateway health and loads the best available routes into the OS route tables.
MGT_SVC_AGENT	Service Agent, which acts as a local agent for the central Service Manager (MGT_SVCMGR) process.
MGT_SVCMGR	Service Manager, used to start (and possibly restart) ARX processes.
MLBD	Metalog Block Device (MLBD) Controller, which manages the internal metalog subsystem. The metalog subsystem stores database-transaction logs for managed volumes.
MS_CLIENT	Microsoft RPC Services - High-level library used to interact with CIFS filers.
MS_RPC	Microsoft RPC Library - Low-level library used to interact with CIFS filers.
MTLDAEMON	Metalog Daemon. This is the process that records metalog (namespace-recovery log) data, on the local NVRAM hardware. This only runs on the ARX-500, ARX-2000, and ARX-4000.
NET	Network, or data-plane, processes.
NET_APP	Miscellaneous Application Events in the network software.
NET_CIFS	NSM CIFS Proxy.
NET_ERR_TRAP	Fastpath resource errors.
NET_HA	NSM HA.
NET_IPC_PROXY	NSM IPC Proxy, for communication between internal ARX processes on the control plane and on the data plane.
NET_LOG_PROXY	NSM Log Proxy.
NET_MSG	NSM Messaging between network processes.
NET_NFS	NSM NFS Proxy.
NET_OS	NSM Embedded Operating System.
NET_SLB	Server Load Balancing in the network software - not supported.
NET_STATS	NSM Statistics Collection.
NET_TCPSP	TCP Splicing.
NETBIOS	Netbios name service, which supports WINS.
NETSNMP	SNMP management on ARX-1500 and ARX-2500.
NFSCONFD	NFS Configuration Daemon, which manages one front-end NFS service.
NISD	NIS daemon messages. NIS servers provide a lookup service for NFS servers, to manage large NFS access lists. The NIS daemon queries all the configured NIS servers and caches the results of those lookups.
NLA	NETLOGON Agent, supports NTLM authentication of CIFS clients, but only for ARX CIFS services configured for constrained delegation.
NLM	NFS Network Lock Manager (NLM) Daemon. This implements the NFS NLM protocol, which manages file locks for NFS clients.
NMPIPE	Named Pipe, for communication between internal ARX processes (MS-RPC operations on CIFS IPC$ named pipes).

Log Component	Description
NSCK	DNAS Namespace Check (NSCK), for examining and repairing managed-volume metadata. Metadata is managed-volume information about its files and directories, such as locations on back-end filers.
NSCKRPT	NSCK Report Messages.
NSM	NSM Shared Memory.
NSM_RON	NSM RON.
NTLMAGENTAPI	NTLM/NTLMv2 authentication library using ARX Secure Agent.
NTP	Network Time Protocol (NTP) subsystem to keep the ARX clock(s) synchronized with an external clock on the IP network.
NVRDAEMON	Non Volatile RAM (NVRAM) Daemon, which manages one or more mtlDaemons (see MTLDAEMON). An mtlDaemon records metalog (namespace-recovery log) data, on the local NVRAM hardware. This only runs on the ARX-500, ARX-2000, and ARX-4000.
OBJ_MGR	Object Manager Database.
OBJ_MGR_LIB	Object Manager Library.
OM	Object Manager (Distributed Database) process. This is for the internal database, which stores the ARX configuration. You can see the full configuration with 'show running-config' and 'show global-config'.
OMIO	Object Manager Record I/O Tracing. This tracing is for the internal database, which stores the ARX configuration. (You can see the full ARX configuration with 'show running-config' and 'show global-config.').
OMTT	Object Manager Transaction Tracing. This tracing is for the internal database, which stores the ARX configuration. (You can see the full ARX configuration with 'show running-config' and 'show global-config.').
PERSONALITY	Personality, which parses XML data transferred between internal processes.
POLICY	not supported.
POLICY_ACTION	Migration requests and responses. You can use this to diagnose migration issues, such as file matching and mismatching.
POLICY_CLI	not supported.
POLICY_PDP	Policy Decision Point - policy scheduler, which uses schedule configuration to trigger scans and other policy events.
POLICY_PEP	Policy Enforcement Point - policy scanning, events, rule-state processing, and rule info.
POLICY_SHADOW	Policy Shadow Volume, a read-only volume that contains replicas of all the files and directories on another volume.
PROBE	Probes for Monitoring and Controlling Subsystems.
PTW	Parallel Treewalk Library.
RELMGR	Release Manager, which stores software-release (.rel) files, unpacks them, prepares them for use on the ARX, and otherwise manages them.
RELMGR_CLI	Release Manager CLI commands. This is deprecated, so it is unlikely to yield any log messages.
REMOTE_CLI	Remote SSH CLI command execution, invoked through the CLI (with the 'rconsole' command) or the GUI.

Log Component	Description
REPLMGR	Replication Manager - Directs Data-Mover Engine (DME) file-migrations.
REPORT	Common subsystem for generating and viewing reports.
RESTARTDETECT	This determines if internal processes can be restarted after a failure.
RON	Resilient Overlay Network (RON) CLI operations.
RON_SPFD	Resilient Overlay Network (RON) Routing Daemon. This manages RON tunnels and determines the best route to remote ARX devices.
RON_VPND	Not supported.
ROOTD	Root Daemon, which performs internal Unix management that requires root-level privileges.
SCM_DSMD	DNAS Switch Monitor.
SEC_CONFIG	Security Configuration.
SEC_LOCAL	Security Local Enforcement.
SECURITY_CLI	Security CLI.
SECURITYD	Security Daemon.
SHMEM	Common, shared, memory-management subsystem. This is primarily used for gathering statistics.
SHRTOP	Share topology library - Used for various filer-subshare computations.
SID_CACHE	Security descriptor translation daemon (XSDD) and its caches. XSDD translates between Security IDs (SIDs) on various file servers and the principal (group or user) names in a Windows domain.
SLMCONFD	SLM Configuration Daemon - not supported.
SMGMT	Storage Management Library.
SMTP	Used for file transfer (through email), email events, and other email-related features.
SNAPD	Snapshot daemon, which coordinates snapshots on multiple filers behind an ARX volume.
SQMD	Redundant Pair Site Quorum Manager, which makes failover decisions based on cluster integrity.
SSB_LIB	SSB Library. The SSB is an internal mechanism for processes to communicate with one another.
SSBEVENTD	SSB Event Daemon. The SSB is an internal mechanism for processes to communicate with one another.
SSH	SSH configuration.
SSRM	Simple Share Resource Monitor, which monitors filers on behalf of the HA subsystem.
STATSD	Statistics Daemon, which exposes all ARX statistics to SNMP, the CLI, and the ARX manager (GUI).
STATSMON	Statistics monitoring.
SUBSHRMGT	Subshare management daemon and its client library. This manages filer subshares. A filer subshare is any CIFS share that is inside another CIFS share.

Log Component	Description
SVCMGR_LIB	SvcMgr Client Library. This is deprecated, so it is unlikely to yield any log messages.
SYNC_CLOCK	Log messages related to sync_clock and sntpc on ASMs - not supported.
SYNC_LIB	Shadow Synchronization Library, which supports a shadow volume (see POLICY_SHADOW).
SYSLOG_NG_GUARD	syslog-ng_guard, which monitors and (if necessary) reanimates the syslog-ng process. The syslog-ng process maintains the syslog file, possibly duplicating it to a remote peer.
TASK_LIB	Task Dispatcher Library - Internal framework used by internal processes for managing event-driven tasks and scheduled tasks.
TCMD	NSM Configuration Manager.
TCORED	NSM Core Dump Manager.
TIME	Not supported.
TRANFS	Transitional File System in DNAS.
TRANFSD	Not supported.
TXNMOND	Object Manager Transaction Monitor. This monitors the internal database, which stores the ARX configuration. (You can see the full ARX configuration with 'show running-config' and 'show global-config.').
UTIL	Fast Filter IPC Utilities, for communication between internal ARX processes and the Chassis Manager (see CHASS).
VCIFS	dNAS CIFS Proxy, which handles requests to volume software on behalf of CIFS clients.
WDOG	Watchdog, which performs internal health checks on ARX hardware such as the SSB. The SSB is an internal mechanism for processes to communicate with one another.
WMA	Windows Management Authorization (WMA) CLI Commands and Library.
WMI	Software Library for managing snapshots on back-end Windows servers using Windows Remote Management protocol (WinRM) and Windows Management Instrumentation (WMI).

Syslog Syntax

Each line in the syslog uses the following syntax:

utc.uuu-tz:switch:slot-proc-board-pid:cmp-ins-sev-id:: msg

For example,

```
2009-05-13T11:05:44.202+0000:bstnA:1-1-SCM-29776:CLI-0-NOTE-CLI_COMMAND:: User admin: Command:
bstnA# show nsck
```

Where

- ***utc*** (`2009-05-13T11:05:44` in the example) is the time in UTC.

- ***uuu*** (`202`) is the millisecond time fraction.
- ***-tz*** (`+0000`) is the hours off UTC (+*nnnn* or -*nnnn*).
- ***switch*** (`bstnA`) is the switch name.
- ***slot-proc-board*** (`1-1-SCM`) is the chassis slot, processor number, and board type (ACM, ASM, SCM, or NSM). Use the show processors command for a list of all processors and their associated modules.
- ***pid*** (`29776`) is the Process ID (PID).
- ***cmp*** (`CLI`) is the log-component name. Refer to *Log Components*, above.
- ***ins*** (`0`) is the process's instance number.
- ***sev*** (`NOTE`) indicates the message severity: DBG (debug) is the least severe, followed by INFO, NOTE, WARN, ERR, and CRIT (critical).
- ***id*** (`CLI_COMMAND`) is the message-catalog ID, a unique ID for each log message. MSG*n* is an ID for an uncatologued message. The *n* is the severity of the message, from 7 (debug) down to 2 (critical). Refer to the *ARX Log Catalog* for an alphabetical list of log messages, sorted by their IDs. To view the documentation for a specific message, use the show documentation command (described later in the chapter).
- ***msg*** (`User admin: Command: bstnA#  show nsck`) is the message text itself.

Reading IDs for Namespaces, Volumes, and Shares

Syslog messages identify namespaces, volumes, and shares with numeric IDs. The syslog also uses numeric IDs to identify its file-history archives (recall *Creating a File-History Archive*, on page 3-6). To correlate these IDs with the names used in the CLI, use the show id-mappings command:

```
show id-mappings
```

For example:

```
bstnA(cfg)# show id-mappings

Namespace                          ID
-----------------------------  -----
medco                            1
wwmed                            2
medarcv                          3
insur                            4

Volume                    Namespace                    ID
------------------------  -------------------------  -----
/vol                      medco                        1
/acct                     wwmed                        2
/rcrds                    medarcv                      3
/lab_equipment            medarcv                      4
/test_results             medarcv                      5
/claims                   insur                        7

Share                Volume               Namespace               ID
-------------------  -------------------  ---------------------  -----
corporate            /vol                 medco                    1
sales                /vol                 medco                    2
```

```
generic              /vol                 medco                   3
budget               /acct                wwmed                   5
bills                /acct                wwmed                   6
bills2               /acct                wwmed                   7
it5                  /acct                wwmed                   8
metadata-share       /acct                wwmed                   4
rx                   /rcrds               medarcv                11
charts               /rcrds               medarcv                12
bulk                 /rcrds               medarcv                13
metadata-share       /rcrds               medarcv                10
equip                /lab_equipment       medarcv                15
leased               /lab_equipment       medarcv                16
backlots             /lab_equipment       medarcv                17
scanners             /lab_equipment       medarcv                18
metadata-share       /lab_equipment       medarcv                14
chemistry            /test_results        medarcv                19
hematology           /test_results        medarcv                20
shr1-old             /claims              insur                  25
shr1-next            /claims              insur                  26
metadata-share       /claims              insur                  24

Archive Name                           ID
-------------------------------------  -----
fileRecordsMed                           23

bstnA(cfg)# ...
```

Paging Through the Syslog (show logs)

Use the show logs command to page through the active syslog or one of the syslog backups:

show logs *file-name*

where *file-name* (1-255 characters) is the name of the desired syslog file.

You can then use <Space> to step through the file one page at a time. Press **q** to quit out of the file.

For example, this command shows the contents of the active syslog:

```
bstnA(cfg)# show logs syslog
2009-05-13T15:42:45.492+0000:bstnA:1-1-SCM-4445:BESPD-0-NOTE-SHARE_AVAILABLE:: Storage Transition:
[192.168.25.28] path [hematology_results] id [20] (Status now available)
2009-05-13T15:42:45.825+0000:bstnA:1-1-SCM-2141:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/test_results:hematology', name 'fs3', ip '192.168.25.28',
2009-05-13T15:42:54.234+0000:bstnA:1-1-SCM-3570:CRMD-0-WARN-CRMD_NO_MSHIP:: Cannot obtain a valid
HA membership [[ep:0,<>]]
2009-05-13T15:43:23.334+0000:bstnA:1-1-SCM-2141:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP:
policyRuleShareFrequencyZeroClear (528) nsp 'medarcv', vol '/rcrds', shr 'charts', nfp rule
'medFm', : Share frequency is non-zero
2009-05-13T15:43:39.960+0000:bstnA:3-2-ASM_FC-1873:TRANFS-3-WARN-MSG4:: get_space failed at
\\192.168.25.28\hematology_results - STATUS_TRANSACTION_TIMED_OUT
2009-05-13T15:43:52.492+0000:bstnA:1-1-SCM-4444:BESPD-0-NOTE-SHARE_UNAVAILABLE:: Storage
Transition: [192.168.25.28] path [hematology_results] id [20] (Status now unavailable); 4 probes
failed over 30 seconds (durations:   15s   15s 0.1ms 0.0ms); Last probe status:
[STATUS_BAD_NETWORK_PATH]
2009-05-13T15:43:52.826+0000:bstnA:1-1-SCM-2141:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOffline
(670) shr 'medarcv:/test_results:hematology', name 'fs3', ip '192.168.25.28',
```

```
2009-05-13T15:43:54.270+0000:bstnA:1-1-SCM-3570:CRMD-0-WARN-CRMD_NO_MSHIP:: Cannot obtain a valid
HA membership [[ep:0,<>]]
2009-05-13T15:44:14.143+0000:bstnA:1-1-SCM-2141:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP:
policyRuleSourceUnavailable (533) nsp 'medarcv', vol '/rcrds', nfp rule 'medFm', : Source
unavailable
2009-05-13T15:44:14.152+0000:bstnA:1-1-SCM-2141:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP:
policyRuleShareFrequencyZeroRaise (529) nsp 'medarcv', vol '/rcrds', shr 'charts', nfp rule
'medFm', : Share frequency is zero
2009-05-13T15:44:51.542+0000:bstnA:1-1-SCM-4445:BESPD-0-NOTE-SHARE_AVAILABLE:: Storage Transition:
[192.168.25.28] path [hematology_results] id [20] (Status now available)
2009-05-13T15:44:51.859+0000:bstnA:1-1-SCM-2141:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/test_results:hematology', name 'fs3', ip '192.168.25.28',
--MORE--
```

Searching Through the Syslog

Use the grep command to apply a regular expression to the syslog as you page through it, viewing only the lines that contain a desired string:

grep *pattern* logs [*file-name*]

where

pattern (1-255 characters) is the regular-expression pattern to search for in the syslog. The grep displays all lines containing this pattern.

The regular expression syntax follows grep's "basic" regular-expression syntax:

- **.** matches any single character.
- **.*** matches any string, including the null string.
- **[...]** matches any one of the enclosed characters.
- **[a-z]** matches any character in the sorted range, a through z.
- **** matches the next character, even if it has special meaning (for example, **\.** matches a period instead of any character).
- **[^...]** matches any character that is *not* enclosed. For example, [^0-9] matches any single character that is not a number from 0 to 9.

logs is a required keyword; this specifies the "logs" directory, where the syslogs are stored.

file-name (optional, 1-255 characters) identifies one file to search. If you omit the file name, the CLI searches through all files in the logs directory. Use the show logs command to see a list of accessible log files.

For example, the following command searches the active syslog for the string, "shareOnline:"

```
bstnA(cfg)# grep shareOnline logs syslog
2009-05-13T10:36:51.150+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medco:/vol:corporate', name 'nas1', ip '192.168.25.21',
2009-05-13T10:36:51.175+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medco:/vol:sales', name 'nas2', ip '192.168.25.44',
```

```
2009-05-13T10:36:51.199+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medco:/vol:generic', name 'nas3', ip '192.168.25.47',
2009-05-13T10:37:33.335+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'wwmed:/acct:budget', name 'das1', ip '192.168.25.19',
2009-05-13T10:37:33.414+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'wwmed:/acct:bills', name 'das8', ip '192.168.25.25',
2009-05-13T10:37:33.468+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'wwmed:/acct:bills2', name 'das3', ip '192.168.25.22',
2009-05-13T10:37:33.522+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'wwmed:/acct:it5', name 'das7', ip '192.168.25.24',
2009-05-13T10:39:04.019+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/rcrds:rx', name 'fs4', ip '192.168.25.29',
2009-05-13T10:39:04.122+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/rcrds:charts', name 'fs1', ip '192.168.25.20',
--MORE--
```

Filtering Messages Out

You can use the ignore flag to filter out a second string from the grep output:

grep *pattern* logs [*file-name*] ignore *ignore-pattern*

where

pattern, **logs**, and ***file-name*** are described above,

ignore is required, and

ignore-pattern (1-1024 characters) is a pattern to ignore. Surround this with quotes (“”) if it contains any spaces.

For example, the following command sequence repeats the above example but ignores all messages containing the “acct” string:

```
bstnA(gbl)# grep shareOnline logs syslog ignore acct
2009-05-13T10:36:51.150+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medco:/vol:corporate', name 'nas1', ip '192.168.25.21',
2009-05-13T10:36:51.175+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medco:/vol:sales', name 'nas2', ip '192.168.25.44',
2009-05-13T10:36:51.199+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medco:/vol:generic', name 'nas3', ip '192.168.25.47',
2009-05-13T10:39:04.019+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/rcrds:rx', name 'fs4', ip '192.168.25.29',
2009-05-13T10:39:04.122+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/rcrds:charts', name 'fs1', ip '192.168.25.20',
2009-05-13T10:39:04.222+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/rcrds:bulk', name 'fs2', ip '192.168.25.27',
2009-05-13T10:39:14.632+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/lab_equipment:equip', name 'nas10', ip '192.168.25.49',
2009-05-13T10:39:15.084+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/lab_equipment:leased', name 'nas10', ip '192.168.25.49',
2009-05-13T10:39:15.286+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/lab_equipment:backlots', name 'fs2', ip '192.168.25.27',
2009-05-13T10:39:15.457+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/lab_equipment:scanners', name 'fs5', ip '192.168.25.71',
2009-05-13T10:39:15.508+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/test_results:chemistry', name 'fs1', ip '192.168.25.20',
2009-05-13T10:39:15.560+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/test_results:hematology', name 'fs3', ip '192.168.25.28',
2009-05-13T10:39:24.245+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/test_results:2005_charts', name '[medarcv]',
2009-05-13T10:55:18.850+0000:bstnA:1-1-SCM-2138:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'insur:/claims:shr1-old', name 'nas1', ip '192.168.25.21',
```

```
...
bstnA(gbl)#
```

Searching the End of the Syslog

The show and grep commands search from the beginning of the syslog file, starting with the oldest messages and moving toward the most-recent logs. To skip ahead to the most-recent logs, use the tail option at the end of the grep command:

grep *pattern* logs [*file-name*] [ignore *ignore-pattern*] tail *number-lines*

where

pattern, **logs**, ***file-name***, and **ignore** ***ignore-pattern*** are described above,

tail is required, and

number-lines (1-4096) is the number of lines to show.

For example, the following grep command shows only the 10 most-recent matching lines in the active syslog:

```
bstnA(gbl)# grep shareOnline logs syslog tail 10
2009-05-13T13:15:30.519+0000:bstnA:1-1-SCM-2141:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/rcrds:rx', name 'fs4', ip '192.168.25.29',
2009-05-13T13:41:01.761+0000:bstnA:1-1-SCM-2141:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/test_results:hematology', name 'fs3', ip '192.168.25.28',
2009-05-13T13:51:41.818+0000:bstnA:1-1-SCM-2141:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/test_results:hematology', name 'fs3', ip '192.168.25.28',
2009-05-13T14:22:31.470+0000:bstnA:1-1-SCM-2141:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/test_results:hematology', name 'fs3', ip '192.168.25.28',
2009-05-13T14:39:45.403+0000:bstnA:1-1-SCM-2141:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/rcrds:rx', name 'fs4', ip '192.168.25.28',
2009-05-13T15:42:45.825+0000:bstnA:1-1-SCM-2141:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/test_results:hematology', name 'fs3', ip '192.168.25.28',
2009-05-13T15:44:51.859+0000:bstnA:1-1-SCM-2141:L2SW_LVL7-0-NOTE-MSG5:: SNMP TRAP: shareOnline
(680) shr 'medarcv:/test_results:hematology', name 'fs3', ip '192.168.25.28',
2009-05-13T16:10:39.029+0000:bstnA:1-1-SCM-29206:CLI-0-NOTE-CLI_COMMAND:: User admin: Command:
bstnA> grep shareOnline logs syslog
2009-05-13T16:14:10.839+0000:bstnA:1-1-SCM-29206:CLI-0-NOTE-CLI_COMMAND:: User admin: Command:
bstnA> grep shareOnline logs syslog ignore acct
2009-05-13T16:17:57.262+0000:bstnA:1-1-SCM-29206:CLI-0-NOTE-CLI_COMMAND:: User admin: Command:
bstnA> grep shareOnline logs syslog tail 10
bstnA(gbl)#
```

Showing the Documentation for a Syslog Message

Many syslog messages (as well as messages from the CLI) have more-verbose documentation associated with them; you can access this additional documentation from the CLI. From any mode, use the show documentation command to view the documentation for a given message:

show documentation *msg-catalog-id*

where ***msg-catalog-id*** (1-255 characters) identifies the message. This ID appears with the message in both the syslog (recall the ***id*** field in each syslog message; see *Syslog Syntax*, on page 8-16) and in CLI output.

For example, the following command sequence shows a syslog message, then shows the documentation for it:

```
bstnA(gbl)# grep RON_SPFD logs syslog
2009-05-13T11:34:10.620+0000:bstnA:1-1-SCM-3558:RON_SPFD-0-NOTE-RON_SYSTEM_CLOCK_CHANGED:: System
clock changed, adjusting schedule by 0 seconds, 131744 microseconds.
bstnA(gbl)# show documentation RON_SYSTEM_CLOCK_CHANGED

RON_SYSTEM_CLOCK_CHANGED documentation:

The RON Shortest Path First Daemon has adjusted its internal schedule in response to a system clock
change.
bstnA(gbl)# ...
```

Copying the Syslog File to an FTP Server

You can save the syslog file off to a remote FTP server for further analysis. From priv-exec mode, use the copy logs command and specify the FTP server in a destination URL:

copy logs *syslog-file* ftp://[*username*:*password*@]*ftp-site*/*file*

where

syslog-file identifies the syslog file to copy,

***username*:*password*@** (optional) is an FTP username and password (the default is set by the ip ftp-user command, as shown in *Setting a Default FTP or SCP User*, on page 5-3),

ftp-site identifies the FTP server with an IP address or FQDN (for example, “172.16.88.3” or “ftp.myftpsite.com”), and

file is the chosen file name. Lead with an extra “/” if the path starts at the root of the server machine; for example, “porthos//var/log/arx/syslog” specifies “/var/log/arx/syslog” on server porthos. Omit the leading slash if the file is going to the home directory for *username*.

For example, the following command exits from cfg mode to priv-exec mode, then copies the active-syslog file to a remote server:

```
bstnA(cfg)# exit
bstnA# copy logs syslog ftp://juser:jpasswd@ftp.wwmed.com/syslog_5-6
bstnA# ...
```

Copying the Syslog File through SCP

For secure sites, you can upload with the Secure Copy (SCP) protocol. The URL has a different syntax for SCP transfers:

copy logs *syslog-file* scp://*username*@*server*:*file* [accept-host-key]

where

syslog-file identifies the syslog file to copy,

***username*@** is a valid username at the remote host,

server identifies the SCP server with an IP address or FQDN (for example, “172.16.100.18” or “deb1.mynet.com”), and

file is the chosen file name at the remote server. Lead with a slash (*scp-server*:*/file*) if the file path is absolute. Without the slash, the path is presumed to start in the home directory for ***username***.

accept-host-key (optional) tells the CLI to accept an unknown host key if offered by the SCP server. The host key authenticates the server; if the key is unknown, it is possible that an attacker has taken the server’s hostname and/or IP address. Note that any SCP server is “unknown” if the switch has not had an SCP exchange with it since the switch’s last reboot.

The CLI prompts for the ***username***’s password. Enter a password that is valid at the remote site.

For example, the following command exits from cfg mode to priv-exec mode, then sends the “syslog” file to juser’s home directory on rh1.wwmed.com:

```
bstnA(cfg)# exit
bstnA# copy logs syslog scp://juser@rh1.wwmed.com:arx_syslog
Password: jpasswd
bstnA# ...
```

Copying the Syslog File into an ARX Volume

For sites that do not support FTP or SCP, you can copy the syslog file to an ARX volume:

copy logs ***syslog-file*** **{cifs|nfs}** ***namespace volume file-path***

where

syslog-file identifies the syslog file to copy,

cifs | nfs is a required choice. This is the network protocol used to copy the syslog file into the ARX volume.

namespace (1-30 characters) identifies the destination namespace.

volume (1-1024 characters) is the volume name.

file-path (1-255 characters) is the path from the volume root (above) to the destination file.

For example, the following command exits from cfg mode to priv-exec mode, then sends the “syslog” file to a directory in the “medarcv~/rcrds” volume:

```
bstnA(cfg)# exit
bstnA# copy logs syslog cifs medarcv /rcrds admin
bstnA# ...
```

Sending the Syslog File to an E-Mail Recipient

You can alternatively send the syslog file as an E-mail attachment. Before you use E-mail, you must configure the Simple Mail Transfer Protocol (SMTP) on the switch, starting with the smtp command in cfg mode: see the chapter on E-mail and SMTP in the *ARX® CLI Reference*.

Use the following syntax to send the syslog file in an E-mail message:

copy logs *syslog-file* smtp://[*e-mail-address*/]*file*

where

e-mail-address (optional) identifies the recipient of the E-mail message (for example, "myCoWorker@myco.com"). If you omit this, it defaults to the E-mail recipient set with the cfg-smtp to command. Use a slash (/) to separate this from the file name.

file is the chosen file name.

For example, the following command sequence sets up SMTP, exits from cfg mode to priv-exec mode, then mails the "syslog" file to "juser@wwmed.com:"

```
bstnA(cfg)# smtp
bstnA(cfg-smtp)# mail-server email1.wwmed.com
bstnA(cfg-smtp)# from admin@acopia.wwmed.com
bstnA(cfg-smtp)# exit
bstnA(cfg)# exit
bstnA# copy logs syslog smtp://juser@wwmed.com/arx_syslog
bstnA# ...
```

Adjusting Logging Levels

Each log component has a logging level that determines the volume of logs that it generates. You can set a different logging level for every log component, allowing you to focus attention on a single component or a small group of them.

From cfg mode, use the logging level command to change the logging level for one (or all) log components:

logging level *component* {critical | error | warning | notice | info | debug}

where

component can be **all** or any log component (see *Log Components*, on page 8-9), and

critical | error | warning | notice | info | debug chooses the logging level for the component. The **critical** level limits the component to critical log messages only, **error** allows both error and critical messages, and so on. **Debug** level is the most verbose; this causes the component(s) to issue every possible log message during their execution.

For example, the following command sets the logging level to debug (most-verbose) for the TRANFS component:

```
bstnA(cfg)# logging level TRANFS debug
bstnA(cfg)# ...
```

Disabling Log Messages

You can stop all log messages from a given component, or for all components. Use the disable keyword at the end of the logging level command:

logging level *component* disable

where

component can be **all** or any log component (see *Log Components*, on page 8-9), and

disable stops all logging from the ***component***.

For example, the following command disables logging from the POLICY_PDP component:

```
bstnA(cfg)# logging level POLICY_PDP disable
bstnA(cfg)# ...
```

Showing All Log Components and Levels

Use the show logging levels command to see the logging levels for all log components:

show logging levels

The output is a table where the X next to each log component marks its logging level. If the APP_RON_RTMD component has an X in the "warning" column, for example, the APP_RON_RTMD is set to the "warning" logging level.

All of the log components shown in the example below are set to "notice" logging level:

```
bstnA(cfg)# show logging levels


Configured Log Levels:
                    disable  debug   info   notice  warning  error critical
AFN_NET            |        |       |       |    X  |       |       |      |
APP_RON_RTMD       |        |       |       |    X  |       |       |      |
AUTH_CONSOLE       |        |       |       |    X  |       |       |      |
AUTH_HTTP          |        |       |       |    X  |       |       |      |
AUTH_HTTPS         |        |       |       |    X  |       |       |      |
AUTH_SSH           |        |       |       |    X  |       |       |      |
AUTH_TELNET        |        |       |       |    X  |       |       |      |
BOOTPD             |        |       |       |    X  |       |       |      |
CHASS              |        |       |       |    X  |       |       |      |
CHASS_LIB          |        |       |       |    X  |       |       |      |
CIFSCLI            |        |       |       |    X  |       |       |      |
CIFSCONFD          |        |       |       |    X  |       |       |      |
CIFS_INIT          |        |       |       |    X  |       |       |      |
CIFS_SHIM_LIB      |        |       |       |    X  |       |       |      |
CLI                |        |       |       |    X  |       |       |      |
CLI_WIZARD         |        |       |       |    X  |       |       |      |
CLUSTER            |        |       |       |    X  |       |       |      |
COMMONLIB          |        |       |       |    X  |       |       |      |
```

...

Focusing on One Logging Level

You can specify a particular log component to show its current logging level:

show logging levels *component*

where ***component*** can be **all** or any log component (see *Log Components*, on page 8-9).

The output is one row from the table shown above. For example, the following command shows the logging level for the "POLICY" component:

```
bstnA(cfg)# show logging levels POLICY


Configured Log Levels:
                    disable  debug   info   notice  warning  error critical
POLICY             |       |       |       |    X  |       |       |      |

bstnA(cfg)# ...
```

Reverting Back to the Default Logging Level

The default logging level for all components is "info." Use no logging level to revert to the default logging level:

no logging level {*component* | all}

where you must choose either a single ***component*** or **all** components. Refer to *Log Components*, on page 8-9 for a complete list of components.

For example, the following command sequence reverts all components to the default logging level, info. To prevent the syslog from filling up too quickly, you should do this after every examination of the syslog:

```
bstnA(cfg)# no logging level all
bstnA(cfg)# ...
```

Sending Logs to an External Server

You can configure the ARX to send its syslog messages to one or more external servers. The external server receives log messages as they are generated and appends them to its local syslog file.

From cfg mode, use the logging destination command to add a server to the list of logging destinations:

```
logging destination target [udp | tcp] [port port]
```

where

target (1-80 characters) identifies an external server. This can be an IP address or, if you have a DNS lookups configured, a hostname. (To configure the switch for hostname lookups, see *Configuring DNS Lookups*, on page 4-31 of the *ARX® CLI Network-Management Guide*.)

udp | tcp (optional) chooses a transport protocol to reach the external server. This defaults to UDP if you omit it.

port *port* (optional, 1-65,535) selects a particular UDP or TCP port to receive the log messages. The default is port 514.

You can use this command multiple times to configure multiple destination servers. For example, the following command sequence configures two servers to receive syslog messages:

```
bstnA(cfg)# logging destination 172.16.202.8
bstnA(cfg)# logging destination 10.1.1.90 tcp port 9999
bstnA(cfg)# ...
```

Removing a Logging Destination

Use the no form of the logging destination command to remove one log server from the list:

```
no logging destination target
```

where *target* (1-80 characters) identifies the server to remove. This can be an IP address or a hostname.

For example:

```
bstnA(cfg)# no logging destination 10.1.1.90
bstnA(cfg)# ...
```

Sending Log Messages from a VLAN Interface (optional)

By default, the log messages come from the out-of-band management interface (labeled MGMT on the switch's front panel). You have the option to change the source address to an in-band (VLAN) management interface, instead: this uses the client/server ports. (For more information on in-band management interfaces, see *Configuring an In-Band (VLAN) Management Interface*, on page 4-4 of the *ARX® CLI Network-Management Guide*.)

From cfg mode, use the management source command to change the source interface for outbound syslog messages:

```
management source vlan vlan-id
```

where *vlan-id* (1-4096 characters) identifies the VLAN. Use show interface vlan to list all configured VLANs and their management interfaces (see *Listing all VLAN Management Interfaces*, on page 4-6 of the *ARX® CLI Network-Management Guide*).

For example, the following command sequence uses the in-band management interface for VLAN 25:

```
bstnA(cfg)# management source vlan 25
bstnA(cfg)# ...
```

Using the Out-of-Band Management Interface

To send syslog messages from the out-of-band management interface, the default, use the no form of the management source command:

```
no management source
```

For example:

```
prtlndA(cfg)# no management source
prtlndA(cfg)# ...
```

The management source mgmt command has the same effect:

```
management source mgmt
```

For example:

```
bstnA(cfg)# management source mgmt
bstnA(cfg)# ...
```

Showing all Logging Destinations

To show all configured logging destinations, use show logging destination from any mode:

```
show logging destination
```

For example:

```
bstnA(cfg)# show logging destination

Source Interface:   mgmt (10.1.1.7)


External Syslog Destination(s)
------------------------------
172.16.202.8

bstnA(cfg)# ...
```

Listing Current System Tasks

The show system tasks command shows all running processes and the CPU and memory that each process is consuming. This is similar to the **ps** command on a Unix system, or the Task Manager on a Windows system. Invoke this command from any mode:

```
show system tasks
```

The processes are listed in the “Subsystem” column. For example:

```
bstnA(cfg)# show system tasks

Subsystem                    Instance  Memory (Kb)   CPU%
---------------------------  --------  ------------  ----
afnEmailHomed                1         6980          0
afnEximMgrd                  1         6060          0
afnFTd                       1         10252         0
afnFcnd                      1         25644         0
afnFfpMgrd                   1         5012          0
afnSlmConfd                  1         13296         0
afncrmd                      1         5412          0
afnipceventmgrd              1         5204          0
afnlicd                      1         13260         0
afnloggermgrd                1         6392          0
afnmerged                    1         4996          0
afnnetd                      1         6596          0
afnpdpd                      1         6636          0
afnrootd                     1         9840          0
afnsnapd                     1         13872         0
afnsqmd                      1         10280         0
afnssbdevd                   1         5772          0
afnssrmd                     1         5964          0
afnstatsd                    1         6916          0
apiLogFilter                 1         2104          0
apiLogFilter                 1         2112          0
apiLogger                    1         2268          0
apiLogger                    1         2112          0
apiLogger                    1         2104          0
apiLogger                    1         2108          0
apiLogger                    1         2108          0
bespd                        1         30888         0
bootpd                       1         4932          0
chassnew                     1         6536          0
cifsconfd                    1         7804          0
cli_daemon                   1         38000         0
coreCollector (script)       1         1728          0
coreTimer (script)           1         1680          0
...
tcmd                         1         6408          0
tcored                       1         4980          0
temipcproxyd                 1         5080          0
temstatsd                    1         6616          0
txnmond                      1         7876          0
vg_launch (script)           1         1284          0
vg_launch (script)           1         1284          0
vg_launch (script)           1         1544          0
xsdd                         1         7808          0
Other Tasks                  36        173196        0
bstnA(cfg)# ...
```

Collecting Diagnostic Information

Some problems require thorough examination and diagnosis. To facilitate this, the CLI has a command for collecting all interesting diagnostic information on the ARX and sending it to a remote site. The diagnostic information is a collection of very large files (including multiple core files), so chose a destination server with plenty of free disk space.

From priv-exec mode, use the collect diag-info ftp command to collect all diagnostic data (except reports, which can make the file excessively large) and send it to an FTP site:

collect diag-info ftp://[*username*[:*password*]@]*ftp-site*/*file*.tgz

where

***username*[:*password*]@** (optional) is an FTP username and optional password (the default is the username/password set by the ip ftp-user command),

ftp-site identifies the FTP server with an IP address or FQDN (for example, “172.16.88.3” or “ftp.myftpsite.com”), and

***file*.tgz** is the chosen file name. Lead with an extra “/” if the path starts at the root of the server machine; for example, “athos//usr/local/diag_files.tgz” specifies “/usr/local/diag_files.tgz” on server athos. Omit the leading slash if the file is going to the home directory for *username*.

The CLI prompts for a password if you enter the *username* without one. The CLI then prompts for confirmation before collecting the diagnostics. Answer **yes** to continue. The collection process can be time-consuming; prompts appear to show the progress of the operation as it occurs.

For example, the following command exits from cfg mode to priv-exec mode, then sends diagnostic data to arxftp.f5.com:

```
bstnA(cfg)# exit
bstnA# collect diag-info ftp://juser@arxftp.f5.com/acopia-diags.tgz
Password: jpasswd

Collect diagnostic information? [yes/no] yes
% INFO: Completed CLI show information collection at 02/12/08 01:22:03.
% INFO: Completed log file collection at 02/12/08 01:22:03.
% INFO: Completed core file collection at 02/12/08 01:22:04.
% INFO: Completed state information at 02/12/08 01:23:33.
% INFO: Completed configuration information at 02/12/08 01:23:33.
% INFO: Preparing diagnostic file at 02/12/08 01:23:33.
% INFO: Collection complete at 02/12/08 01:24:02.
% INFO: The copy completed successfully.
bstnA# ...
```

Sending the Information Securely, through SCP

For secure sites, you can upload the diagnostic data with the Secure Copy (SCP) protocol. The URL has a different syntax for SCP transfers:

collect diag-info scp://*username*@*server*:*file*.tgz [accept-host-key]

where

***username*@** is a valid username at the remote host,

server identifies the SCP server with an IP address or FQDN (for example, “172.16.100.18” or “deb1.mynet.com”), and

file*.tgz** is the file name for the diagnostics data. Lead with a slash (*scp-server*:*/file*) if the file path is absolute. Without the slash, the path is presumed to start in the home directory for ***username.

accept-host-key (optional) tells the CLI to accept an unknown host key if offered by the SCP server. The host key authenticates the server; if the key is unknown, it is possible that an attacker has taken the server’s hostname and/or IP address. Note that any SCP server is “unknown” if the switch has not had an SCP exchange with it since the switch’s last reboot.

The CLI prompts for the ***username***’s password. Enter a password that is valid at the remote site. Then the CLI prompts for confirmation before collecting diagnostics; enter **yes** to continue. As mentioned above, you should allow at least one minute for the collection process to complete.

For example, the following command exits from gbl mode to priv-exec mode, then uploads diagnostic data through SCP:

```
bstnA(cfg)# exit
bstnA# collect diag-info scp://juser@rh1.wwmed.com:acopia-diags.tgz accept-host-key
Password: jpasswd

Collect diagnostic information? [yes/no] yes
% INFO: Completed CLI show information collection at 02/13/08 19:22:32.
% INFO: Completed log file collection at 02/13/08 19:22:32.
% INFO: Completed core file collection at 02/13/08 19:22:32.
% INFO: Completed state information at 02/13/08 19:23:24.
% INFO: Completed configuration information at 02/13/08 19:23:24.
% INFO: Preparing diagnostic file at 02/13/08 19:23:24.
% INFO: Collection complete at 02/13/08 19:23:56.
% INFO: The copy completed successfully.
bstnA# ...
```

Placing the Diagnostics File into an ARX Volume

Some sites do not allow any FTP access to their data centers, and do not support SCP. For those sites, you can send the diagnostic data to a client-accessible volume on the ARX. You can use the nfs or cifs clause to send the output file to a given directory in a given volume:

collect diag-info {cifs|nfs} ***namespace volume dest-path***

where

cifs | nfs is a required choice. This is the network protocol used to transfer the diagnostics file to the ARX volume.

namespace (1-30 characters) identifies the destination namespace.

volume (1-1024 characters) is the destination-volume name.

dest-path (1-255 characters) is the intended path from the volume root (above) to the diagnostics file. The directory you specify here must exist on the volume.

For example, the following command exits from gbl mode to priv-exec mode, then uploads diagnostic data to the "medarcv~/rcrds" volume:

```
bstnA(cfg)# exit
bstnA# collect diag-info cifs medarcv /rcrds admin/collect.tgz

Collect diagnostic information? [yes/no] yes
% INFO: Completed CLI show information collection at 04/30/2009 11:32:24 AM.
% INFO: Completed log file collection at 04/30/2009 11:32:25 AM.
% INFO: Completed core file collection at 04/30/2009 11:32:25 AM.
% INFO: Completed state information at 04/30/2009 11:33:54 AM.
% INFO: Completed configuration information at 04/30/2009 11:33:55 AM.
% INFO: Preparing diagnostic file at 04/30/2009 11:33:55 AM.
% INFO: Collection complete at 04/30/2009 11:34:52 AM.
% INFO: Copying 39 megabytes from the specified source . . .
% INFO: The copy source file 'collectDiagInfo.22837.tgz' to destination file 'collect.tgz'
completed successfully.
bstnA# ...
```

Sending the Information in an E-Mail Message

You can alternatively send the diagnostics information as an E-mail attachment. Before you use E-mail, you must configure the Simple Mail Transfer Protocol (SMTP) on the switch, starting with the smtp command in cfg mode: see the chapter on E-mail and SMTP in the *ARX® CLI Reference*.

In the same collect diag-info command, use the following syntax to send the diag-info as an E-mail attachment:

collect diag-info smtp://[*e-mail-address*/]*file*.tgz

where

e-mail-address (optional) identifies the recipient of the E-mail message (for example, "myCoWorker@myco.com"). If you omit this, it defaults to the E-mail recipient set with the cfg-smtp to command. Use a slash (/) to separate this from the file name.

***file*.tgz** is the chosen file name. The ".tgz" extension is required.

As with the FTP and SCP uploads, the CLI prompts for confirmation before collecting the diagnostics. Answer **yes** to continue. The collection process can be time-consuming; prompts appear to show the progress of the operation as it occurs.

For example, the following command sequence sets up SMTP, exits from cfg mode to priv-exec mode, then sends diagnostic data via E-mail to "juser@wwmed.com:"

```
bstnA(cfg)# smtp
bstnA(cfg-smtp)# mail-server email1.wwmed.com
bstnA(cfg-smtp)# from admin@acopia.wwmed.com
bstnA(cfg-smtp)# exit
bstnA(cfg)# exit
bstnA# collect diag-info smtp://juser@wwmed.com/acopia-diags.tgz
```

```
Collect diagnostic information? [yes/no] yes
% INFO: Completed CLI show information collection at 02/13/08 19:34:05.
% INFO: Completed log file collection at 02/13/08 19:34:05.
% INFO: Completed core file collection at 02/13/08 19:34:05.
% INFO: Completed state information at 02/13/08 19:35:13.
% INFO: Completed configuration information at 02/13/08 19:35:13.
% INFO: Preparing diagnostic file at 02/13/08 19:35:13.
% INFO: Collection complete at 02/13/08 19:35:45.
% INFO: Transferred 0 megabytes; still copying . . .
% INFO: The copy completed successfully.
bstnA# ...
```

Collecting the Information Locally

You have the option to store the diagnostic-information file on the switch's internal disks. To conserve disk space, you can only store one diag-info file at a time.

collect diag-info *file*.tgz

where ***file*** is the chosen file name. You must use a .tgz extension on the end of the file name. The file is stored in the diag-info directory.

The CLI prompts for confirmation before overwriting any previously-collected diag-info files. Answer **yes** to continue. As stated above, the collection process can be time-consuming; allow at least one minute for the collection process to finish.

For example, the following command exits from cfg mode to priv-exec mode, then saves diagnostic data to a local file:

```
bstnA(cfg)# exit
bstnA# collect diag-info juser-ns-wwmed.tgz

    Only one copy of diagnostic information may be stored on the switch at one time.
    This automatically deletes any saved diagnostic information.

Are you sure? [yes/no] yes
% INFO: Completed CLI show information collection at 02/13/08 19:39:03.
% INFO: Completed log file collection at 02/13/08 19:39:03.
% INFO: Completed core file collection at 02/13/08 19:39:03.
% INFO: Completed state information at 02/13/08 19:40:10.
% INFO: Completed configuration information at 02/13/08 19:40:10.
% INFO: Preparing diagnostic file at 02/13/08 19:40:10.
% INFO: Collection complete at 02/13/08 19:40:38.
% INFO: The copy completed successfully.
bstnA# ...
```

Showing the Diag-Info File

Use the show diag-info command to view the directory listing with the new diag-info file:

show diag-info

For example:

```
bstnA# show diag-info
```

```
  diag-info
      juser-ns-wwmed.tgz      May 18 08:17  6.3M

bstnA# ...
```

Running the Collection Asynchronously

You can run the collection process in the background while you continue to enter more CLI commands. To accomplish this, use the optional async keyword before the URL:

collect diag-info *destination* async

where

destination can be any of the formats defined above for FTP, SCP, SMTP, an ARX volume, or a local file, and

async makes the collection process asynchronous.

The CLI prompts for confirmation; answer **yes** to continue. With the async option, the CLI displays a report name and then returns. You can use the tail reports *report-name* follow command to follow the collection process, or you can enter other CLI commands.

For example, the following command exits from cfg mode to priv-exec mode, then asynchronously collects and uploads diagnostic data:

```
bstnA(cfg)# exit
bstnA# collect diag-info ftp://juser:jpasswd@arxftp.f5.com/acopia-diags.tgz async

Collect diagnostic information? [yes/no] yes
Scheduling report: collect_diag_200711292228.rpt

bstnA# ...
```

Collecting Partial Information

You can choose from several optional keywords to expand or narrow the focus of the collect command:

collect {diag-info | config | cores | logs | reports | state | capture | all} *destination* [async]

where:

diag-info collects everything except reports, as described above.

config collects the configuration database, database logs, and the output from show running/global config.

cores gathers any core-dump files on the system, along with all shared-object libraries used by the system.

logs assembles the syslog, the procdat log (process data; a more-detailed log file for diagnosis by engineers), upgrade and installation logs, disk-usage logs, Kerberos state files, and logs from the system's service manager.

reports chooses all reports. (As mentioned above, these are omitted from the **diag-info** option because they can be excessively large at some installations.)

state collects individual log files from every processor in the switch, three system log files (syslog, procdat, and traplog), a chronological list of all CLI commands used since the switch booted, and the output from a series of useful show commands.

capture assembles all packet-capture files, which you can list with the show capture command. You can use a CLI command to capture IP traffic in a file for diagnostic purposes, as described in a later chapter. This command can send the capture file(s) to F5 for detailed examination.

all collects all information, including all reports. The **diag-info** option omits reports because they can be excessively large on some switches.

destination is either a local file name or any of the upload formats described above.

async (optional) makes the collection process asynchronous.

As shown earlier, the CLI prompts for confirmation before collecting the diagnostics. Answer **yes** to continue. The collection process can be time-consuming, depending on system state and the option that you have chosen; prompts appear to show the progress of the operation as it occurs.

For example, the following command exits from gbl mode to priv-exec mode, then sends state information to arxftp.f5.com:

```
bstnA(cfg)# exit
bstnA# collect state ftp://arxftp.f5.com/sys-state.tgz

Collect diagnostic information? [yes/no] yes
% INFO: Completed CLI show information collection at 02/13/08 19:52:57.
% INFO: Completed log file collection at 02/13/08 19:52:57.
% INFO: Completed state information at 02/13/08 19:54:12.
% INFO: Completed configuration information at 02/13/08 19:54:12.
% INFO: Preparing diagnostic file at 02/13/08 19:54:12.
% INFO: Collection complete at 02/13/08 19:54:21.
% INFO: The copy completed successfully.
bstnA# ...
```

Collecting Logs within a Time Frame

You can specify a time frame for log-file collection, so that the CLI collects only the logs from a certain time. You can use this option to focus on an eventful time period:

collect logs [start-time *start* [utc]] [end-time *end* [utc]] *destination* [async]

where you can choose a start and/or end time for the collected logs:

start-time ***start*** **[utc]** (optional) establishes a start time. If omitted, the command collects all of its logs up to the **end-time**.

- ***start*** is a date and time. Enter this in one of two formats: ***mm/dd/yyyy:HH:MM:SS*** (designed for manual entry) or ***yyyy-mm-dd*****T*****HH:MM:SS*** (the time format found in the log files).
- **utc** (optional) indicates that the ***start*** is UTC instead of local time. Without this, the start is assumed to be in local time, as established by the clock timezone command. This is intended for a copy-and-paste from a log file. (All time stamps in the log files are in UTC.)

end-time ***end*** **[utc]** (optional) establishes an end time for the collected logs. If omitted, the command collects all of its logs after the **start-time**. This has the same options as described above.

destination is either a local file name or any of the upload formats described above.

async (optional) makes the collection process asynchronous.

As with a full collection, the CLI prompts for confirmation before collecting the logs. Answer **yes** to continue. The collection process can be time-consuming, depending on system state and the options that you have chosen; allow at least one minute for the process to finish.

For example, the following commands collect logs from a specific three-hour interval (9:30AM to 12:30PM on May 6, 2007) and send them through E-mail:

```
bstnA(cfg)# exit
bstnA# collect logs start-time 05/06/2008:09:30:00 end-time 05/06/2008:12:30:00
smtp://juser@wwmed.com/mornLogs.tgz

Collect diagnostic information? [yes/no] yes
% INFO: Completed CLI show information collection at 05/06/08 20:09:50.
% INFO: Completed log file collection at 05/06/08 20:09:52.
% INFO: Completed configuration information at 05/06/08 20:09:52.
% INFO: Preparing diagnostic file at 05/06/08 20:09:52.
% INFO: Collection complete at 05/06/08 20:10:10.
% INFO: The copy completed successfully.
bstnA# ...
```

Setting Up Remote Monitoring By F5 Support

You can regularly send diagnostic information to F5 Support so that they can monitor the health of your system, watching for any degradation in resources and possibly stopping issues before they affect client service. F5 Support may also use this information to identify and eliminate performance bottlenecks that may not be obvious. This information is encrypted and then sent through SMTP as an E-mail attachment. The data-gathering, encryption, and e-mailing process is done on a schedule. You set the schedule (possibly along with some other options) as described in this section.

Important

*You must configure DNS and SMTP before you set up this data gathering mechanism. For instructions on setting up DNS, refer to the ip name-server command in the **ARX® CLI Reference**. For instructions on setting up SMTP, refer to the smtp command in the same manual.*

From gbl mode, use the auto-diagnostics command to begin configuring the schedule for this regular data collection:

```
auto-diagnostics
```

This places you in gbl-auto-diag mode, where you must set a schedule for the data to be gathered, encrypted, and mailed. You also have some additional options, such as adding more E-mail recipients for the gathered data.

For example, the following command begins setting up auto diagnostics for the "bstnA" switch:

```
bstnA(gbl)# auto-diagnostics
bstnA(gbl-auto-diag)# ...
```

Auto diagnostics can be disabled using the no auto-diagnostics command, which removes all in-memory and persistent database information about the existing auto-diagnostics configuration.

Setting the Collection Schedule

The first step in setting up auto-diagnostic collection is assigning a schedule for it. To create a schedule, use the gbl schedule command (refer to *Appendix 12, Creating a Policy Schedule* of the *ARX® CLI Storage-Management Guide* for details). To apply a schedule to auto-diag collection, use the schedule command in gbl-auto-diag mode:

```
schedule name
```

where ***name*** (1-64 characters) identifies the schedule. Use the show schedule command to list all schedules.

A schedule is required for auto-diagnostics to run. You can change to a new schedule later by re-running the command.

For example, the following command sequence applies a schedule, "daily4am," to auto-diag collection:

```
bstnA(gbl)# auto-diagnostics
bstnA(gbl-auto-diag)# schedule daily4am
bstnA(gbl-auto-diag)# ...
```

Adding Additional Show Commands to the Collection

By default, the auto-diagnostics collection operation collects the following commands from the current ARX:

- *show clock*
- *show chassis*
- *show processors*
- *show processors usage*
- *show health*
- *show version*
- *show running-config*
- *show global-config*
- *show ron*
- *show namespace* all
- *show arp* all
- *show redundancy all*
- *show cores*
- *show statistics namespace ... fastpath* all fastpath
- *show policy* details
- *show cifs-service user-sessions* all summary
- *show statistics cifs authentication* all
- *show active-directory status* detailed
- *show statistics filer* detailed
- *show external-filer* all

You can use the additional-command operation to add more show commands to this collection:

additional-command "*show-command-string*"

where ***show-command-string*** is the additional show command (128-byte limit) to add to the auto-diag collection. This command must start with the keyword, "show," and must be accessible from any CLI mode. Surround the string with quotation marks ("") if it contains any spaces.

For example, the following command sequence adds two show commands to auto-diag collection:

```
bstnA(gbl)# auto-diagnostics
bstnA(gbl-auto-diag)# additional-command "show global service"
bstnA(gbl-auto-diag)# additional-command "show share status"
```

```
bstnA(gbl-auto-diag)# ...
```

On the backup switch, the default set of show commands is smaller, as it omits global CLI commands. The default list of show commands supported on the backup switch is:

- *show clock*
- *show version*
- *show chassis*
- *show processors*
- *show processors usage*
- *show health*
- *show running-config*
- *show ron*
- *show arp* all
- *show redundancy all*
- *show cores*

Removing Additional Show Commands

Use the command no additional-command to remove a user-added show command from future auto-diagnostic collections. This removes only the specified additional command, not any of the default commands listed in the previous section. The command syntax follows:

```
no additional-command "show-command-string"
```

For example, this command sequence removes the extra show command show share status from the auto-diagnostics collection at the "bstnA" switch:

```
bstnA(gbl)# auto-diagnostics
bstnA(gbl-auto-diag)# no additional-command "show share status"
bstnA(gbl-auto-diag)# ...
```

Adding Additional E-mail Recipients

You can use the mail-to command to add additional recipients for the collected E-mails. Unlike the collection for F5 Support, the E-mail attachment for these recipients is *not* encrypted. The ARX encrypts the collected information for F5 Support because it is likely to traverse the Internet:

```
mail-to recipient
```

where ***recipient*** is the E-mail address (128-byte limit) for someone to receive the collected diagnostics.

For example, the following command sequence adds one E-mail recipient (in addition to F5 Support) for auto-diag collections:

```
bstnA(gbl)# auto-diagnostics
bstnA(gbl-auto-diag)# mail-to juser@wwmed.com
bstnA(gbl-auto-diag)# mail-to jsmith@wwmed.com
bstnA(gbl-auto-diag)# ...
```

Removing an Additional E-mail Recipient

Use no mail-to to remove one E-mail recipient from the auto-diagnostics list:

no mail-to *recipient*

where ***recipient*** (1-768 characters) is the E-mail address to remove.

You cannot remove "e-support@f5.com," the address for F5 Support.

For example, the following command sequence removes one of the E-mail recipients that was added in the previous example:

```
bstnA(gbl)# auto-diagnostics
bstnA(gbl-auto-diag)# no mail-to jsmith@wwmed.com
bstnA(gbl-auto-diag)# ...
```

Displaying the Auto-Diagnostics Configuration

You can display the current auto-diagnostics configuration by executing the show auto-diagnostics command. An example of the output from this command follows:

```
bstnA# show auto-diagnostics

  Schedule:         daily4am
    Description:      two hours between 4 and 6 AM
    Start Time:       Sun Sep  4 04:00:00 2005
    Stop Time:        Wed Jan  7 04:00:00 2015 (Expires in 1665 d 22:58:06)
    Interval:         1 days
    Duration:         02:00:00
    Status:           Paused (runs in 21:58:06)

    Previous:
    Run Time:         Wed Jun 16 04:00:00 2010
    End Time:         Wed Jun 16 06:00:00 2010

    Next:
    Run Time:         Thu Jun 17 04:00:00 2010
    End Time:         Thu Jun 17 06:00:00 2010


  Mail-to:   juser@wwmed.com, mmorrison@wwmed.com

  Additional commands:
    show global service
    show share status

  Status:
    Last time executed: 16 Jun 2010 09:19:53 AM UTC
```

```
Status of last execution: Success [ collect_diag_201006160919.rpt DONE ]
Date last sent: 16 Jun 2010 09:55:27 AM UTC
Number of times sent since last reboot: 2
```

Performing a Local Test of the Auto-Diagnostics Configuration

After you've configured auto-diagnostics behavior for your ARX, you can execute a local test to verify that it works the way that you expect it will.

Type the privileged-exec mode CLI command auto-diagnostics test local to send an unencrypted auto-diagnostics mail test message to the Email addresses configured with the gbl-auto-diag mode mail-to command. No mail is sent to F5 Networks when this command is executed.

Running Show Commands from a Remote Host

You can use SSH (the Secure SHell command) from a remote machine to run any show command at an ARX. Earlier chapter describe how to run various snapshot commands and/or a volume restore through SSH; recall *Running Snapshot Operations from a Remote Host*, on page 2-34 and *Running the Restore from a Remote Host*, on page 4-10. You use the same **ssh** syntax to run show commands:

```
ssh admin-user@mip "show-command"
```

where

admin-user is the username for a valid administrative account at the ARX (use show users to list all of them, as shown in *Listing All Administrative Users*, on page 2-14 of the *ARX® CLI Network-Management Guide*),

mip is a management-IP address for the ARX (use show interface mgmt to show the out-of-band management interface, or show interface vlan to show all in-band management interfaces), and

show-command is a show command to run remotely. Surround this with quotation marks (“”) if it contains any spaces.

The output of the show command appears in the local shell. For example, this command sequence runs from a Unix machine named “mgmt17.” The command runs show processors on “bstnA,” a remote ARX:

```
juser@mgmt17:~$ ssh admin@10.1.1.7 "show processors"
Command>show processors

Proc     Module  State       Up Time             Total Kb  Free Kb  CPU1M   CPU5M
----     ------  ----------  ------------------  --------  -------  ------  -----
1.1      ACM     Up          0 days, 01:50:00    16415268  15558320 3       2
2.1      NSM     Up          0 days, 01:45:20    2714622   2105952  0       0
2.2      NSM     Up          0 days, 01:45:20    2714622   2105952  0       0
2.3      NSM     Up          0 days, 01:45:20    2714622   2105952  0       0
2.4      NSM     Up          0 days, 01:45:20    2714622   2105952  1       1
2.5      NSM     Up          0 days, 01:45:15    2714622   2148961  0       0
2.6      NSM     Up          0 days, 01:45:15    2714622   2148961  0       0
2.7      NSM     Up          0 days, 01:45:15    2714622   2148961  0       0
2.8      NSM     Up          0 days, 01:45:15    2714622   2148961  1       1
2.9      NSM     Up          0 days, 01:45:15    2714622   2105953  0       0
2.10     NSM     Up          0 days, 01:45:15    2714622   2105953  0       0
2.11     NSM     Up          0 days, 01:45:15    2714622   2105953  0       0
2.12     NSM     Up          0 days, 01:45:15    2714622   2105953  1       1
juser@mgmt17:~$
```

Using Quotes within the Show Command

Some show commands require quotes within the command itself. This confuses the CLI parser, and single quotes (‘) are also problematic. To work around this issue, use the URL encoding for the double quote, %22, for the inner quotes.

For example, the following command shows a namespace named “test ns.” The show namespace command requires quotes around “test ns” because it has a space in it.

```
juser@mgmt17:~$ ssh admin@10.1.1.7 “show namespace %22test ns%22”
...
```

Notification Rules

The ARX provides support via its API for external applications such as F5 Data Manager and for third-party applications to obtain file system change notification and configuration management services. Support for those API services includes the ability to define notification rules on the ARX which govern the creation of notification snapshots. Notification snapshots summarize the changes made to the virtual filesystem by the external applications that access it.

Notification rules are defined at the volume level, and provide information about API-based actions that are taken. When a notification rule is defined, it implicitly enables browsing of the snapshot directory in the volume. All volume-level snapshot options apply to snapshots from notification rules.

Notification rules support the use of privileged exec mode snapshot commands.

◆ Note

The snapshot remove *command must remove the oldest snapshot before it removes any other snapshots. Without this restriction, file changes may be lost when a snapshot in the middle is removed. If a snapshot is removed while it is being used, the client will be notified, and it should rescan the files.*

Executing the gbl-ns-vol mode CLI command notification rule creates a notification rule object and enters gbl-ns-vol-ntfy mode for defining the rule. This mode provides the following CLI commands:

- enable, which activates the current snapshot rule.
- report *prefix,* where *prefix* is the prefix to be prepended to each report name.
- retain *number,* where *number* is the number of reports to be held, up to 1024.
- schedule *name,* where *name* is the task schedule by which notification snapshots will be generated.

The command syntax is:

notification rule *name*

where *name* is the notification rule to be defined or edited.

For example, the following command sequence creates a notification rule named “accessReports”, defines it, and enables it:

```
bstnA(gbl-ns-vol[wwmed~/acct])# notification rule accessReports
bstnA(gbl-ns-vol-ntfy[wwmed~/acct~accessReports])# report ar_
bstnA(gbl-ns-vol-ntfy[wwmed~/acct~accessReports])# schedule
hourly
bstnA(gbl-ns-vol-ntfy[wwmed~/acct~accessReports])# retain 10
bstnA(gbl-ns-vol-ntfy[wwmed~/acct~accessReports])# enable
```

Statistics Monitoring

The ARX software provides robust statistics monitoring functionality that accumulates and, optionally, analyzes performance data for services and shares, notifying you if the evidence suggests a possible problem with their performance. This is done by comparing each sample of performance data against a moving historical average of previous data collected for the same sampled object, and generating a notification if that sample deviates from the moving average by a configured percentage over a configured number of sample intervals. By default, performance data is sampled every five minutes.

Notification of performance-related events occurs through log messages and/or SNMP traps. Each event type that is detected generates a separate event; however, multiple instances of the same event type are not reported more than once.

The data collected by the statistics monitor resides on each switch at /acopia/var/log/stats-logs as comma-separated value (CSV) files that can be opened in spreadsheet applications such as Microsoft Excel. These files can be copied off the switch as desired for further examination and for consultation with F5 support staff as necessary. One way of doing this is to use the collect stats-logs CLI command to gather the statistics monitoring files and write them to a location from which they can be shared easily.

Statistics monitoring data will not be written if the logs partition is 50% full or more.

Note

The various clear statistics *CLI commands will reset these statistics and will be indicated by a reset marker in the log file. Clearing the global configuration will clear the statistics monitor data; if you wish to retain this data, make certain to copy it off the switch prior to clearing the global configuration.*

Note

In a redundant ARX pair, the statistics monitor is active only on the active switch.

Execute the show stats-monitor CLI command to display the current statistics monitor configuration:

```
bstnA# show stats-monitor
Statistics Monitor Configuration
--------------------------------
  Status               Enabled
  Sampling Interval    60 seconds
  Max Data Size        500 MB
  Max Raw Data Size    80% of Max
  Used Data Size       7.4 MB
```

```
Used Raw Data Size   7.3 MB
Notification Parameters
----------------------
Applies To       Statistics    Traps        Metric    Parameters
----------       ----------    -----        ------    ----------
CIFS Shares      Errors        Disabled     Mov.Avg   >100%  for 6 Samples
CIFS Shares      RespTime      Enabled      Mov.Avg   >25%   for 2 Samples
```

Statistics monitoring is always enabled on the active switch.

The data size specifies the maximum amount of historical data that the statistics monitor is allowed to retain, and is not configurable. Similarly, the raw data size percentage, which controls the portion of the statistics log capacity that can be used by the raw data files, is not configurable, either.

Statistics Monitor Command Mode

Note

All commands related to configuring the statistics monitor or accessing the log files that it generates are in terminal beta mode. You must execute the terminal beta *CLI command in order for any of these commands to be available.*

After you've activated terminal beta mode, type stats-monitor in global configuration mode to enter gbl-stmon mode. Use this mode to access the statistics monitor log files, and to configure the statistics monitor's sampling and notification behaviors.

Changing the Data Sampling Interval

By default, performance data is sampled every 300 seconds.

To change the data sampling interval, use the sampling interval CLI command in gbl-stmon mode to specify the number of seconds between sampling intervals. This can range from a minimum of 30 to a maximum of 86,400 (one day).

For example:

```
bstnA(gbl-stmon)# sampling interval 120
```

This configures an interval of two minutes between data samples.

Changing the Event Notification Parameters

Use the notify command in gbl-stmon mode to enter gbl-stmon-nfy mode for configuring event notification for a CIFS service, NFS service, or filer share. This controls when notification is sent because recent data is anomalous in the context of similar performance statistics used to calculate the moving average for the relevant parameter.

The syntax is:

```
notify {cifs-service | filer-share | nfs-service}
```

This causes you to enter a submode corresponding to the type of service that you specified, such as gbl-stmon-nfy[~cifs-service] if you executed notify cifs-service. If you specify filer-share, you must also specify cifs or nfs. Within each submode, you can use the commands metric, moving-average, and trap to configure the corresponding event notification parameters for that service type.

Metric Configuration

In the current release, moving-average is the sole metric against which error counts and response times are compared to determine if they are within expected norms or are anomalous. This is not configurable, currently.

The command syntax that describes the current configuration is:

```
metric errors moving-average
metric response-time moving-average
```

Changing the Moving Average Configuration

You can configure how anomalies in error counts and response-times are compared to the moving average for the purpose of event notification. For each, there are two components to triggering a notification. The first is to specify the number of sample intervals over which the presence of an anomalous statistic will trigger an event notification; this changes the moving average window. The second is to specify the threshold percentage, which, when the statistic exceeds the moving average by that amount for the specified number of intervals, an event notification is triggered. No notification will occur until a full moving average window of data exists in the statistics monitor raw files for that event.

The syntax for these commands follows:

```
moving-average errors interval <1-65535>
moving-average errors threshold <percentage value>
moving-average response-time interval <1-65535>
moving-average response-time threshold <percentage value>
```

Changing the Trap Configuration

SNMP trap generation is disabled by default. You can specify whether an error count event or a response-time event sends an SNMP trap by using these commands:

```
trap errors
trap response-time
```

Configuring Email Notification

The Email group, *stats-monitor*, can be configured for trap notification for the following events:

- *filer-slow-clear*: Filer share slowness is cleared.

- *filer-slow-raise*: A filer share is slow.
- *service-slow-clear*: Service slowness is cleared.
- *service-slow-raise*: A service is slow.
- *filer-errors-clear*: Filer errors are normal.
- *filer-errors-raise*: Filer errors exceed the notification thresholds.
- *service-errors-clear*: Service errors are normal.
- *service-errors-raise*: Service errors exceed the notification thresholds.

For each event, the notification threshold is configured according to the number of occurrences within a specified time interval, as is done for events for other Email groups. Refer to the *ARX® CLI Reference* for complete details of the group stats-monitor command.

Statistics Monitor Log Files

List the statistics monitor log files that are present on the switch by executing the show stats-logs command.

For example:

```
bstnA(gbl-stmon)# show stats-logs


  stats-logs
      cifs-service_20121109_051448.raw.stats.csv             11/09 03:04   14 kB
      cifs-service_20121109_060000.hourly.stats.csv          11/09 03:04   5.1 kB
      cifs-service-auth_20121109_051448.raw.stats.csv        11/09 03:04   5.0 kB
      cifs-service-auth_20121109_060000.hourly.stats.csv     11/09 03:04   1.8 kB
      cifs-share_20121109_051448.raw.stats.csv               11/09 03:04   384 kB
      cifs-share_20121109_060000.hourly.stats.csv            11/09 03:04   45 kB
      cifs-work-queue_20121109_051448.raw.stats.csv          11/09 03:04   12 kB
      cifs-work-queue_20121109_060000.hourly.stats.csv       11/09 03:04   6.0 kB
      domain-controller_20121109_051448.raw.stats.csv        11/09 03:04   99 kB
      domain-controller_20121109_060000.hourly.stats.csv     11/09 03:04   11 kB
      ...
```

The following stats-log file types are supported:

- *cifs-service*: This file type contains data plane CIFS service statistics, with services identified by their FQDN.
- *cifs-service-auth*: This file type contains CIFS service authentication statistics. The services are identified by their FQDN.
- *cifs-share*: This file type contains data plane and control plane CIFS filer share statistics, with shares identified by their namespace, volume, and share name in the ARX configuration.
- *cifs-work-queue*: This file type contains CIFS work queue statistics. Work queues are identified by their volume group ID.
- *domain-controller*: This file type contains domain controller statistics. Domain controllers are identified by their IP address.
- *metadata*: This file type contains metadata statistics; objects in this file are identified by their namespace and volume.

- *metalog*: This file type contains metalog statistics; objects in this file are identified by their volume group ID.
- *migration*: This file type contains migration statistics. Objects in this file are identified by their namespace and volume.
- *nfs-service*: This file type contains data plane NFS service statistics, with services identified by their FQDN.
- *nfs-share*: This file type contains data plane and control plane NFS filer share statistics, with shares identified by their namespace, volume, and share name in ARX configuration.

File Naming Convention

Each statistics log file name indicates the type of object for the data contained in the file, and includes a file creation timestamp and the string "raw" or "hourly" to indicate whether the file contains an original data sample or an hourly aggregate of performance data.

For example:

cifs-service_20110621_170225.raw.stats.csv

Log File Contents

You can view a particular log file by executing the show stats-logs *filename* command. For example:

```
bstnA(gbl-stmon)# show stats-logs cifs-service_20121109_060000.hourly.stats.csv
BEGIN cifs-service version=2 timestamp=1352440800
TIMESTAMP,KEY,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,CO
UNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TO
TALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,
COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,
TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIM
E,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTE
R,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALT
IME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUN
TER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTA
LTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,CO
UNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TO
TALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,COUNTER,TOTALTIME,
COUNTER,COUNTER,COUNTER,COUNTER,COUNTER,COUNTER,COUNTER,COUNTER,COUNTER,COUNTER,COUNTER,COUNTER,CO
UNTER
Timestamp,fqdn,CheckDir-RTT,CheckDir-RTC,Close-RTT,Close-RTC,Create-RTT,Create-RTC,CreateDir-RTT,C
reateDir-RTC,Delete-RTT,Delete-RTC,DeleteDir-RTT,DeleteDir-RTC,Echo-RTT,Echo-RTC,FindClose2-RTT,Fi
ndClose2-RTC,Flush-RTT,Flush-RTC,LockingAndX-RTT,LockingAndX-RTC,LogoffAndX-RTT,LogoffAndX-RTC,Neg
otiate-RTT,Negotiate-RTC,NTCancel-RTT,NTCancel-RTC,NTCreateAndX-RTT,NTCreateAndX-RTC,NTRename-RTT,
NTRename-RTC,NTTransact/Create-RTT,NTTransact/Create-RTC,NTTransact/GetUserQuota-RTT,NTTransact/Ge
tUserQuota-RTC,NTTransact/Ioctl-RTT,NTTransact/Ioctl-RTC,NTTransact/NotifyChange-RTT,NTTransact/No
tifyChange-RTC,NTTransact/QuerySecurity-RTT,NTTransact/QuerySecurity-RTC,NTTransact/SetSecurity-RT
T,NTTransact/SetSecurity-RTC,Open-RTT,Open-RTC,OpenAndX-RTT,OpenAndX-RTC,QueryInfo-RTT,QueryInfo-R
TC,QueryInfoDisk-RTT,QueryInfoDisk-RTC,Read-RTT,Read-RTC,ReadAndX-RTT,ReadAndX-RTC,Rename-RTT,Rena
me-RTC,Seek-RTT,Seek-RTC,SessionSetupAndX-RTT,SessionSetupAndX-RTC,SetInfo-RTT,SetInfo-RTC,Transac
tion-RTT,Transaction-RTC,Transaction2/CreateDir-RTT,Transaction2/CreateDir-RTC,Transaction2/FindFi
rst2-RTT,Transaction2/FindFirst2-RTC,Transaction2/FindNext2-RTT,Transaction2/FindNext2-RTC,Transac
tion2/GetDFSReferral-RTT,Transaction2/GetDFSReferral-RTC,Transaction2/QueryFSInfo-RTT,Transaction2
/QueryFSInfo-RTC,Transaction2/QueryFileInfo-RTT,Transaction2/QueryFileInfo-RTC,Transaction2/QueryP
athInfo-RTT,Transaction2/QueryPathInfo-RTC,Transaction2/SetFileInfo-RTT,Transaction2/SetFileInfo-R
```

```
TC,Transaction2/SetPathInfo-RTT,Transaction2/SetPathInfo-RTC,TreeConnectAndX-RTT,TreeConnectAndX-R
TC,TreeDisconnect-RTT,TreeDisconnect-RTC,Write-RTT,Write-RTC,WriteAndX-RTT,WriteAndX-RTC,UnknownSM
B-RTT,UnknownSMB-RTC,Smb2Cancel-RTT,Smb2Cancel-RTC,Smb2ChangeNotify-RTT,Smb2ChangeNotify-RTC,Smb2C
lose-RTT,Smb2Close-RTC,Smb2Create-RTT,Smb2Create-RTC,Smb2Echo-RTT,Smb2Echo-RTC,Smb2Flush-RTT,Smb2F
lush-RTC,Smb2Ioctl-RTT,Smb2Ioctl-RTC,Smb2Lock-RTT,Smb2Lock-RTC,Smb2Logoff-RTT,Smb2Logoff-RTC,Smb2N
egotiate-RTT,Smb2Negotiate-RTC,Smb2QueryDirectory-RTT,Smb2QueryDirectory-RTC,Smb2QueryInfo-RTT,Smb
2QueryInfo-RTC,Smb2Read-RTT,Smb2Read-RTC,Smb2SessionSetup-RTT,Smb2SessionSetup-RTC,Smb2SetInfo-RTT
,Smb2SetInfo-RTC,Smb2TreeConnect-RTT,Smb2TreeConnect-RTC,Smb2TreeDisconnect-RTT,Smb2TreeDisconnect
-RTC,Smb2Write-RTT,Smb2Write-RTC,Smb2UnknownSMB-RTT,Smb2UnknownSMB-RTC,FormatError,NetworkError,Re
sourceError,SignatureError,SMBError,TimeoutError,Smb2FormatError,Smb2NetworkError,Smb2ResourceErro
r,Smb2SignatureError,Smb2SMBError,Smb2TimeoutError
1352443454,ac1.MEDARCH.ORG,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0
1352444654,ac1.MEDARCH.ORG,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0,0,0,0
1352448254,ac1.MEDARCH.ORG,0,0,49644,85,0,0,0,0,8030,6,9981,4,0,0,0,0,0,0,0,18492,40,7354,26,0,0
,228471,74,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,59657,12,0,0,782,5,0,0,17369,16,0,0,0,0,334775,104,0,0,
70840,3,0,0,389526,45,0,0,0,0,13000,70,1615,5,215673,62,4446,14,0,0,63491,64,3080,9,0,0,79011,112,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0
1352448254,labtests.MEDARCH.ORG,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,105690,42,144897,43,0,0,0,0,1094,6,0,0,1611,5,4751,21,0,0,700,4,0,0,49
727,28,40922,23,10949,11,3370,8,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
1352451854,ac1.MEDARCH.ORG,0,0,52261,91,0,0,240,1,8030,6,9981,4,0,0,0,0,0,0,0,0,31016,45,9028,32,0
,0,228471,74,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,70893,18,0,0,937,6,0,0,17369,16,0,0,0,0,356538,126,0,
0,114421,6,0,0,463310,67,0,0,0,0,27786,132,1615,5,300057,96,4446,14,0,0,73588,84,5358,18,0,0,10412
5,148,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0
1352451854,labtests.MEDARCH.ORG,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,111329,53,170745,48,0,0,0,0,1888,10,0,0,2902,8,5644,25,0,0,1771,10,0,0
,56596,34,40922,23,13630,16,6444,13,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
1352455454,ac1.MEDARCH.ORG,0,0,52261,91,0,0,240,1,8030,6,9981,4,0,0,0,0,0,0,0,0,31016,45,9028,32,0
,0,228471,74,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,70893,18,0,0,937,6,0,0,17369,16,0,0,0,0,356538,126,0,
0,114421,6,0,0,463310,67,0,0,0,0,27786,132,1615,5,300057,96,4446,14,0,0,73588,84,5358,18,0,0,10412
5,148,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0
1352455454,labtests.MEDARCH.ORG,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,111329,53,170745,48,0,0,0,0,1888,10,0,0,2902,8,5644,25,0,0,1771,10,0,0
,56596,34,40922,23,13630,16,6444,13,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
...
```

Within the file, the prefixes of the field names correspond to the name of the relevant CIFS or NFS protocol message; RTT stands for the accumulated round trip time, and RTC stands for the count of operations over which the RTT was accumulated. These response times are expressed in microseconds. For each error counter, there is a single column giving the error count.

For easier reading and better usability, copy the file off of the switch and view its contents using a spreadsheet application, or any other application capable of displaying a CSV file in an acceptable fashion.

New raw files are created when the existing file reaches 10 MB in size or the file contains 24 hours of data.

New hourly files are created when the existing file reaches 5 MB in size or the file contains seven days of data.

9

Troubleshooting Network Connections

Pinging an IP Address

From any mode, use the ping command to send an ICMP ECHO request to another interface:

ping *ip-address*

where ***ip-address*** (for example, 10.10.10.1) is the destination for the ping.

The ARX pings the IP address repeatedly until you stop it by pressing **<Ctrl-C>**. Once you stop it, a report summarizes the results of the ping test.

For example:

```
bstnA# ping 172.16.100.183
PING 172.16.100.183 (172.16.100.183) 0 data bytes
8 bytes from 172.16.100.183: icmp_seq=1 ttl=62 time=3 ms. from 2.3
8 bytes from 172.16.100.183: icmp_seq=2 ttl=62 time=<1 ms. from 2.3
8 bytes from 172.16.100.183: icmp_seq=3 ttl=62 time=<1 ms. from 2.3
8 bytes from 172.16.100.183: icmp_seq=4 ttl=62 time=<1 ms. from 2.3
8 bytes from 172.16.100.183: icmp_seq=5 ttl=62 time=<1 ms. from 2.3
8 bytes from 172.16.100.183: icmp_seq=6 ttl=62 time=<1 ms. from 2.3
8 bytes from 172.16.100.183: icmp_seq=7 ttl=62 time=<1 ms. from 2.3
8 bytes from 172.16.100.183: icmp_seq=8 ttl=62 time=<1 ms. from 2.3
8 bytes from 172.16.100.183: icmp_seq=9 ttl=62 time=<1 ms. from 2.3
8 bytes from 172.16.100.183: icmp_seq=10 ttl=62 time=<1 ms. from 2.3
8 bytes from 172.16.100.183: icmp_seq=11 ttl=62 time=<1 ms. from 2.3

-------172.16.100.183 ping statistics
11 packets transmitted, 11 packets received, 0% packet loss
round-trip min/avg/max 0/0/3 ms
bstnA# ...
```

Limiting the Ping Count

To limit the number of pings up front, use the count clause with the ping command:

ping *ip-address* count *number*

where

ip-address is the destination for the ping, and

number (1-10,000) is the number of pings to send.

For example:

```
bstnA# ping 172.16.100.183 count 4
PING 172.16.100.183 (172.16.100.183) 0 data bytes
8 bytes from 172.16.100.183: icmp_seq=1 ttl=62 time=<1 ms. from 2.3
8 bytes from 172.16.100.183: icmp_seq=2 ttl=62 time=<1 ms. from 2.3
8 bytes from 172.16.100.183: icmp_seq=3 ttl=62 time=<1 ms. from 2.3
8 bytes from 172.16.100.183: icmp_seq=4 ttl=62 time=<1 ms. from 2.3

-------172.16.100.183 ping statistics
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max 0/0/0 ms
bstnA# ...
```

Pinging From an Alternate Source IP

You can use an alternate source-IP address in a ping. By default, the source IP is calculated based on the originating processor's routing table.

There are three types of internal IP addresses that you can use as sources for ping: proxy IPs, Virtual IPs, and management IPs.

- The network processors use *proxy-IP* addresses to communicate with servers. Use the show ip proxy-addresses command (see *Showing all Proxy IPs*, on page 4-8 of the *ARX® CLI Network-Management Guide*) to list all proxy-IP addresses. Each network processor is assigned its own proxy-IP.
- The network processors use *virtual-IP* (VIP) addresses to communicate with clients. Use the show global server command (see *Showing All Global Servers*, on page 10-12 of the *ARX® CLI Storage-Management Guide*) to list all VIPs.
- There are two types of management interface: one in-band interface for each VLAN and one out-of-band (OOB) management interface. Use the show interface vlan command to list all VLAN-management interfaces (see *Listing all VLAN Management Interfaces*, on page 4-6 of the *ARX® CLI Network-Management Guide*). Use show interface mgmt to find the source address for the OOB Mgmt interface (see *Showing the Interface Configuration and Status*, on page 4-21 of the *ARX® CLI Network-Management Guide*).

If you set a source IP without setting the source processor, the source IP determines the processor. For example, if you choose the IP for the out-of-band management port, the CLI sends the ping from the SCM processor.

Use the source clause with the ping command to send an alternate source IP in the ICMP ECHO request:

`ping` *`ip-address`* `[from` *`slot.processor`*`] source` *`source-address`* `[count` *`number`*`]`

where

ip-address is the destination for the ping,

from *slot.processor* (optional, 1-2.1-12) chooses a source processor. If you choose the processor and the ***source-address***, they may conflict: the ping may not return to the processor that sent it. If you omit this, the CLI chooses the correct processor. In general, this should be omitted.

source-address is the source-IP address to send in the ICMP ECHO.

count *number* (optional, 1-10,000) limits the number of pings, as explained above.

For example, this command sends a ping with a source IP of 10.50.20.110:

```
bstnA> ping 10.53.2.10 source 10.50.20.110 count 2
PING 10.53.2.10 (10.50.20.110) 0 data bytes
```

```
8 bytes from 10.53.2.10: icmp_seq=1 ttl=0 time=100 ms. src addr 10.50.20.110
8 bytes from 10.53.2.10: icmp_seq=2 ttl=0 time=100 ms. src addr 10.50.20.110

-------10.53.2.10 ping statistics
2 packets transmitted, 2 packets received, 0% packet loss
round-trip min/avg/max 1/1/2 ms
bstnA> ...
```

Pinging the OOB Management Network

As implied above, you can ping the out-of-band (OOB) management network by using the OOB Mgmt address as the source IP. This ping starts at the SCM processor and goes through the management gateway. For example, the following command sequence finds the OOB management IP and gateway and then pings the management gateway. These pings go over the management network, not the client/server network:

```
prtlndA> show interface mgmt

Slot                    1
Interface               1
Description
Admin Status            Enabled
Link Status             Up
Speed                   1  Gb/s
Duplex                  Full
Auto Negotiation        Disabled
MAC Address             00:15:17:6b:a9:71
MTU Size                1500
IP Address              10.1.23.11
Subnet Mask             255.255.0.0
prtlndA> show ip route

Destination/Mask    Gateway          Cost Interface        Age
------------------------------------------------------------------------
0.0.0.0/0           192.168.74.1     128  VLAN74           9199
0.0.0.0/0           10.1.23.1        128  Mgmt             9430
192.168.74.0/24     0.0.0.0          0    VLAN74           Direct
192.168.74.0/24     0.0.0.0          128  VLAN             Direct
192.168.74.0/24     0.0.0.0          0    VLAN74           Direct
192.168.74.0/24     0.0.0.0          128  VLAN             Direct
192.168.78.0/24     0.0.0.0          128  VLAN             Direct
192.168.78.0/24     192.168.74.2     128  VLAN74           9214
10.1.23.0/24        0.0.0.0          0    Mgmt             Direct


prtlndA> ping 10.1.23.1 source 10.1.23.11 count 4
PING 10.1.23.1 (10.1.23.11) 0 data bytes
8 bytes from 10.1.23.1: icmp_seq=1 ttl=255 time=<1 ms. src addr 10.1.23.11
8 bytes from 10.1.23.1: icmp_seq=2 ttl=255 time=<1 ms. src addr 10.1.23.11
8 bytes from 10.1.23.1: icmp_seq=3 ttl=255 time=<1 ms. src addr 10.1.23.11
8 bytes from 10.1.23.1: icmp_seq=4 ttl=255 time=1 ms. src addr 10.1.23.11

-------10.1.23.1 ping statistics
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max 1/0/1 ms
prtlndA> ...
```

Using Traceroute

You can use the expect traceroute command to show each IP-router hop between the NSM and a given IP address. Like ping, the expect traceroute command is accessible from any mode:

expect traceroute *destination-address* [timeout *seconds*]

where

destination-address is the IP-address for the traceroute, and

timeout *seconds* (optional, 1-2096) limits the time for the traceroute process.

If a hop is unreachable, the command outputs asterisks (*) until you interrupt it or it times out. Use **<Ctrl-C>** to interrupt the expect traceroute command.

The packet starts at an inband (VLAN) management interface. Inband-management interfaces were discussed in *Configuring an In-Band (VLAN) Management Interface*, on page 4-4 of the *ARX® CLI Network-Management Guide*.

This sample command traces the route to the filer at 192.168.25.19:

```
bstnA# expect traceroute 192.168.25.19


 1  192.168.25.1  0.587 ms  0.553 ms  0.491 ms
 2  192.168.25.19  0.367 ms  0.363 ms  0.367 ms

bstnA# ...
```

Testing Throughput with TTCP

The Test TCP (TTCP) utility is a benchmarking tool to measure throughput over TCP or UDP connections. You can run a TTCP test from the SCM processor to any filer, client station, or other server that supports TTCP. (TTCP tools are freely available from the Internet.) After configuring a remote station to receive TTCP transmissions, use the expect ttcp transmit command to invoke a 10-second test:

expect ttcp transmit *ttcp-server-address* [timeout *seconds*]

where

ttcp-server-address is the IP-address for the pre-configured server, and

timeout *seconds* (optional, 1-2096) sets a time limit for the TTCP test.

When the test ends, the CLI shows the results in a one-line report. For example, the following command runs a test to the station at 192.168.98.55:

```
bstnA# expect ttcp transmit 192.168.98.55

Starting 10 second TCP transmit test. Hit ^C to abort...
 1003.0552 MB /  10.00 sec =  841.3741 Mbps 9 %TX

bstnA#
```

The %TX field at the end of the output is the percent-CPU used by the SCM processor.

Cancelling a TTCP Transmission

For cases where a connection error is causing the transmit to hang (for example, if TTCP is not configured at the server), use <Ctrl-C> to cancel the test. For example,

```
bstnA# expect ttcp transmit 192.168.97.55

Starting 10 second TCP transmit test. Hit ^C to abort...<Ctrl-C>
bstnA#
```

The optional **timeout** argument also cancels the test.

Testing a RON Tunnel

To test the RON connection from one ARX to another, run expect ttcp server at the receiving switch and then expect ttcp transmit from the sending switch. (For information on creating a RON tunnel, refer back to *Creating a Tunnel to Another ARX*, on page 6-4 of the *ARX® CLI Network-Management Guide*.) Like the previous expect commands, you run expect ttcp server from any mode:

expect ttcp server [timeout *seconds*]

The CLI stops to wait for a TTCP transmission from another switch. The CLI remains blocked until you press <Ctrl-C> to stop waiting. For example, the following command waits for a TTCP transmission on the switch, "prtlndA:"

```
prtlndA# expect ttcp server
Waiting for TTCP client. Hit ^C to abort...
```

Go to the sending switch to run the test. The IP address to use for the server switch is the.1 address on the server's private subnet. Use the show ron route command (see *Showing the RON Routes*, on page 6-10 of the *ARX® CLI Network-Management Guide*) to find the private subnet for the server switch. To continue the example, this command sequence finds the private subnet for "prtlndA," then runs a TTCP test over the RON tunnel:

```
bstnA# show ron route

Default Policy
--------------

  Destination            Subnet           via Tunnel        Milliseconds
  ------------------------------------------------------------------------
  canbyA             169.254.116.0/24     toCanby               0.1
  newptA            169.254.151.64/26     toNewport             0.1
  provA             169.254.195.128/26    toProvidence          0.1
  prtlndA            169.254.140.0/24     toPortland            0.1
  prtlndB            169.254.104.0/24     toPortlandB           0.1
  stkbrgA            169.254.131.0/24     toStockbridge         0.3
  stoweA              169.254.76.0/24     toStowe               0.1

bstnA# expect ttcp transmit 169.254.24.1

Starting 10 second TCP transmit test. Hit ^C to abort...
  586.5138 MB /  10.00 sec =  491.9229 Mbps 12 %TX 23 %RX

bstnA#
```

When the test is over, go back to the server switch and press <Ctrl-C>:

```
Waiting for TTCP client. Hit ^C to abort... <Ctrl-C>
prtlndA#
```

The server switch can only receive TTCP transmissions from another ARX on the same RON.

Tracing NSM Processes in the “fastpath” Log

To debug a connectivity issue, you can trace the software processes on an NSM processor. A traced NSM processor sends its log messages to a file named “fastpath,” which is similar to the “syslog” file described in the previous chapter. From cfg mode, use the logging fastpath processor command to choose an NSM processor to trace:

logging fastpath processor *slot.processor*

where

slot (2 on an ARX-4000; 1 on an ARX-2500, ARX-2000, ARX-1500, or ARX-500) is the slot number of the desired NSM, and

processor (1-12) is the processor number. Use show processors for a complete list of processors (and their modules and slots) on the ARX.

Important

NSM processes can be very verbose on a busy ARX. This slows client/server traffic at the chosen processor. Use this option only under the guidance of F5 personnel.

The first time you issue any logging fastpath command, the CLI issues a warning about the performance impact. Enter **yes** to continue. This prepares the processor for minimal logging. You can increase the volume of logs for individual processes and process groups on the NSM, as shown below.

For example, the following command configures NSM processor 2.1 for logging:

```
bstnA(cfg)# logging fastpath processor 2.1

Enabling fastpath logging adversely impacts the performance of the switch.

Proceed? [yes/no] yes
bstnA(cfg)# ...
```

NSM-Log Components

Every NSM-log message originates from a log component on the NSM. A *log component* is a source of fastpath-log messages, typically an internal NSM process or group of processes. They are similar to the log components

that write to the syslog, described in *Log Components*, on page 8-9. The table below is an alphabetical list of all NSM log components, along with a brief description.

Log Component	Description
NSM_BSD	Board Support Package.
NSM_CACHE	NSM Cache.
NSM_CIFS	CIFS Proxy on the data plane.
NSM_CM	Connection Manager.
NSM_DGRAM	Datagram.
NSM_DME	Data Mover Engine (DME) proxy on the data plane. This migrates files from one back-end filer to another.
NSM_DMS	NSM DMS.
NSM_DT	Direct Transport.
NSM_ECHO	Packet Echo Service.
NSM_GWMON	Gateway Monitor.
NSM_HA	High-Availability.
NSM_HOT	Hot files.
NSM_IPA	AIPC Server and Client, for internal communication between network processes.
NSM_LIB	Networking Library Functions.
NSM_LOG	System Logging.
NSM_LOOP	Packet Loop Service.
NSM_MSG	Inter-NSM Messaging.
NSM_NAT	The Start of tem message categories.
NSM_NAT	NAT Subsystem.
NSM_NET_HEALTH	Network Health Checks.
NSM_NFS	NFS Proxy on the data plane.
NSM_NFS_CLIENT	NSM NFS client.
NSM_NFS_DNAS	The NFS proxy interactions with DNAS (becoming obsolete).
NSM_NFS_FC	NFS File Cache Management Proxy.
NSM_NFS_IPC	The NFS Proxy IPC (obsolete?).
NSM_NFS_MOUNT	Obsolete.
NSM_NFS_SERVER	NFS file server state machine.
NSM_NFS_STATE	The NFS proxy state machine (becoming obsolete).
NSM_NLM	NFS Lock Manager (NLM) proxy on the data plane.
NSM_PING	PING Application.
NSM_PORT	Port Management.
NSM_PRESTO	Presto Console Log.

Log Component	Description
NSM_RON	RON Fastpath Proxy.
NSM_SMB2	CIFS Proxy.
NSM_SNTP	SNTP.
NSM_STATS	Statistics from the data plane.
NSM_TEME	Fastpath environment-kernel.
NSM_TFTP	TFTP Client.
NSM_TTCP	TTCP Service.
NSM_VIP	VIP Processing.
NSM_VIP_FENCE	VIP Fencing, which can occur when an ARX volume is creating a coordinated snapshot. The VIP fence, while it is raised, prevents clients from changing any volume state while its back-end filers are each taking their snapshots.
NSM_XID_MAP	NFS RPC XID Mapping.

Adjusting NSM Logging Levels

Each NSM-log component has a logging level that determines the volume of logs that it generates. You can set a different logging level for every NSM-log component, allowing you to focus attention on a single NSM component or a small group of them.

From cfg mode, use the logging fastpath component command to change the logging level for one NSM-log component:

logging fastpath component *name level*

where

name (1-128 characters) is an NSM-log component from the list of *NSM-Log Components*, on page 9-9, and

level (0-10) chooses the logging level for the component. A 0 (zero) stops all messages from the component. A 1 causes the component to log non-recoverable errors only; this is the default. A 2 or 3 adds warnings and recoverable errors. A 4 adds logs about internal configuration changes. Levels 5 through 10 add increasing densities of per-packet messages.

Important

The log output can get very verbose. A busy NSM processor may fail if this level is set too high for too long. You should only use this feature under the strict guidance of F5 personnel.

The first time you enter any logging fastpath command, the CLI warns you about the performance impact. If the warning appears, enter **yes** to proceed.

This logging level applies to all NSMs that have been enabled for logging with logging fastpath processor, described earlier.

For example, the following command sets the logging level to 6 (a low volume of per-packet logs) for the NSM_CIFS component:

```
bstnA(cfg)# logging fastpath component NSM_CIFS 6
bstnA(cfg)# ...
```

Filtering the Log Messages

You can focus the log output by filtering for a particular subnet, IP address, or other search string. Add a filter clause to the end of the logging fastpath component command:

logging fastpath component *name* filter *match-string* {include | exclude}

where

name (1-128 characters) is an NSM-log component from the list of *NSM-Log Components*, on page 9-9.

match-string (1-80 characters) is a string to match against. Quote the string if it contains any spaces.

include | exclude is a required choice: **include** collects any log message that matches the ***match-string***, while **exclude** omits all matching log messages and gathers all other messages.

You can repeat this command multiple times to expand or narrow the search further. A packet that matches any of the entered filters is included in the log file (or excluded from it, if you use the exclude keyword). Log messages are matched in the order that you enter them, so a command to include logs with "10.10.53.99" is ineffective if a previous filter excludes "10.10."

As above, the CLI may prompt with a performance warning before it sets the filter. If necessary, enter **yes** to continue.

For example, this command sequence filters all NSM_CIFS logs down to packets containing either 192.168.25.15, 172.16.100.183, or both:

```
bstnA(cfg)# logging fastpath component NSM_CIFS filter 192.168.25.15
bstnA(cfg)# logging fastpath component NSM_CIFS filter 172.16.100.183
bstnA(cfg)# ...
```

Disabling Log Messages for an NSM Component

You can stop all log messages from a given NSM component; this applies to every instance of the component, on every NSM processor that is enabled for logging. Use the no form of the logging fastpath component command:

no logging fastpath component *name*

where ***name*** identifies the NSM-log component to disable (see *NSM-Log Components*, on page 9-9), and

This is equivalent to logging fastpath component *name* 0.

For example, the following command disables logging from the NSM_CIFS component:

```
bstnA(cfg)# no logging fastpath component NSM_CIFS
bstnA(cfg)# ...
```

Showing All NSM-Log Components and Levels

Use the show fastpath logging command to see the logging levels for all NSM-log components:

```
show fastpath logging
```

This displays two tables. The first lists all of the NSM processors with logging enabled. The second lists all of the components that are being traced, along with any filtering configuration.

For example:

```
bstnA(cfg)# show fastpath logging
   Slot         Processor
-------------------------

    1              1
    1              2
                  3


        Component      Trace Level     Filter Type    Filter String
-----------------------------------------------------------------------

       NSM_CIFS            6
       NSM_CIFS                        inclusive      192.168.25.15
       NSM_CIFS                        inclusive      172.16.100.183

...
```

Disabling Logs from an NSM Processor

Remember to stop all NSM processors from logging as soon as you finish your tracing session; NSM logging degrades performance. Use no logging fastpath processor to disable logging from a certain processor:

```
no logging fastpath processor slot.processor
```

where

slot (2 on an ARX-4000; 1 on an ARX-2500, ARX-2000, ARX-1500, or ARX-500) is the slot number of the desired NSM, and

processor (1-12) is the processor number. Use show processors for a complete list of processors (and their modules and slots) on the ARX.

For example, the following command sequence stops three NSM processors from logging:

```
bstnA(cfg)# no logging fastpath processor 2.1
bstnA(cfg)# no logging fastpath processor 2.2
bstnA(cfg)# no logging fastpath processor 2.3
```

```
bstnA(cfg)# ...
```

Accessing the fastpath Log

You can use show logs fastpath to view the fastpath file and tail logs fastpath follow to follow it as it grows. To search through the fastpath file for a pattern, use grep *pattern* logs fastpath. These are the same commands used to access the syslog file, as described in.*Accessing the Syslog*, on page 8-7.

For example, the following command shows the fastpath log file:

```
bstnA(cfg)# show logs fastpath
2006-11-22T08:43:29.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: cifs_conn_hdlr.c:117:  New
cifs connection cid:19, type:7 remote-addr:172.16.100.183
2006-11-22T08:43:29.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: cifs_nbss.c:50:
nbss-sess-req: session-request rxed from client:172.16.100.183
2006-11-22T08:43:29.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: cifs_nbss.c:279:  netbios
session request 72 from client:172.16.100.183
2006-11-22T08:43:29.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: cifs_nbss.c:288:     cifss
nbss name :41433120202020202020202020202020
2006-11-22T08:43:29.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: cifs_nbss.c:289:      cifss
nbss name in the session request:31302E35352E31302E31303620202020
2006-11-22T08:43:29.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: cifs_nbss.c:304:
netbios session established
2006-11-22T08:43:29.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: smb_neg.c:320:  neg-prot
request from cid:19, client:172.16.100.183 - send to dnas
2006-11-22T08:43:29.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: smb_neg.c:245:  send
neg-prot reply back to cid:19 client:172.16.100.183
2006-11-22T08:43:29.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: smb_sess.c:891:
session-setup request from cid:19, mid:1, client:172.16.100.183 - send to dnas
2006-11-22T08:43:29.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: smb_sess.c:469:   rxed dnas
sess resp:cid:19, mid:1, user/domain:MEDARCH\LAB
2006-11-22T08:43:29.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: smb_sess.c:647:  send
session-setup reply to client:172.16.100.183 cid:19 uid:1
2006-11-22T08:43:29.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: smb_tree.c:838:  rxed tree
connect andx req from client:172.16.100.183, cid:19, uid:1
2006-11-22T08:43:29.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: smb_tree.c:345:   rxed new
session resp from dnas for cid:19, mid:1
2006-11-22T08:43:29.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: smb_tree.c:224:   send tc to
dnas:33, TID:1
2006-11-22T08:43:29.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: smb_tree.c:1816:   rxed tree
connect resp from dnas:send reply client:172.16.100.183,cid:19,mid:1,tid:1
2006-11-22T08:43:30.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: cifs_proxy.c:415:
cifss-conn:CIFS connection DOWN event. Client:192.168.25.15:53711  VIP:192.168.25.15:139
2006-11-22T08:43:30.000+0000:bstnA:5-5-NSM_TX-2124:NET_APP-0-7-MSG7:: cifs_conn_hdlr.c:733:  CIFS
client conn teardown cid :19 client:172.16.100.183
...
```

NSM-Log-Message Syntax

Each line in the fastpath log file uses the following syntax:

utc.uuu+tz:switch:slot-proc-board-pid:cmp-ins-sev-id:: msg

This is the same format as is used in the syslog file (recall *Syslog Syntax*, on page 8-16). For example,

```
2006-11-22T08:43:30.000+0000:bstnA:3-2-NSM_TX-2124:NET_APP-0-7-MSG7:: cifs_proxy.c:415:
cifss-conn:CIFS connection DOWN event. Client:192.168.25.15:53711  VIP:192.168.25.15:139
```

Where

- ***utc*** (`2006-11-22T08:43:30` in the example) is the time in UTC.
- ***uuu*** (`000`) is the millisecond time fraction.
- ***+tz*** (`+0000`) is the hours off UTC (+*nnnn* or -*nnnn*).
- ***switch*** (`bstnA`) is the switch name.
- ***slot-proc-board*** (`3-2-NSM_TX`) is the chassis slot, processor number, and board type (always NSM). Use the show processors command for a list of all processors and their associated modules.
- ***pid*** (`2124`) is the Process ID (PID).
- ***cmp*** (`NET_APP`) is the log-component name. Refer to *NSM-Log Components*, on page 9-9.
- ***ins*** (`0`) is the process’s instance number.
- ***sev*** (`7`) is the message severity, from 7 (debug) to 2 (critical).
- ***id*** (`MSG7`) is the message-catalog ID, a unique ID for each log message. MSG*n* is an ID for an uncatologued message. The *n* is the severity of the message, from 7 (debug) down to 2 (critical).
- ***msg*** (`cifs_proxy.c:415: cifss-conn: CIFS ...`) is the message text itself.

Capturing IP Traffic in a File

From priv-exec mode, you can use the capture session command to start capturing IP traffic and stream it into a file:

```
capture session session-id ip ip-address [and-ip ip2] vlan vlan-id file prefix
```

where

session-id (1-4; 1-2 on the ARX-500) identifies this capture session, so that you can close it later.

ip-address is the address to match against. The capture includes all packets whose source IP or destination IP matches this address.

and-ip *ip2* (optional) is a second IP address. If you select this option, the capture includes only the packets between *ip-address* and *ip2* in both directions.

vlan *vlan-id* (1-4095) specifies the VLAN.

prefix (1-255 characters) is the prefix you choose for the file name. Each file is named as follows:

prefix.cap

Files are named with this format if there are more than one:

prefix_nnnnn_yyyymmddHHMMSS.cap

where *prefix* is what you choose here, *nnnnn* is a file number for this session (there can be multiple files for each capture session, as explained later), and *yyyymmddHHMMSS* is the date and time the file was created.

A capture session begins when you invoke the command and ends when the capture file reaches its maximum size, or when you stop it with no capture session. The maximum file size and the no capture session command are both described below.

The ARX prepares for the capture session by internally re-routing some of its IP traffic. This may take several minutes, especially on a busy system. After you enter the command, a message appears to warn you of the possible delay.

◆ Note

Jumbo frames are not included in a capture session.

For example, the following command sequence exits to priv-exec mode, captures traffic between two IP addresses, and places it in a file with the prefix, “clientCap.” This only captures packets on VLAN 25.

```
bstnA(cfg)# end
bstnA# capture session 1 ip 172.16.100.183 and-ip 192.168.25.10 vlan 25 file clientCap

% INFO: Configuring network; this may take up to 6 minutes to complete.

bstnA# ...
```

Changing the Size and Number of Capture Files

By default, the capture session stops when it fills a single file with 16,000 kilobytes of data. You can use the filesize and/or filecount options to change the size and number of capture files:

capture session *session-id* ip *ip-address* [and-ip *ip2*] vlan *vlan-id* file *prefix* [filesize *kilobytes*] [filecount *count*]

where

session-id, ***ip-address***, ***ip2***, ***vlan-id***, and ***prefix*** are all described above.

filesize *kilobytes* (optional, 1-50,000; 1-1,000,000 on an ARX-4000) limits the size of the file; once the file reaches this limit, the capture stops. (One kilobyte is 1,000 bytes, not 1,024.) The default is 16,000 kilobytes.

filecount *count* (optional, 1-10) sets up a series of capture files; when the capture reaches the filesize limit, it can close the current capture file and start a new one. The default is 1 file. If you set a *count* of 2 or more, the capture process rotates the capture files indefinitely, and does not stop capturing packets until you use no capture session.

For example, the following command sequence starts another capture session that is limited to a single 500-Kbyte file:

```
bstnA(cfg)# end
bstnA# capture session 4 ip 192.168.25.19 vlan 25 file shadow_usage filesize 500

% INFO: Configuring network; this may take up to 6 minutes to complete.

bstnA# ...
```

Focusing on CIFS Traffic

You can filter out all packets in the capture except those related to CIFS. Specifically, you can focus on traffic to or from the following ports on your chosen IP address(es):

- UDP/88 or TCP/88
- UDP/137
- UDP/138
- TCP/139
- TCP/445

To focus only on traffic to or from the above ports, use the optional protocol cifs arguments at the end of the command:

capture session *session-id* ip *ip-address* ... [protocol cifs]

where

all of the options except the final one are described above.

protocol cifs (optional) filters out all traffic except CIFS-related traffic.

For example, the following command sequence starts another capture session for CIFS-related traffic:

```
bstnA(cfg)# end
bstnA# capture session 2 ip 192.168.25.27 vlan 25 file fsrvr protocol cifs

% INFO: Configuring network; this may take up to 6 minutes to complete.

bstnA# ...
```

Filtering Out CIFS Traffic

You can use protocol non-cifs to filter out all CIFS-related traffic from the capture file(s).

capture session *session-id* ip *ip-address* ... [protocol non-cifs]

Specifically, this ignores all packets to or from the list of ports in the previous section.

Stopping the Capture

From priv-exec mode, use no capture session to stop the capture immediately or remove a capture session that has stopped:

no capture session *session-id* [no-merge]

where

session-id (1-4) identifies the capture session to close.

no-merge (optional) prevents the command from automatically merging the output files. This is only relevant if the capture session generated 2 or more files, as specified with the filecount option. If you omit this, the merged filename is "*prefix*.cap" and contains all captured packets in chronological order.

Any capture session persists until you use this command to stop it, whether or not it has stopped writing to a file.

For example, this stops the session that we started in an earlier example:

```
bstnA# no capture session 1

% INFO: Configuring network; this may take up to 6 minutes to
complete.

bstnA# ...
```

Capturing All Proxy-IP Traffic

You can monitor all traffic to and from the back-end filers with the proxy-all option. This captures any packet whose source or destination IP address is any of the proxy-IP addresses on the ARX (recall *Adding a Range of Proxy-IP Addresses*, on page 4-7 of the *ARX® CLI Network-Management Guide*). This option does not require a VLAN ID:

capture session *session-id* proxy-all file *prefix* [filesize *kilobytes*] [filecount *count*]

For example, the following command captures two 16,000-KByte files of proxy-IP traffic:

```
bstnA(cfg)# end
bstnA# capture session 2 proxy-all file proxyTraffic filecount 2

% INFO: Configuring network; this may take up to 6 minutes to complete.

bstnA# ...
```

Note

The proxy-all *option is not available on ARX-VE.*

Listing All Capture Files

A capture file is accessible from the "capture" directory: use show capture to see the file listing:

show capture

For example, the following command shows several capture files:

```
bstnA# show capture


  capture
      clientCap.cap           Oct 28 01:23  15 MB
      nasTraffic.cap          Oct 28 01:29  5.7 MB
      ntaps_00001_20091028053845.cap Oct 28 01:38  24 B
      proxyTraffic_02032_20091028052930.cap Oct 28 01:29  150 kB
      proxyTraffic_02033_20091028052930.cap Oct 28 01:29  55 kB
      shadow_usage.cap        Oct 28 02:35  24 B

bstnA# ...
```

Showing Capture Details

The capture-file contents are the same as the output from TShark, a commonly-used Network Analyzer. TShark is the command-line version of Wireshark (formerly Ethereal), a graphical interface to the same Network-Analyzer functions. To view the capture file, use show capture *file-name*:

show capture *file-name*

where ***file-name*** (1-1024 characters) is the name of the chosen capture file.

For example, the following command sequence shows the contents of "clientCap.cap:"

```
bstnA# show capture clientCap.cap
...
48672  37.714814 172.16.100.183 -> 192.168.25.10 NFS V3 SETATTR Call, FH:0x92090400
48673  37.715214 192.168.25.10 -> 172.16.100.183 NFS V3 SETATTR Reply (Call In 48672)
48674  37.715413 172.16.100.183 -> 192.168.25.10 NFS V3 LOOKUP Call, DH:0x04090400/hp_psaux.ver
48675  37.715663 192.168.25.10 -> 172.16.100.183 NFS V3 LOOKUP Reply (Call In 48674)
Error:NFS3ERR_NOENT
48676  37.715762 172.16.100.183 -> 192.168.25.10 NFS V3 CREATE Call, DH:0x04090400/hp_psaux.ver
Mode:UNCHECKED
48677  37.719208 192.168.25.10 -> 172.16.100.183 NFS V3 CREATE Reply (Call In 48676)
48678  37.719556 172.16.100.183 -> 192.168.25.10 NFS V3 WRITE Call, FH:0x98090400 Offset:0 Len:114
UNSTABLE
48679  37.720406 192.168.25.10 -> 172.16.100.183 NFS V3 WRITE Reply (Call In 48678) Len:114
UNSTABLE
48680  37.720505 172.16.100.183 -> 192.168.25.10 NFS V3 COMMIT Call, FH:0x98090400
48681  37.725897 192.168.25.10 -> 172.16.100.183 NFS V3 COMMIT Reply (Call In 48680)
48682  37.726146 172.16.100.183 -> 192.168.25.10 NFS V3 LOOKUP Call, DH:0x04090400/hp_psaux.ver
48683  37.726644 192.168.25.10 -> 172.16.100.183 NFS V3 LOOKUP Reply (Call In 48682),
FH:0x98090400
48684  37.726745 172.16.100.183 -> 192.168.25.10 NFS V3 SETATTR Call, FH:0x98090400
48685  37.733885 192.168.25.10 -> 172.16.100.183 NFS V3 SETATTR Reply (Call In 48684)
48686  37.734333 172.16.100.183 -> 192.168.25.10 NFS V3 GETATTR Call, FH:0x98090400
48687  37.734533 192.168.25.10 -> 172.16.100.183 NFS V3 GETATTR Reply (Call In 48686)  Regular
File mode:0654 uid:5198 gid:1007
48688  37.736529 172.16.100.183 -> 192.168.25.10 NFS V3 SETATTR Call, FH:0x98090400
48689  37.736829 192.168.25.10 -> 172.16.100.183 NFS V3 SETATTR Reply (Call In 48688)
Error:NFS3ERR_PERM
48690  37.736978 172.16.100.183 -> 192.168.25.10 NFS V3 GETATTR Call, FH:0x98090400
48691  37.737178 192.168.25.10 -> 172.16.100.183 NFS V3 GETATTR Reply (Call In 48690)  Regular
File mode:0654 uid:5198 gid:1007
48692  37.737278 172.16.100.183 -> 192.168.25.10 NFS V3 GETATTR Call, FH:0x98090400
48693  37.737477 192.168.25.10 -> 172.16.100.183 NFS V3 GETATTR Reply (Call In 48692)  Regular
File mode:0654 uid:5198 gid:1007
48694  37.737577 172.16.100.183 -> 192.168.25.10 NFS V3 SETATTR Call, FH:0x98090400
48695  37.738276 192.168.25.10 -> 172.16.100.183 NFS V3 SETATTR Reply (Call In 48694)
48696  37.738526 172.16.100.183 -> 192.168.25.10 NFS V3 LOOKUP Call, DH:0x04090400/nvram.stamp
48697  37.738725 192.168.25.10 -> 172.16.100.183 NFS V3 LOOKUP Reply (Call In 48696)
Error:NFS3ERR_NOENT
...
```

Summarizing a Capture File

You can show a high-level summary of the IP traffic captured in one of the above files. To show the summary, specify the capture file and use an optional summary keyword in the show capture command:

show capture *capture-file* summary [cifs | non-cifs]

where

capture-file (1-1024 characters) is the name of the chosen capture file.

summary specifies the output format.

cifs | non-cifs (optional) is a filter for the output. The **cifs** flag shows only CIFS-related traffic (to and from the ports listed earlier), and the **non-cifs** flag filters out all CIFS-related traffic.

You can only use this summary option when the capture session is stopped. Use no capture session to stop the session, even if the session has stopped writing to its only output file.

The output is the same as that of TShark with the **-z** option. It shows a table of UDP conversations and another table of TCP conversations, followed by separate RTT tables for NFSv2, NFSv3, and SMB (or CIFS). The first two tables, UDP Conversations and TCP Conversations, show the numbers of UDP frames and bytes exchanged between each pairing of IP addresses. Each IP-address pair is typically a VIP or proxy-IP and some external address, and appears on one line. These are followed by three tables with RTT statistics for specific NFS-procedure calls and/or CIFS commands.

For example, the following command summarizes a capture file with UDP conversations, TCP conversations, NFSv3 calls, and CIFS commands:

```
bstnA# show capture nasTraffic.cap summary
================================================================================
UDP Conversations
Filter:<No Filter>
                                          |       <-      | |       ->      | |     Total     |
                                          | Frames  Bytes | | Frames  Bytes | | Frames  Bytes |
192.168.25.21:2049   <-> 192.168.25.34:640        827   1029774     855    175726    1682   1205500
192.168.25.21:2049   <-> 192.168.25.31:640         20      6972      20      3992      40     10964
192.168.25.31:640    <-> 192.168.25.21:sunrpc       0         0       2       196       2       196
================================================================================
================================================================================
TCP Conversations
Filter:<No Filter>
                                          |       <-      | |       ->      | |     Total     |
                                          | Frames  Bytes | | Frames  Bytes | | Frames  Bytes |
192.168.25.31:63805  <-> 192.168.25.21:445         20     11475      26      3246      46     14721
192.168.25.55:11884  <-> 192.168.25.21:445         11      1869      14      1981      25      3850
192.168.25.21:2049   <-> 192.168.25.31:649          2       222       1       226       3       448
192.168.25.21:2049   <-> 192.168.25.31:648          2       222       1       226       3       448
192.168.25.21:2049   <-> 192.168.25.31:647          2       222       1       226       3       448
192.168.25.31:63664  <-> 192.168.25.21:445          1       127       2       193       3       320
192.168.25.21:2049   <-> 192.168.25.31:644          2       182       1        86       3       268
192.168.25.21:2049   <-> 192.168.25.31:654          2       182       1        86       3       268
192.168.25.21:4046   <-> 192.168.25.31:643          1        56       1        60       2       116
192.168.25.31:641    <-> 192.168.25.21:sunrpc       1        60       1        56       2       116
================================================================================

=====================================================================
NFS Version 2 RTT Statistics:
Filter:
Procedure        Calls   Min RTT   Max RTT   Avg RTT
NULL                 0   0.00000   0.00000   0.00000
GETATTR              0   0.00000   0.00000   0.00000
SETATTR              0   0.00000   0.00000   0.00000
ROOT                 0   0.00000   0.00000   0.00000
LOOKUP               0   0.00000   0.00000   0.00000
READLINK             0   0.00000   0.00000   0.00000
READ                 0   0.00000   0.00000   0.00000
WRITECACHE           0   0.00000   0.00000   0.00000
WRITE                0   0.00000   0.00000   0.00000
CREATE               0   0.00000   0.00000   0.00000
REMOVE               0   0.00000   0.00000   0.00000
RENAME               0   0.00000   0.00000   0.00000
LINK                 0   0.00000   0.00000   0.00000
SYMLINK              0   0.00000   0.00000   0.00000
```

```
MKDIR                0   0.00000   0.00000   0.00000
RMDIR                0   0.00000   0.00000   0.00000
READDIR              0   0.00000   0.00000   0.00000
STATFS               0   0.00000   0.00000   0.00000
=====================================================================

=====================================================================
NFS Version 3 RTT Statistics:
Filter:
Procedure        Calls   Min RTT   Max RTT   Avg RTT
NULL                 7   0.00005   0.00127   0.00048
GETATTR             92   0.00004   0.00214   0.00036
SETATTR              2   0.00099   0.03040   0.01570
LOOKUP              70   0.00004   0.00445   0.00042
ACCESS               0   0.00000   0.00000   0.00000
READLINK             0   0.00000   0.00000   0.00000
READ                 2   0.00014   0.02093   0.01053
WRITE              652   0.00017   0.11836   0.00512
CREATE               0   0.00000   0.00000   0.00000
MKDIR                0   0.00000   0.00000   0.00000
SYMLINK              0   0.00000   0.00000   0.00000
MKNOD                0   0.00000   0.00000   0.00000
REMOVE               2   0.00023   0.00025   0.00024
RMDIR                0   0.00000   0.00000   0.00000
RENAME               0   0.00000   0.00000   0.00000
LINK                 0   0.00000   0.00000   0.00000
READDIR              2   0.00030   0.00815   0.00422
READDIRPLUS          0   0.00000   0.00000   0.00000
FSSTAT               8   0.00009   0.00379   0.00099
FSINFO               1   0.00014   0.00014   0.00014
PATHCONF             0   0.00000   0.00000   0.00000
COMMIT               0   0.00000   0.00000   0.00000
=====================================================================

=====================================================================
SMB RTT Statistics:
Filter:
Commands                  Calls   Min RTT   Max RTT   Avg RTT
Echo                          1   0.00011   0.00011   0.00011
Tree Disconnect               2   0.00038   0.00079   0.00059
Negotiate Protocol            2   0.00093   0.00684   0.00389
Session Setup AndX            4   0.00768   0.09215   0.03479
Logoff AndX                   2   0.00074   0.00124   0.00099
Tree Connect AndX             2   0.00023   0.00174   0.00099

Transaction2 Commands     Calls   Min RTT   Max RTT   Avg RTT
FIND_FIRST2                   2   0.00050   0.00264   0.00157
QUERY_FS_INFO                 4   0.00009   0.00189   0.00103
QUERY_PATH_INFO               2   0.00124   0.00189   0.00157

NT Transaction Commands   Calls   Min RTT   Max RTT   Avg RTT
=====================================================================
bstnA# ...
```

Showing All Capture Sessions

Use the show capture sessions command to show the current status of all capture sessions, if any are running:

```
show capture sessions
```

For example, the following command shows two active capture sessions:

```
bstnA# show capture sessions

Session Ip               Additional Ip    VLAN File Size  File Count
------- ---------------- ---------------- ---- ---------- ----------
2       proxy-all        proxy-all        25   100        2
3       n/a              192.168.25.21    25   100        3


Session State      File Name
------- ---------- ------------------------------
2       Capturing  proxyTraffic
3       Capturing  nasTraffic

bstnA#
```

Merging the Files from a Multi-File Capture

If you created multiple files from a single capture session and preserved them as separate files, you can later use the capture merge command to merge them into a single file. The capture session command may produce multiple capture files if the ARX reboots in the middle of the session, or if someone stops the capture session with the no-merge option (described earlier). The merge operation sorts all of the captured packets into chronological order, starting with the earliest.

Like the capture session command, you can invoke capture merge from priv-exec mode:

capture merge file *prefix*

where ***prefix*** (1-256 characters) is the prefix that is common to the files that you want to merge. This is typically the same prefix used to create the files in the capture session command. This is also the name of the output file created by the merge operation (*prefix*.cap).

For example, the following command sequence shows four capture files, merges two of them from the same session, and then shows that there are three files remaining:

```
bstnA(cfg)# end
bstnA# show capture

  capture
      clientCap.cap           Oct 27 14:32  15 MB
      nasTraffic.cap          Oct 27 14:37  6.0 MB
      proxyTraffic_01352_20091027183657.cap Oct 27 14:37  151 kB
      proxyTraffic_01353_20091027183700.cap Oct 27 14:37  42 kB

bstnA# capture merge file proxyTraffic

% INFO: Merging 2 capture files into proxyTraffic.cap; this may take up to 1 minute to complete.

bstnA# show capture


  capture
      clientCap.cap           Oct 27 14:32  15 MB
      nasTraffic.cap          Oct 27 14:37  6.0 MB
```

```
      proxyTraffic.cap        Oct 27 14:57  193 kB

bstnA# ...
```

Copying the Capture File(s) to a Remote Site

You can send all of your packet-capture files back to a remote site (or F5 Support) for a detailed analysis. On the remote machine, you can use the graphical Wireshark program or the CLI-based TShark program to view the capture file(s). The ARX formats its packet-capture files to be legible by any program that supports the Packet Capture (Pcap, or WinPcap) library.

You can use the collect command to collect all packet-capture files, along with other diagnostic information, and send them to a remote site through FTP, SCP, or e-mail. This is described in *Collecting Diagnostic Information*, on page 8-30. To collect only the packet-capture files, use the collect capture command (from *Collecting Partial Information*, on page 8-34).

Configuring Port Mirroring

You can also monitor all of the traffic on a specific port, using an external packet analyzer. *Port mirroring* captures all packets going through designated source interface(s) and copies them to the "MIRROR" interface. A network analyzer at the "MIRROR" interface can therefore see all traffic going through the source interface(s) in real time.

Important

Oversubscription of the destination monitor port during port mirroring can result in a service interruption. Proceed only upon advice from F5 Support.

Port mirroring is not supported on the ARX-500. The ARX-2000 and ARX-4000 support port mirroring, but neither has a separate MIRROR port; for these types of chassis, send the packets to another client/server interface (as described in the subsection below).

From cfg mode, use the monitor system command to monitor one source interface:

```
monitor system source-interface slot/port {rx | tx | both}
```

where

slot/port is the source port. Use the show interface summary command to find all ports configured on the ARX.

{rx | tx | both} chooses the direction of monitored packets. Choose **rx** for received (ingress) packets, **tx** for transmit (egress) packets, or **both**.

A notice appears, cautioning you against excessive traffic through the MIRROR port (similar to the **Important** note above). Enter **yes** to continue.

Important

Remember to shut off port mirroring after you have finished diagnosing your problem. Port mirroring slows the source interface(s) with numerous packet selection, duplication, and forwarding operations.

Monitor sessions do not persist across reboots.

Using an Alternative Destination Interface

You can optionally use another external interface as the destination instead of the "MIRROR" interface. This is your only option with the ARX-2000 and ARX-4000, which have no separate "MIRROR" interface. From cfg mode, use the destination-interface clause at the end of the monitor module command:

```
monitor module source-interface slot/port {rx | tx | both}
destination-interface slot/port
```

where

source-interface ***slot/port*** is the source port (for example, "source-interface 2/9" on an ARX-4000). Use the show interface summary command to find all ports configured on the ARX.

{rx | tx | both} chooses the direction of monitored packets. Choose **rx** for received (ingress) packets, **tx** for transmit (egress) packets, or **both**.

destination-interface ***slot/port*** is the destination port where the network analyzer is connected (for example, "destination-interface 1/6"). Choose a destination port with equal or greater bandwidth than the source port.

You can monitor one source port at a time with monitor module; use monitor system (above) to monitor multiple source ports. As with monitor system, a prompt warns you not to overwhelm the destination port; enter **yes** to continue.

For example, the following command sequence monitors packets on port 2/4, duplicating the packets for a network analyzer on port 2/6:

```
bstnA(cfg)# monitor module source-interface 2/4 both destination-interface 2/6
Warning: Oversubscription of the destination monitor port may result in a service interruption.
Proceed only upon advice from Technical Support.

Are you sure? [yes/no] yes
bstnA(cfg)# ...
```

Showing Active Port-Mirroring Sessions

Use the show monitor command to view the current state of any active monitor session:

```
show monitor
```

For example, the following command sequence shows one active monitor session:

```
bstnA(cfg)# show monitor

Monitor Session: System
Source : Slot 2 Port 4
Destination :Slot 2 Port 6
Mode: Both

bstnA(cfg)# ...
```

Shutting Down Port Mirroring

Remember to shut off port-mirroring sessions when you are not actively monitoring them. Port mirroring puts a strain on port performance. Use the no monitor command to shut down a monitor session:

```
no monitor {system | module}
```

{system | module} chooses the type of monitor session to shut down.

For example, the following command sequence shuts down a monitor module session:

```
bstnA(cfg)# no monitor module
bstnA(cfg)# ...
```

Shutting Down Mirroring for One Source Port

Recall that you can use monitor system to monitor more than one port. For the multi-port case, you can use no monitor system to selectively disable one source port:

`no monitor system source-interface` ***`slot/port`***

where ***slot/port*** is the source port to stop monitoring (for example, "source-interface 1/5").

For example, the following command sequence stops monitoring port 2/3:

```
bstnA(cfg)# no monitor system source-interface 2/3
bstnA(cfg)# ...
```

Showing Filer-Connection Statistics

To see cumulative statistics for filer connections, you can use the show statistics filer connections command:

```
show statistics filer connections
```

The output is a table with one row per filer. Each row shows the current connects and the maximum connections since the last reboot of the ARX. For example:

```
bstnA# show statistics filer connections

                                          Connections
Filer                                     Current     Max
----------------------------------------  ----------  ----------
das1                                      12          12
fs1                                       9           12
fs2                                       5           13
fs3                                       5           19
fs4                                       11          20
fs5                                       2           10
fs6                                       3           17
fs7                                       2           17
das2                                      0           0
das3                                      12          12
nas1                                      206         210
das7                                      12          12
das8                                      12          12
nas2                                      12          12
nas3                                      12          12
nas10                                     4           19
nas11                                     4           6
nasE1                                     14          16
smb1                                      0           0
```

Focusing On a Particular Filer

You can use the name of a particular filer at the end of the command to focus on the connection statistics for one filer. You can also specify a particular network processor:

```
show statistics filer connections filer-name [processor slot.proc]
```

where

filer-name (optional, 1-64 characters) identifies an external filer by the name defined on the ARX. For a complete list of filer names, use the show external-filer command (see *Listing External Filers*, on page 6-18 of the *ARX® CLI Storage-Management Guide*).

processor *slot.proc* (optional) focuses the output on a particular network processor. Use show processors for a complete list of processors (and their modules and slots) on the ARX.

The output is two tables, one for CIFS connections and one for NFS connections. The CIFS table breaks down the statistics between data-plane connections (directly from ARX clients) and control-plane connections

(which go through one or more processes on the ACM or ASM), and shows counters for client sessions within the data-plane connections. If you choose the **processor** option, the output is similar but with a more-limited scope.

For example, this command shows the connection statistics for the "fs2" filer:

```
bstnA# show statistics filer fs2 connections

  Filer fs2 (192.168.25.27)               All processors
  ----------------------------------  --------------------

  CIFS:

    Connection limit:                     none

    Current data connections:             0
    Current control connections:          5
    Max connections (data+control):       8
    Time of max connections:              11/16/2012 02:16:57 -0500

    Current data sessions:                0
    Max data sessions:                    0
    Time of max data sessions:            n/a

    Max sessions per data connection:     0
    Time of max sessions/connection:      n/a

  NFS:

    Current connections:                  0
    Max connections:                      0
    Time of max connections:              n/a

bstnA# ...
```

Clearing the Filer-Connection Statistics

You can use the clear statistics filer connections command to clear the filer-connection statistics from the ARX database. This reduces all of the "Max" statistics to the current counts. Rebooting the ARX has the same effect. You can clear these statistics from priv-exec mode:

clear statistics filer [*filer-name*] connections

> where ***filer-name*** (optional, 1-64 characters) identifies a single external filer. This clears only the connection statistics that pertain to a single filer. Use the filer name defined on the ARX. For a complete list of filer names, use the show external-filer command (see *Listing External Filers*, on page 6-18 of the *ARX® CLI Storage-Management Guide*).

The CLI prompts for confirmation before clearing any connection statistics; enter **yes** to proceed.

For example, this command sequence clears the connection statistics for the "fs2" filer, and then shows that the "Max" counts now match the "Current" counts:

```
bstnA# clear statistics filer fs2 connections
```

```
Clear the connection statistics on filer 'fs2'.

Are you sure? [yes/no] yes
bstnA# show statistics filer fs2 connections

  Filer fs2 (192.168.25.27)                All processors
  -----------------------------------  --------------------

  CIFS:

    Connection limit:                      none

    Current data connections:              0
    Current control connections:           5
    Max connections (data+control):        5
    Time of max connections:               11/16/2012 02:17:33 -0500

    Current data sessions:                 0
    Max data sessions:                     0
    Time of max data sessions:             n/a

    Max sessions per data connection:      0
    Time of max sessions/connection:       n/a

  NFS:

    Current connections:                   0
    Max connections:                       0
    Time of max connections:               n/a


bstnA#
```

Dropping All Connections to a Filer

If a back-end filer is overwhelmed with TCP connections, you can forcibly drop all connections to the filer at once. This can occur after you reduce the limit on CIFS connections to the filer (see *Limiting CIFS Connections to the Filer*, on page 6-10 of the *ARX® CLI Storage-Management Guide*); new clients cannot connect until the number of current connections drops below the new limit.

Dropping TCP connections may cause a noticeable interruption for your currently-connected clients. From priv-exec mode, you can use drop filer-connections to close all TCP sessions with a particular filer:

drop filer-connections *filer-name* [processor *slot.processor*]

where

filer-name (1-64 characters) identifies a single external filer. Use the filer name defined on the ARX. For a complete list of filer names, use the show external-filer command (see *Listing External Filers*, on page 6-18 of the *ARX® CLI Storage-Management Guide*).

slot.processor identifies a single NSM processor.

The CLI prompts for confirmation before dropping all connections; enter **yes** to proceed.

For example, this command sequence drops all connections to the “smb1” filer:

```
bstnA# drop filer-connections smb1

Drop the connections to filer smb1.

Proceed? [yes/no] yes
bstnA#
```

10

Troubleshooting Managed Volumes

Overview

This chapter describes how to

- show usage statistics for managed volumes,
- remove shares from them,
- correct share-import errors,
- cleanly remove a namespace and all of its associated configuration objects,
- manage file collisions after an import,
- fix file- and directory-naming issues in a CIFS namespace,
- fix file- and directory-naming issues in a multi-protocol namespace,
- find the exact location of a file on your back-end storage
- show policy usage and history, and
- address inconsistent directory attributes.

This extends the previous chapters, which described tools for collecting diagnostic information and troubleshooting network connections.

Showing Share Statistics

The ARX records read/write statistics, including latency measures, for each of its filer shares. It keeps these statistics from the moment each share is "enabled" as part of an ARX volume. (For instructions on configuring and enabling a volume share, refer to the *ARX® CLI Storage-Management Guide*: *Adding a Share*, on page 8-8 for direct volumes, or *Adding a Share*, on page 9-33 for managed volumes.) The statistics are kept at the NSM, which runs the ARX's *fastpath* processes. Use the show statistics namespace all fastpath command to see these statistics for all shares:

```
show statistics namespace all fastpath
```

For example, the following command shows usage statistics for all filer shares:

```
bstnA> show statistics namespace all fastpath

Fastpath summary statistics for shares by namespace and volume
collected at 11/16/2012 01:43:50 -0500

  Namespace: medco
  Volume: /vol
  All shares in volume

    Bytes read:                 55,799,614
    Read calls:                      4,186
    Bytes written:             138,547,158
    Write calls:                     8,537
    Other calls:                    76,410
    All calls:                      89,133
    Average latency:              3992 uSec

  Namespace: wwmed
  Volume: /acct
  All shares in volume

    Bytes read:                258,786,613
    Read calls:                     34,869
    Bytes written:             441,038,367
    Write calls:                    60,269
    Other calls:                   442,726
    All calls:                     537,864
    Average latency:               525 uSec

  Namespace: medarcv
  Volume: /rcrds
  All shares in volume

    Bytes read:                    513,948
    Read calls:                         16
    Bytes written:               1,764,404
    Write calls:                       112
    Other calls:                       152
    All calls:                         280
    Average latency:           169796 uSec

  Namespace: medarcv
  Volume: /lab_equipment
  All shares in volume
```

```
  Bytes read:                          0
  Read calls:                          0
  Bytes written:                       0
  Write calls:                         0
  Other calls:                         6
  All calls:                           6
  Average latency:           2787097 uSec

Namespace: medarcv
Volume: /test_results
All shares in volume

  Bytes read:                          0
  Read calls:                          0
  Bytes written:                       0
  Write calls:                         0
  Other calls:                         0
  All calls:                           0
  Average latency:                    --

Namespace: insur
Volume: /claims
All shares in volume

  Bytes read:                      4,790
  Read calls:                          2
  Bytes written:              11,922,796
  Write calls:                     1,458
  Other calls:                       185
  All calls:                       1,645
  Average latency:              1478 uSec

Namespace: labResearch
Volume: /labMice
All shares in volume

  Bytes read:                          0
  Read calls:                          0
  Bytes written:                       0
  Write calls:                         0
  Other calls:                         0
  All calls:                           0
  Average latency:                    --

Namespace: labResearch
Volume: /labRats
All shares in volume

  Bytes read:                          0
  Read calls:                          0
  Bytes written:                       0
  Write calls:                         0
  Other calls:                         0
  All calls:                           0
  Average latency:                    --


bstnA>
```

Showing Share Statistics from One Namespace

To focus on the share statistics from one namespace, use the namespace argument in the show statistics namespace command, followed by the fastpath keyword:

show statistics namespace *name* fastpath

where

name (1-30 characters) identifies the desired namespace, and

fastpath is a required keyword.

This displays summary statistics for each volume in the namespace.

For example, the following command shows share statistics for the "medarcv" namespace:

```
bstnA> show statistics namespace medarcv fastpath

Fastpath summary statistics for shares by namespace and volume
collected at 11/16/2012 01:43:43 -0500

  Namespace: medarcv
  Volume: /rcrds
  All shares in volume

    Bytes read:                    513,948
    Read calls:                         16
    Bytes written:              1,764,404
    Write calls:                       112
    Other calls:                       152
    All calls:                         280
    Average latency:           169796 uSec

  Namespace: medarcv
  Volume: /lab_equipment
  All shares in volume

    Bytes read:                          0
    Read calls:                          0
    Bytes written:                       0
    Write calls:                         0
    Other calls:                         6
    All calls:                           6
    Average latency:          2787097 uSec

  Namespace: medarcv
  Volume: /test_results
  All shares in volume

    Bytes read:                          0
    Read calls:                          0
    Bytes written:                       0
    Write calls:                         0
    Other calls:                         0
    All calls:                           0
    Average latency:                    --


bstnA>
```

Showing Statistics from One Volume

You can narrow the focus further to a particular volume. After the namespace name, add the volume path to the show statistics namespace command:

show statistics namespace *name* volume *vol-path* fastpath

where

name (1-30 characters) identifies the desired namespace,

vol-path (1-1024 characters) identifies the volume, and

fastpath is a required keyword.

For example, the following command shows share statistics from the "medarcv~/rcrds" volume:

```
bstnA> show statistics namespace medarcv volume /rcrds fastpath

Fastpath summary statistics for shares by namespace and volume
collected at 11/16/2012 01:43:43 -0500

  Namespace: medarcv
  Volume: /rcrds
  All shares in volume

    Bytes read:                    513,948
    Read calls:                         16
    Bytes written:              1,764,404
    Write calls:                       112
    Other calls:                       152
    All calls:                         280
    Average latency:           169796 uSec


bstnA>
```

Focusing on One Share

You can narrow the focus down to a single share in the volume. After the volume path, add the share name to the show statistics namespace command:

show statistics namespace *name* volume *vol-path* share *share-name* fastpath

where

name (1-30 characters) identifies the desired namespace,

vol-path (1-1024 characters) identifies the volume,

share-name (1-64 characters) identifies the specific share, and

fastpath is a required keyword.

For example, the following command shows share statistics from the "charts" share in the "medarcv~/rcrds" volume:

```
bstnA> show statistics namespace medarcv volume /rcrds share charts fastpath

Fastpath summary statistics for shares by namespace and volume
```

```
collected at 11/16/2012 01:43:43 -0500

  Namespace: medarcv
  Volume: /rcrds
  Share: charts
  Filer: fs1
  CIFS share: histories

    Bytes read:                     338,456
    Read calls:                          10
    Bytes written:                  338,456
    Write calls:                         22
    Other calls:                         68
    All calls:                          100
    Average latency:            124867 uSec

bstnA>
```

Showing Metadata Statistics

A managed volume stores metadata about the files that it contains at the back-end. In order for the managed volume to perform effectively, the latency between the managed volume software running on the ARX and the filer share on which the metadata resides must be low. The show statistics metadata command enables you to display a number of metadata read and write statistics for each managed volume associated with the ARX.

The command syntax is:

show statistics metadata [namespace *name* [volume *volumepath*]]

where *name* specifies a namespace for which to display metadata statistics, and *volumepath* species a volume for which to display metadata statistics.

An example of the output from this command follows:

```
bstnA# show statistics metadata

  Namespace: wwmed
  Volume:    /acct
  Location:  nas1 /vol/vol2/meta1

      Last Reset Time:              11/16/2012 00:52:17 -0500

      Read Operations:              20
      Read Response Time:           3.20 usec/op
      Read Bytes:                   60416
      Read Rate:                    944.00 MB/s
      Read Errors:                  0
      Read Operations In Progress:  0
      Last Read Operation:          11/16/2012 00:52:17 -0500

      Write Operations:             6955
      Write Response Time:          59.14 usec/op
      Write Bytes:                  6331392
      Write Rate:                   15.39 MB/s
      Write Errors:                 0
      Write Operations In Progress: 0
      Last Write Operation:         11/16/2012 01:36:44 -0500

  Namespace: medarcv
  Volume:    /lab_equipment
  Location:  nas1 /vol/vol2/meta6

      Last Reset Time:              11/16/2012 00:53:08 -0500

      Read Operations:              28
      Read Response Time:           2.79 usec/op
      Read Bytes:                   57344
      Read Rate:                    735.18 MB/s
      Read Errors:                  0
      Read Operations In Progress:  0
      Last Read Operation:          11/16/2012 01:07:33 -0500

      Write Operations:             73
      Write Response Time:          358.95 usec/op
      Write Bytes:                  256000
      Write Rate:                   9.77 MB/s
      Write Errors:                 0
      Write Operations In Progress: 0
```

```
    Last Write Operation:         11/16/2012 01:39:03 -0500

Namespace: medarcv
Volume:    /rcrds
Location:  nas1 /vol/vol2/meta3

    Last Reset Time:              11/16/2012 00:52:55 -0500

    Read Operations:              28
    Read Response Time:           2.89 usec/op
    Read Bytes:                   57344
    Read Rate:                    707.95 MB/s
    Read Errors:                  0
    Read Operations In Progress:  0
    Last Read Operation:          11/16/2012 01:12:32 -0500

    Write Operations:             124
    Write Response Time:          237.94 usec/op
    Write Bytes:                  407552
    Write Rate:                   13.81 MB/s
    Write Errors:                 0
    Write Operations In Progress: 0
    Last Write Operation:         11/16/2012 01:39:02 -0500

Namespace: insur
Volume:    /claims
Location:  nas1 /vol/vol2/meta2

    Last Reset Time:              11/16/2012 01:12:51 -0500

    Read Operations:              26
    Read Response Time:           3.46 usec/op
    Read Bytes:                   77824
    Read Rate:                    864.71 MB/s
    Read Errors:                  0
    Read Operations In Progress:  0
    Last Read Operation:          11/16/2012 01:17:30 -0500

    Write Operations:             86
    Write Response Time:          239.01 usec/op
    Write Bytes:                  330752
    Write Rate:                   16.09 MB/s
    Write Errors:                 0
    Write Operations In Progress: 0
    Last Write Operation:         11/16/2012 01:29:22 -0500

Namespace: labResearch
Volume:    /labMice
Location:  nas1 /vol/vol2/meta8

    Last Reset Time:              11/16/2012 01:17:18 -0500

    Read Operations:              18
    Read Response Time:           4.61 usec/op
    Read Bytes:                   52224
    Read Rate:                    629.20 MB/s
    Read Errors:                  0
    Read Operations In Progress:  0
    Last Read Operation:          11/16/2012 01:17:18 -0500

    Write Operations:             39
    Write Response Time:          252.51 usec/op
```

```
      Write Bytes:                  138240
      Write Rate:                   14.04 MB/s
      Write Errors:                 0
      Write Operations In Progress: 0
      Last Write Operation:         11/16/2012 01:18:17 -0500

  Namespace: labResearch
  Volume:    /labRats
  Location:  nas1 /vol/vol2/meta9

      Last Reset Time:              11/16/2012 01:17:25 -0500

      Read Operations:              18
      Read Response Time:           3.50 usec/op
      Read Bytes:                   52224
      Read Rate:                    828.95 MB/s
      Read Errors:                  0
      Read Operations In Progress:  0
      Last Read Operation:          11/16/2012 01:17:25 -0500

      Write Operations:             36
      Write Response Time:          273.81 usec/op
      Write Bytes:                  125952
      Write Rate:                   12.78 MB/s
      Write Errors:                 0
      Write Operations In Progress: 0
      Last Write Operation:         11/16/2012 01:18:18 -0500
bstnA#
```

Execute the command clear statistics metadata to clear and reset metadata counters for the entire system, or for a specified namespace or volume.

Draining One or More Shares

To remove a share from a managed volume with snapshots, you must drain it of all files, wait for all of its back-end snapshots to age out, and then remove the share from the ARX configuration. Volumes without snapshots can use a simpler method, described in the next section. Skip to the next section to remove a share from a managed volume without any snapshot support.

You can move all files from one share (or share farm) to one or more other shares in the same volume, thereby *draining* the source share(s). A placement rule accomplishes this, and prevents any new files from being created on the source share(s). A placement rule is typically used for moving files to their storage tiers, as described in *Placing Files on Particular Shares*, on page 14-4 of the *ARX® CLI Storage-Management Guide*. This is a special use case for a placement rule.

Use place-rule to create a new placement rule, as described in the storage manual. This puts you into gbl-ns-vol-plc mode, where you must choose a source share or share farm, set the storage target for the files, and enable the rule. There are also some optional commands you can invoke from this mode. This section explains the options that are relevant to draining a share.

For example, the following command sequence creates an empty placement rule, “emptyRH,” for the namespace, “wwmed:”

```
bstnA(gbl)# namespace wwmed volume /acct
bstnA(gbl-ns-vol[wwmed~/acct])# place-rule emptyRH

This will create a new policy object.

Create object 'emptyRH'? [yes/no] yes
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# ...
```

Identifying the Source Share(s)

The next step in draining a share or shares is to identify them. The placement rule places all files from the source share(s) onto the target storage. From gbl-ns-vol-plc mode, use the source command to specify the source share(s).

From gbl-ns-vol-plc mode, use one of two source commands to specify the source share(s):

`source share` ***`share-name`***

where **share** ***share-name*** (1-64 characters) identifies a single source share, or

`source share-farm` ***`share-farm-name`***

where ***share-farm-name*** (1-64 characters) is a group of shares in a share farm.

Use the show global-config namespace command to see the shares or share farms in each volume: see *Showing Namespace Configuration*, on page 7-26 of the *ARX® CLI Storage-Management Guide*.

For example, the following command set selects the "it5" share:

```
bstnA(gbl)# namespace wwmed volume /acct
bstnA(gbl-ns-vol[wwmed~/acct])# place-rule emptyRH
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# source share it5
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# ...
```

Choosing the Target Storage

The next step in draining shares is to choose the target storage for the share's files. You can choose one target: another share or share farm in the current volume. From gbl-ns-vol-plc mode, use the target rule to set the share's storage target.

For example, the following command sequence selects a share farm, "fm1," as the target for the "emptyRH" placement rule:

```
bstnA(gbl)# namespace wwmed volume /acct
bstnA(gbl-ns-vol[wwmed~/acct])# place-rule emptyRH
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# target share-farm fm1
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# ...
```

As another example, the following command sequence selects a share, "nas3," as the new home for all files in the source share:

```
bstnA(gbl)# namespace archives volume /etc
bstnA(gbl-ns-vol[archives~/etc])# place-rule move2nas
bstnA(gbl-ns-vol-plc[archives~/etc~move2nas])# target share nas3
bstnA(gbl-ns-vol-plc[archives~/etc~move2nas])# ...
```

Balancing Capacity in a Target Share Farm

When a share farm is a file-placement target, the first configured share in the farm is the default share for placed files. Most files are placed on the same share as their parent directory, but a file defaults to the first share if its parent directory is outside the share farm. The first share in the farm can therefore take a heavier file burden over time.

To migrate files off of any share that is running low on free space, you can configure *auto migration* for the share farm. Refer to *Auto Migrating Existing Files*, on page 15-15 of the *ARX® CLI Storage-Management Guide*.

Setting Other Placement-Rule Options

Other options for a file-placement rule may also be of use at your site, though they are not directly relevant to draining a share. These include:

- migrate close-file

  to close any files held open by CIFS clients, thereby making it possible to migrate them. For details, refer to *Automatically Closing All Open Files (CIFS)*, on page 14-14 of the *ARX® CLI Storage-Management Guide*.

- limit-migrate

  to control the bandwidth used for migrations. For details, refer to *Limiting Each Migration (optional)*, on page 14-17 of the same storage guide.

- report

  to create a progress report for the drain operation. This is recommended. For details, refer to *Configuring Progress Reports*, on page 14-22 of the storage guide.

For example, the following command sequence limits the migration bandwidth to 10G and sends the migration results to a report:

```
bstnA(gbl)# namespace wwmed volume /acct
bstnA(gbl-ns-vol[wwmed~/acct])# place-rule emptyRH
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# limit-migrate 10G
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# report emptyRH_ verbose
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# ...
```

Retaining Copies of Files on the Share (optional)

Before you enable the rule and start draining the share, you have the option to retain copies of those files. From gbl-ns-vol-shr mode, use the migrate retain-files command to put this safeguard in place:

```
migrate retain-files
```

The file copies go to a hidden directory, "~acopia_msnap," at the root of the share. To restore the files later, you can access the back-end share directly (it is not included in the volume).

Important

Do not use this command for a share in a share farm. If a share-farm-balancing policy tries to move files off of this share, this placement rule makes copies of them, thereby ensuring that the share's free space never actually changes. The balancing policy will therefore move files off of the share indefinitely, unnecessarily occupying space on other shares.

Use this only as a safeguard when draining all files from a single share.

For example, the following command sequence retains all migrated files in the it5 share:

```
bstnA(gbl)# namespace wwmed volume /acct share it5
bstnA(gbl-ns-vol-shr[wwmed~/acct~it5])# migrate retain-files
bstnA(gbl-ns-vol-shr[wwmed~/acct~it5])# ...
```

Disabling File Retention

To disable the feature that retains copies of migrated files, use the no form of the command:

```
no migrate retain-files
```

For example, the following command sequence disables the migration safeguard in the it5 share:

```
bstnA(gbl)# namespace wwmed volume /acct share it5
bstnA(gbl-ns-vol-shr[wwmed~/acct~it5])# no migrate retain-files
bstnA(gbl-ns-vol-shr[wwmed~/acct~it5])# ...
```

Making the Rule Tentative

A *tentative* rule is configured to appear in the system logs (syslog) as "tentative," showing the potential effects of the rule if it was enabled. (The log component, POLICY_ACTION, creates the syslog messages; syslog access and log components are described in *Accessing the Syslog*, on page 8-7.) If you configured reporting for the rule (as shown above), the report shows which files would be migrated if the rule was fully enabled. Tentative rules do not change policy enforcement, but they do consume processing time in the policy software. From gbl-ns-vol-plc mode, use the tentative command to put the placement rule in a tentative state:

```
tentative
```

To see the simulated rule run, you must enable the rule (as described below). Then use show logs syslog or grep *pattern* logs syslog to view the syslog.

For example, the following command sequence makes the "emptyRH" rule tentative:

```
bstnA(gbl)# namespace wwmed volume /acct
bstnA(gbl-ns-vol[wwmed~/acct])# place-rule emptyRH
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# tentative
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# ...
```

Removing the Tentative State

Use the no tentative command to fully activate the drain rule. You can then disable and re-enable the rule (as described below) to start the draining process. For example, the following command sequence activates the "emptyRH" rule. When the rule is enabled, it performs actual migrations:

```
bstnA(gbl)# namespace wwmed volume /acct
bstnA(gbl-ns-vol[wwmed~/acct])# place-rule emptyRH
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# no tentative
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# ...
```

Enabling the Drain Rule

The final step in draining a share is to enable its placement rule. By default, the rule is disabled and ignored by policy software. Use the enable command to enable the rule. For example, the following command sequence enables the "emptyRH" rule:

```
bstnA(gbl)# namespace wwmed volume /acct
bstnA(gbl-ns-vol[wwmed~/acct])# place-rule emptyRH
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# enable
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# ...
```

Disabling the Rule

Disabling the rule removes it from consideration. Use no enable from gbl-ns-vol-plc mode to disable a placement rule. For example, the following command sequence disables the "emptyRH" rule:

```
bstnA(gbl)# namespace wwmed volume /acct
bstnA(gbl-ns-vol[wwmed~/acct])# place-rule emptyRH
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# no enable
bstnA(gbl-ns-vol-plc[wwmed~/acct~emptyRH])# ...
```

Verifying That All Files Are Removed

You can use a metadata-only report to verify that the share is empty. Use the nsck ... metadata-only command to generate a report about the share (as described in *Focusing on One Share*, on page 7-10), and then use show reports to view the report. For example, the following command sequence generates a report on the "it5" share, proving that it has no files:

```
bstnA(gbl)# nsck wwmed report metadata-only share it5
Scheduling report: metadata_only.10.rpt on switch bstnA
bstnA(gbl)# show reports metadata_only.10.rpt
**** Metadata-Only Report: Started at Fri Nov 13 01:28:37 2009 ****
**** Software Version: 5.01.000.11919 (Nov  6 2009 22:21:42) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:
**** Namespace: wwmed
**** Volume: /acct
**** Path: /acct
**** Share: it5

Share                Physical Filer
-------------------  ------------------------------------------------------------
[it5              ]  192.168.25.24:/lhome/it5

**** Legend:
****   FL = File: The reported entry is a file.
****   DR = Directory: The reported entry is a directory.
****   SL = Symlink: The reported entry is a symbolic link.
****   LN = Link: The reported entry has a link count greater than one.
****   NL = No Lock: Was unable to lock parent directory during report.
****   CC = NFS case-blind name collision.
****   IC = Name contains invalid CIFS characters.
****   FN = Name may conflict with a filer-generated name.
****   SP = A persistent split is registered in the metadata, due to a FGN.
****   NF = Name is only accessible to NFS clients.

Type                  Share                Path
--------------------  -------------------  ------------------------------------
[   DR             ]  [it5              ]  /docs
[   DR             ]  [it5              ]  /docs/masterIndex
[   DR             ]  [it5              ]  /docs/HW_InstallA5C
[   DR             ]  [it5              ]  /docs/common
[   DR             ]  [it5              ]  /docs/SLM_Beta
[   DR             ]  [it5              ]  /docs/snmpRef
[   DR             ]  [it5              ]  /docs/license
[   DR             ]  [it5              ]  /docs/HW_InstallA1K
[   DR             ]  [it5              ]  /docs/GUIHelp
[   DR             ]  [it5              ]  /docs/cliReference
[   DR             ]  [it5              ]  /docs/SecureAgent
```

```
[   DR              ]  [it5              ]  /docs/logCatalog
[   DR              ]  [it5              ]  /docs/ReleaseNotes
[   DR              ]  [it5              ]  /docs/useCases
[   DR              ]  [it5              ]  /docs/Glossary
[   DR              ]  [it5              ]  /docs/architecture
[   DR              ]  [it5              ]  /docs/Quickstart_CardA1K
[   DR              ]  [it5              ]  /docs/cliOperator
[   DR              ]  [it5              ]  /docs/GUI
[   DR              ]  [it5              ]  /docs/sitePlanning
[   DR              ]  [it5              ]  /docs/HW_Install
[   DR              ]  [it5              ]  /docs/Quickstart_Card
[   DR              ]  [it5              ]  /docs/done_cli_cmds
[   DR              ]  [it5              ]  /docs/done_ref_
[   DR              ]  [it5              ]  /docs/GUIHelp/Temp
[   DR              ]  [it5              ]  /docs/GUIHelp/Output
[   DR              ]  [it5              ]  /docs/GUIHelp/Support
[   DR              ]  [it5              ]  /docs/Glossary/Frame_files
[   DR              ]  [it5              ]  /docs/HW_Install/HW_Book
[   DR              ]  [it5              ]  /docs/HW_InstallA1K/HW_Book
[   DR              ]  [it5              ]  /docs/HW_InstallA5C/HW_Book
[   DR              ]  [it5              ]  /docs/SecureAgent/Framefiles
[   DR              ]  [it5              ]  /docs/GUIHelp/Output/images
[   DR              ]  [it5              ]  /docs/GUIHelp/Support/images
[   DR              ]  [it5              ]  /docs/useCases/multiTier
[   DR              ]  [it5              ]  /docs/useCases/specialCases
[   DR              ]  [it5              ]  /docs/useCases/singleTier
[   DR              ]  [it5              ]  /docs/useCases/network

**** Total Files:                          0
**** Total Directories:                   38
**** Total Hard Links (nlink>1):           0
**** Total Symlinks:                       0
**** Total Locking Errors:                 0

**** Total items:                 38
**** Elapsed time:           00:00:01
**** Metadata-Only Report: DONE at Fri Nov 13 01:28:38 2009 ****
```

Removing the Drain Rule

You can remove a rule to both disable it and delete its configuration. From gbl-ns-vol mode, use the no form of the place-rule command to remove a placement/drain rule. For example, the following command sequence removes the drain rule, “drainSrvW08,” from the “medarcv~/rcrds” volume:

```
bstnA(gbl)# namespace medarcv volume /rcrds
bstnA(gbl-ns-vol[medarcv~/rcrds])# no place-rule drainSrvW08
bstnA(gbl-ns-vol[medarcv~/rcrds])# ...
```

Removing an Imported Share

You can remove a share from its managed volume without affecting service to the volume's clients. The remove-share migrate command migrates all of the share's files and master directories to another share or share farm in the same volume. (A *master directory* is the first-imported (or only) instance of a directory in the managed volume.)

Important

This is not recommended in a volume that supports snapshots. In those volumes, you should drain the share first, wait until all retained snapshots have "aged out," and then run this command. For instructions on draining all files from a share, recall Draining One or More Shares, on page 10-12, above.

Before you remove the share, you must first verify that it is not referenced by any policies or rules; look for the share name in the volume's rules (use the show policy namespace ... volume command, as shown in *Focusing on One Volume*, on page 14-41 of the *ARX® CLI Storage-Management Guide*). Also, the share must be online in the namespace: use show namespace status to confirm this (see *Monitoring the Import*, on page 9-56 of the same manual).

To retain copies of all migrated files, you have the option to use migrate retain-files in the share before you remove it (recall *Retaining Copies of Files on the Share (optional)*, on page 10-14).

From priv-exec mode, use remove-share migrate to remove the share:

```
remove-share migrate namespace volume share destination [close-file
[exclude fileset]] [async]
```

where:

namespace (1-30 characters),

volume (1-1024 characters), and

share (1-64 characters) all identify the share to remove, and

destination (1-64 characters) is the destination share or share farm. This command migrates all of the ***share***'s files and master directories to the ***destination***. This must be a share or share farm in the same ***volume***.

close-file (optional) causes the operation to close all files open by CIFS clients in order to migrate them. If you omit this option, files opened by CIFS clients cannot migrate off of the selected share. This would leave files stranded on the share and cause the share removal to fail.

exclude *fileset* (optional if you choose close-file, 1-64 characters) is a fileset to exclude from automatic closure. If a file in this fileset is open through CIFS, the rule places it on a retry queue instead of

automatically closing it. If such a file remains open for the duration of the share removal, it never migrates off of the share and the share-removal fails.

async (optional if you specify a destination) makes this command return immediately, rather than waiting for the share removal to finish.

The CLI prompts you with a warning before removing the share; enter **yes** to proceed. The CLI blocks as the volume migrates the files and directories off of the share, then prints a message indicating the success of the share removal.

◆ Note

If the removal is interrupted by a reboot and/or failover, it may be incomplete after the switch(es) finish booting. You can re-run the command to finish removing the share.

For example, the following command removes the '/acct~it5' share from the 'wwmed' namespace, and migrates its files and master directories to the 'fm1' share farm.

```
bstnA(gbl)# end
bstnA# remove-share migrate wwmed /acct it5 fm1

WARNING !! Share 'it5' will be removed from volume '/acct' in
namespace 'wwmed' after migrating files to 'fm1'
Proceed? [yes/no] yes
bstnA# ...
```

Removing the Empty Directory Tree

When you remove a share from a namespace, it leaves its empty directory tree on the share's back-end filer. After removing the share, you can access the filer directly and remove the empty directory tree.

Extra Processing in a Multi-Protocol Volume

If the share is in a multi-protocol (NFS and CIFS) volume, the volume may require additional processing for the migration. The extra processing guards against a file or directory on the source share colliding with a filer-generated name (*FGN*) at the destination. The volume does not record filer-side FGNs in its metadata; it deduces when collisions are possible and then uses probes to check for them.

Migrating NFS-Only Directories

Any file or directory with a back-end FGN is labeled *NFS-only*; CIFS clients cannot access it through the volume, and the ARX software cannot reliably find its CIFS-side FGN. The CIFS attributes for NFS-only files and directories are therefore inaccessible to the ARX. This is especially

important for directories. If the volume cannot find a directory's CIFS attributes, it cannot make a *precise* duplicate of the directory. The migrations for NFS-only directories therefore fail.

You can work around this in one of two ways: rename all NFS-only directories so that they are accessible through CIFS, or remove an attribute-consistency check at the destination share. The first option is recommended, but not always possible. The second option causes an undesirable side-effect; it reduces any unknown CIFS attributes to zero. To remove the attribute-consistency check, go to gbl-ns-vol-shr mode for the destination share and use the no strict-attribute-consistency command:

```
no strict-attribute-consistency
```

Important

This removes the CIFS attributes from NFS-only directories that migrate to the share. We recommend that you use this feature only under the guidance of F5 personnel.

For example, the following command sequence turns off strict-attribute consistency in the 'shr1-next' share:

```
bstnA(gbl)# namespace insur
bstnA(gbl-ns[insur])# volume /claims
bstnA(gbl-ns-vol[insur~/claims])# share shr1-next
bstnA(gbl-ns-vol-shr[insur~/claims~shr1-next])# no strict-attribute-consistency
bstnA(gbl-ns-vol-shr[insur~/claims~shr1-next])# ...
```

Forcing Strict-Attribute Consistency

Strict-attribute consistency is recommended for most situations. Typically, you restore this consistency check after the migration is finished. From gbl-ns-vol-shr mode, use strict-attribute-consistency to enable the consistency checks:

```
strict-attribute-consistency
```

For example, the following command sequence re-instates strict-attribute consistency in the 'shr1-next' share:

```
bstnA(gbl)# namespace insur
bstnA(gbl-ns[insur])# volume /claims
bstnA(gbl-ns-vol[insur~/claims])# share shr1-next
bstnA(gbl-ns-vol-shr[insur~/claims~shr1-next])# strict-attribute-consistency
bstnA(gbl-ns-vol-shr[insur~/claims~shr1-next])# ...
```

Finding Reports About the Share Removal

Each remove-share operation creates two reports as it runs:

- drain_share_rule_for_share_*share-name_time*.rpt, and
- remove.*job-id*.*share-name*.*share-id*.rpt

These describe two of the phases in removing a share. The first phase uses a placement rule to drain all files and directories from the share (see *Draining One or More Shares*, on page 10-12). The second phase removes the share from the namespace.

Use show reports type Plc to list all file-placement (and drain-share) reports. To list all of the remove-share reports, use show reports type Rm. For example, this shows the two reports (highlighted in bold text) from removing the it5 share:

```
bstnA# show reports type Plc

  reports
    Codes: Plc=Place Rule
      daily_archive_201002240100.rpt Feb 24 01:00  2.3 kB        Plc  DONE: 0 in 00:00:10
      daily_archive_201002240156.rpt Feb 24 01:56  2.3 kB        Plc  DONE: 0 in 00:00:11
      daily_archive_201002240205.rpt Feb 24 02:05  2.3 kB        Plc  DONE: 163 in 00:00:13
      docsPlc_20100224005857.0.rpt Feb 24 00:58  2.1 kB        Plc  FAILED: 0 in 00:00:01
      docsPlc_20100224005857.rpt Feb 24 00:58  2.0 kB        Plc  FAILED: 0 in 00:00:01
      drain_share_rule_for_share_it5_201002240213.rpt Feb 24 02:21  2.3 kB        Plc  DONE: 44 in
00:07:34
      drain_share_rule_for_share_shr1-old_201002240213.rpt Feb 24 02:13  2.3 kB        Plc  DONE:
159 in 00:00:31
      mvDats_201002240240.rpt Feb 24 02:41  2.3 kB        Plc  DONE: 0 in 00:00:10

bstnA# show reports type Rm


  reports
    Codes: Rm=Remove
      remove.19.shr1-old.26.rpt Feb 24 02:14  1.3 kB        Rm   DONE: 42 in 00:00:01
      remove.23.it5.8.rpt     Feb 24 02:21  1.3 kB        Rm   DONE: 942 in 00:00:01

bstnA# ...
```

Showing the Drain-Share Report

The drain-share report shows the number of files migrated off of the share. Use the show reports command to show it. For example, this shows the numbers of files migrated from the it5 share:

```
bstnA# show reports drain_share_rule_for_share_it5_201002240213.rpt
**** File Placement Report: Started at Wed Feb 24 02:13:27 2010 ****
**** Software Version: 5.02.000.12538 (Feb 16 2010 20:14:13) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:

  Place Rule:       drain_share_rule_for_share_it5

    Configuration:
      Namespace:                                wwmed
      Volume Name:                              /acct
      Source share:                             it5
      Target share-farm:                        fm1
      Report:                                   drain_share_rule_for_share_it5
      Report Verbose:                           Disabled
      Report Delete Empty:                      Disabled
      Report Errors Only:                       Disabled
      Migrate limit:                            0
      Volume Scan:                              Enabled
      Inline Notifications:                     Enabled
```

```
      Promote Directories:                          Enabled
      Auto-Close Files:                             Disabled

      Tentative:                                    No
      State:                                        Enabled

    Scan Statistics:
      Scan Started:                                 Wed Feb 24 02:13:43 2010
      Scan Completed:                               Wed Feb 24 02:21:01 2010
      Elapsed Time:                                 00:07:18
      Number of Times Paused:                       0
      Total Time Paused:                            00:00:00
      Number of Times Stopped by Low Space:         0
      Time Waiting for Free Space:                  00:00:00
      Files Scanned:                                6
      Directories Scanned:                          39
      Files in Fileset:                             6
      Files Migrated:                               6
      Size of Files Migrated:                       68 kB
      Directories Promoted:                         38
      Failed Migrations:                            0
      Size of Failed Migrations:                    0 B
      Failed Directory Promotes:                    0
      Files Forced Closed:                          0
      Suppressed Error Messages:                    0

Total processed:              44
Elapsed time:           00:07:34
**** File Placement Report: DONE at Wed Feb 24 02:21:01 2010 ****
bstnA# ...
```

Showing the Remove Report

The remove report shows the details of the share removal from the volume. For example, this shows the removal of the it5 share:

```
bstnA# show reports remove.23.it5.8.rpt
**** Storage Remove Report: Started at Wed Feb 24 02:21:03 2010 ****
**** Software Version: 5.02.000.12538 (Feb 16 2010 20:14:13) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:

**** Namespace:       wwmed
**** Volume:          /acct
**** Source Share:    it5
**** Source IP:       192.168.25.24
**** Source Export:   /lhome/it5
**** Relocate Share:  bills
**** Relocate IP:     192.168.25.25
**** Relocate Export: /work1/accting
**** Options:         remove-file-entries verbose

**** Legend:
****   !! = Error occurred during storage removal.
****   AR = Directory needs manual attribute recovery by user.
****   FF = Found file during remove without remove-file-entries setting.
****   RF = Removed file from metadata with remove-file-entries verbose setting.

Prepare Phase:
============================================================
```

```
  ------------------------------------------------------------
  Prepare Start Time:        Wed Feb 24 02:21:03 2010
  ------------------------------------------------------------

  Number of Keys Evaluated:                471
  Number of Directories Evaluated:         471
  Number of Migrations Aborted:              0
  Number of Directories Promoted:            0
  Number of Directories Migrated:            0
  Number of Directories Recovered:           0

  ------------------------------------------------------------
  Prepare Stop Time:         Wed Feb 24 02:21:03 2010
  Prepare Elapsed Time:      00:00:00
  ------------------------------------------------------------

Commit Phase:
============================================================

  ------------------------------------------------------------
  Commit Start Time:         Wed Feb 24 02:21:03 2010
  ------------------------------------------------------------

  Number of Directories Evaluated:         471
  Number of Files Removed:                   0

  ------------------------------------------------------------
  Commit Stop Time:          Wed Feb 24 02:21:03 2010
  Commit Elapsed Time:       00:00:00
  ------------------------------------------------------------

**** Elapsed time:           00:00:01
**** Storage Remove Report: DONE at Wed Feb 24 02:21:04 2010 ****
bstnA# ...
```

Removing the Share Asynchronously

By default, the CLI waits for the share removal to finish before allowing you to enter any more commands. Add the async flag at the end of the command to return to the CLI immediately after confirming the removal:

remove-share migrate *namespace volume share destination* async

This produces a brief report that summarizes the results, in addition to the more-extensive report described above. The CLI shows the name of the report after you enter **yes** to proceed.

For example:

```
bstnA(gbl)# end
bstnA# remove-share migrate wwmed /acct it5 fm1 async

WARNING !! Share 'it5' will be removed from volume '/acct' in
namespace 'wwmed' after migrating files to 'fm1'
Proceed? [yes/no] yes
Scheduling report: removeShare.it5.rpt

bstnA# ...
```

Removing Shares Without File Migration

You can also remove a share without migrating its files. This may have an impact on clients; any of the share's files that are visible to them will appear to vanish. This is typically reserved for share-import failures (described later in the chapter), or situations where a major change has been announced to the client base. As with the remove-share migrate command, this command migrates the share's master directories.

From priv-exec mode, use the remove-share nomigrate command to remove a share from a namespace without migrating any of its files:

remove-share nomigrate *namespace volume share destination* [async]

where all of the options are described above. The ***destination*** is required for the share's master directories, which must migrate before the share is removed. In multi-protocol volumes, master directories with NFS-only names cannot migrate to the destination share; resolve this as discussed above, in *Migrating NFS-Only Directories*, on page 10-19.

The CLI prompts for confirmation; enter **yes** to continue. The back-end share does not lose any of its files or directories when you use this command. You can access the share directly when the share-removal completes.

For example, the following command sequence removes the '/rcrds~oldrx' share from the 'medarcv' namespace. All of the share's master directories migrate to the 'rx' share:

```
bstnA(gbl)# end
bstnA# remove-share nomigrate medarcv /rcrds oldrx rx async

WARNING !! Share 'oldrx' will be removed from volume '/rcrds' in
namespace 'medarcv' WITHOUT migrating files.
Proceed? [yes/no] yes
Scheduling report: removeShare.oldrx.rpt

bstnA# ...
```

Forcing the Share Removal for Offline Filers

A online-share removal requires a scan of the filer. The scan finds the file attributes (such as ownership and access permissions) for all master directories. In cases where the filer or back-end share is offline, this scan fails and the share cannot be removed. To skip the scan and force the removal, use the force option at the end of the remove-share nomigrate command:

remove-share nomigrate *namespace volume share destination* [async]

where the options are as described above.

Since the volume cannot scan the file attributes for the master directories, it sets all file attributes to 0 (zero) for the relocated master directories. You must manually correct these attributes by accessing the volume through its virtual server's VIP. To identify the directories that need correction, consult

the remove report (described above; see *Showing the Remove Report*, on page 10-22). Each directory that requires attribute resets has an [AR] next to it in the report.

The CLI prompts you with a warning before removing the share; enter **yes** to proceed. The CLI blocks as the volume removes the share, then prints a message indicating the success of the share removal.

For example, the following command sequence removes a share, "ns~/~defunctShr:"

```
bstnA(gbl)# end
bstnA# remove-share nomigrate ns / defunctShr rx async

WARNING !! Share 'defunctShr' will be removed from volume '/' in
namespace 'ns' WITHOUT migrating files.
Proceed? [yes/no] yes
Scheduling report: removeShare.defunctShr.rpt

bstnA# ...
```

Canceling a Share Removal

Before the share removal begins, the volume scans the back-end filer. This scan collects all attributes for all of the share's master directories, to be duplicated at the destination share. It is possible to cancel the share removal during this scan, which takes longer than the other removal phases. From priv-exec mode, use the cancel remove command to cancel a share removal:

cancel remove namespace *ns* volume *vol-path* share *share-name*

where:

ns (1-30 characters) identifies the namespace.

vol-path (1-1024 characters) is the share's volume.

share-name (1-64 characters) is the share.

For example, the following command sequence stops the removal of the "wwmed~/acct~expir" share:

```
bstnA(gbl)# end
bstnA# cancel remove namespace wwmed volume /acct share expir
Storage job cancelled successfully.
bstnA# ...
```

Removing An Offline Share

You can remove an offline, unreachable, or otherwise defunct back-end share from a namespace by using the remove-share offline CLI command in privileged-exec mode. Use this command only for a back-end share that is unreachable on a metadata-based managed volume, not on a direct volume. If the share is still reachable, use *remove-share migrate* or *remove-share nomigrate* instead.

When executed, the command creates a report, "remove.*job-id*.*share-name*.*share-id*.rpt." Use *show reports* to view its contents.

The remove-share offline command deletes all of the share's files from the volume's metadata; from the client's perspective, these files disappear. Given that the back-end filer was unreachable to begin with, those files were inaccessible to clients already. Before removing the share, resolve all dependencies on the share. Use the *show policy* command to verify that no policy rules reference the share.

The command syntax follows:

```
remove-share offline namespace volume share dest [async]
```

where:

- ***namespace*** (1 - 30 characters) is the name of the namespace.
- ***volume*** (1 - 1024 characters) is the name of the volume.
- ***share*** (1 - 64 characters) is the name of the share.
- ***dest*** (1 - 64 characters) is the new destination share or share farm where the original share's master directories (if any) are to be migrated. A *master directory* is the first-imported instance of a directory in the managed volume. Use *show global-config* namespace for a list of shares and share farms in the namespace.
- **async** (optional) makes this command return immediately, rather than waiting for the share removal to finish.

Correcting Share-Import Errors

After you enable a namespace volume, the volume reads all of its back-end shares and *imports* all of their files and directories. If anything goes wrong in any share, the failed share shows a Status of "Error" in the show namespace status output, and one of the errors in *Table 10.1* appears in the more-verbose show namespace output. Use show namespace (not show namespace status) to see the full error message.

Follow the suggestions in the table to find and fix the root cause of the error. Once the error is corrected, retry the import.

The sections below the table describe various methods for either restarting the import or removing the share.

Status Condition	Description/Action
`Error: A directory was removed from the filer during import.`	A directory was found during an early scan of the filer share, then later it was missing. Either a client or client application must have accessed the filer directly (that is, not through this managed volume) during the import. This ARX cannot support unexpected changes to the back-end shares during (or after) import. After you correct the problem, use *nsck ... rebuild* volume to reimport all shares in the volume.
`Error: A filename/directory could not be mapped to the namespace character encoding.`	One of the share's directories has a name with Unicode characters that are unsupported by the *character-encoding nfs* setting. CIFS file names are Unicode and can contain any character, but NFS servers and clients must each configure their character encoding for file names. The volume cannot import a directory with any un-mappable characters in its name. You can use the *import rename-directories* unmapped-unicode command to allow the volume to rename such directories during import, or you can rename them manually at the filer. Then restart the share import: enter gbl-ns-vol-shr mode and re use the *enable (gbl-ns-vol-shr)* command.
`Error: a higher priority share failed to initialize import.`	A share with a higher *import priority* has failed its import, so this share cannot import. If any share import fails, the managed volume cannot import any shares with lower import priorities. Find the import error for the failed share(s), look for the error in this table, and take action as directed. This error is resolved as soon as all higher-priority shares successfully import.

Table 10.1 *Share Status Conditions*

Status Condition	Description/Action
`Error: Access denied by filer.`	This indicates that the ARX does not have proper permissions to connect to the back-end filer, or to perform some operation in the filer share. For NFS exports, check your back-end filer configuration: the back-end share should allow *root* access to all of the ARX's proxy IP addresses. Use the *show exports* command examine all permission settings at the filer. Use the *show ip proxy-addresses* command to list all configured proxy IP addresses. For CIFS shares, the switch uses the namespace's *proxy user* (username and password). The proxy-user credentials must belong to the Administrators group at every filer behind the namespace. Use the *probe exports* command to check this. The *proxy-user (gbl-ns)* command sets the proxy user credentials for a namespace. This may also indicate that the ARX is examining a read-only directory, such as the "~snapshot" directory in a NetApp filer. Such directories should be ignored during import. The gbl-filer *ignore-name* command identifies these types of directories. After you find and fix the error, you can restart the share import: enter gbl-ns-vol-shr mode for this share and re use the *enable (gbl-ns-vol-shr)* command.
`Error: Another share in the volume failed to import.`	All shares in the volume must import successfully for the volume to come online. Address the share that does not have this error, then restart this share import: enter gbl-ns-vol-shr mode for this share and re use the *enable (gbl-ns-vol-shr)* command.
`Error: Attempted to import subdirectory which is also a managed root`	This share is a parent to an already-imported share. Namespace shares cannot overlap. Use the *filer* command to change the path or share name. After you correct the problem, use *nsck ... rebuild* volume to reimport all shares in the volume.
`Error: Attributes of the share root are inconsistent.`	The share's root directory has attributes (such as owner, group, and permission settings) that are inconsistent with those of the already-imported shares. You can access the back-end filer directly to change these attributes, or you can use the *import sync-attributes* command to allow the volume to change the attributes for you. Then re-enable the share (*enable (gbl-ns-vol-shr)*) to restart its import.
`Error: Authentication error during import.`	The ARX does not have proper permissions to connect to the back-end filer. For NFS exports, check your back-end filer configuration: the back-end share should allow *root* access to all of the ARX's proxy IP addresses. Use the *show exports* command examine all permission settings at the filer. Use the *show ip proxy-addresses* command to list all configured proxy IP addresses. For CIFS shares, the switch uses the namespace's *proxy user* (username and password). The proxy-user credentials must belong to the Administrators group at every filer behind the namespace. The *proxy-user (gbl-ns)* command sets the proxy user credentials for a namespace. After you find and fix this issue, you can restart the share import: enter gbl-ns-vol-shr mode for this share and re use the *enable (gbl-ns-vol-shr)* command.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Error: Bad Export Path/Share Name`	The share cannot be found on the external filer. Use the *filer* command to change the path or share name for this share, then re-enable the share (*enable (gbl-ns-vol-shr)*) to retry the import.
`Error: CIFS ABE check failed.`	(CIFS) The volume supports *cifs access-based-enum* (ABE), and attempted to replicate ABE settings between its back-end shares. This replication process failed. The same process also checks for CIFS subshares (*filer-subshares*), so you can use *sync subshares from-namespace* ... tentative to get a full report on this issue. This often occurs because the ARX does not have proper permissions to check for ABE support on this back-end share. The ARX uses the namespace's *proxy user* (username and password) as its identity when it checks for ABE support. The proxy-user credentials must belong to the Administrators group at this back-end filer. You can use the *proxy-user (gbl-ns)* command to choose new proxy user credentials for the namespace. After you find and fix this issue, use *nsck ... rebuild* volume to reimport all shares in the volume.
`Error: CIFS error during import scan.`	(CIFS) This is a CIFS error that is not an access or network error, but prevented the import. The syslog shows the specific error. Use *show logs* syslog to read the syslog, or *grep string* logs syslog to search for a specific string in the syslog. After you correct the error, re-enable the share (*enable (gbl-ns-vol-shr)*) to retry the import.
`Error: CIFS filer connection limit reached.`	(CIFS) The back-end filer imposes a limit on CIFS connections, and the import requires more connections than the filer allows. Correct this at the back-end filer and then re-enable the share to retry the import.
`Error: CIFS operation attempted on a symbolic link.`	(multi-protocol) The volume software encountered an NFS symbolic link on this back-end share, and the volume has *cifs deny-symlinks* enabled. You can resolve this issue by using the no *cifs deny-symlinks* command to allow the volume software to follow these links. Alternatively, you can remove all NFS symbolic links from the back-end share. After you fix this issue, use *nsck ... rebuild* volume to reimport all shares in the volume.
`Error: CIFS operation failed with an error during import.`	(CIFS) The back-end filer returned an unexpected CIFS error during import. The syslog shows the specific error. Use *show logs* syslog to read the syslog, or *grep string* logs syslog to search for a specific string in the syslog. You may need to escalate to F5 Support. After you correct the error, re-enable the share (*enable (gbl-ns-vol-shr)*) to retry the import.
`Error: CIFS operation failed with an error indicating a filer fault.`	(CIFS) The filer returned an unexpected error during the import, and the error indicates a problem at the filer itself. The syslog shows the specific error. (Use *show logs* syslog to read the syslog, or *grep string* logs syslog to search for a specific string in the syslog.) Check the filer itself and correct the problem there. After you correct the error, re-enable the share (*enable (gbl-ns-vol-shr)*) to retry the import.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Error: CIFS operation returned an error indicating an unsupported option.`	(CIFS) The back-end share returned an error indicating that it does not support a CIFS option that the ARX requires. Consult the *F5 Data Solutions Compatibility Matrix* (included in this doc set) to confirm that the filer has been qualified for use behind the ARX. If the share cannot possibly support CIFS behind an ARX, you can use no *share* to remove the share from the volume. After you fix this issue, use *nsck ... rebuild* volume to reimport all shares in the volume.
`Error: CIFS proxy-user not privileged.`	(CIFS) The namespace software attempted to write a test file to the share and failed. Go to the filer and check permissions for the namespace's *proxy-user (gbl-ns)*; the proxy user must be part of the Backup Operators and/or Administrators group on the filer. After you correct the error, re-enable the share (*enable (gbl-ns-vol-shr)*) to retry the import.
`Error: CIFS subshare check failed.`	(CIFS) The volume supports *filer-subshares* and/or *cifs access-based-enum* (ABE), and attempted to replicate subshares, subshare ACLs, and/or ABE settings between its back-end shares. This replication process, also known as *subshare synchronization*, failed. As a result, any front-end export of the failed subshare will be degraded. The output of *show cifs-service fqdn* shows all of the degraded subshares in a given *fqdn* service. Use *sync subshares from-namespace* ... tentative to get a full report on this issue. To repair it, use the *sync subshares from-namespace* or *sync subshares from-service* command without the tentative option.
`Error: Cannot find credentials for connection to filer.`	(CIFS) Proxy user credentials have not been properly configured for the CIFS share. These credentials must be a username and password from the Administrators group. Use the *proxy-user* command to add or edit these credentials, and use the *proxy-user (gbl-ns)* command to apply them to a namespace. After you correct the error, re-enable the share (*enable (gbl-ns-vol-shr)*) to retry the import.
`Error: Collision rename failed due to open file on share.`	(CIFS) The share had a file that collided with an already-imported file, and the volume failed to rename it because a CIFS application has it open and locked. You can use *show cifs-service open-files* to find the open file, *close cifs file* to close it, and then retry the share import (with *enable (gbl-ns-vol-shr)*).

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Error: Could not update attributes on a file/directory during import.`	Failure to update file attributes can be caused by loss of connectivity during the import. Use the *show exports* command and/or *ping* to check the connection to the filer. For CIFS shares, this may indicate permissions problems. Proxy-user credentials may not have been properly configured for the CIFS share. These credentials must be a username and password from the Administrators group. Special, immutable directories can also cause this. The .snapshot directory (on some systems) is an example of an immutable directory, though .snapshot directories are properly ignored by the import software. To ignore other directories on this filer, use the gbl-ext-filer *ignore-name* command. After you correct the error, re-enable the share (*enable (gbl-ns-vol-shr)*) to retry the import.
`Error: Detected collisions/inconsistencies during no modify import.`	Two or more of the volume's shares had common file names that either *collided* or had NFS/CIFS naming *inconsistencies*, and this volume disallows import if it encounters either of these problems. As an example of a collision, suppose share A and share B had the same file in the same path, \docs\index.htm: these files would collide. A naming inconsistency can only occur for a directory in a multi-protocol (NFS and CIFS) namespace; the CIFS-side directory name has unicode characters that are inexpressible on the NFS-side (see the documentation for the *character-encoding nfs* command). The volume must be allowed to *modify* the directory (or one of the colliding files) for the import to succeed: the directory or file must be renamed. All duplicate files and naming inconsistencies are recorded in the import reports for each share. These reports are named "import.*job-id.share-name.share-id*.rpt." Use *show reports* to list all import reports and read their contents. Using the import report for each share, resolve all file collisions and naming inconsistencies before re-importing. Go to the filers and rename the files, move them, and/or resolve that certain file renames are acceptable. Once the issues are cleared, use the gbl-ns-vol *reimport-modify* and *modify* commands to allow modification (renames) on import. (If any other shares are still importing, you must wait for their imports to finish before you can use the modify command.) To rename inconsistent NFS/CIFS directories, use the *import rename-directories* unmapped-unicode command, too. After you correct the problem, use *nsck ... rebuild* volume to reimport all shares in the volume.
`Error: Device Offline`	The share's back-end filer went offline for an extended time in the middle of the import. You can remove the share from the volume, or resolve the issue at the filer and retry the import. To retry the import, you can use the *enable (gbl-ns-vol-shr)* command on this share.
`Error: DFS links found on share during import.`	Managed volumes do not support Distributed File System (DFS) links. To find all of the DFS links on all of a volume's shares, you can import the share(s) with no *modify*. Remove all DFS links from the back-end share. Then use the *enable (gbl-ns-vol-shr)* command to retry the import.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Error: Directory attribute collision detected during import.`	A directory on this share has the same name and path as a directory on another share, but the attributes are different. (Directory attributes are permission settings for various users and groups, along with various other flags.) For example, suppose share A had a \docs\img directory and share B had a directory with the same name but different write permissions for the “Everyone” group. To allow the volume to modify directory attributes on import, you can use *modify* on the volume and *import sync-attributes* on the share. (If any other shares are still importing, you must wait for their imports to finish before you can use the modify command.) Then use the *enable (gbl-ns-vol-shr)* command to retry the import.
`Error: Directory collision; Unable to guarantee strict-attribute-consistency.`	(multi-protocol) A directory on this share has the same name and path as a directory on an already-imported share, but some of the CIFS attributes are obscured for at least one of them. Obscured CIFS attributes, due to filer-generated names for the directories, are common on a multi-protocol filer. The volume requires that all attributes (for CIFS and NFS) be synchronized for duplicate directories, and that is impossible under the circumstances. This can only occur for NFS-only directories, with names that are illegal in CIFS. If possible, change the directory name(s) so that they are accessible from CIFS. As an alternative, you can use no *strict-attribute-consistency* to remove the requirement for strict-attribute consistency; this reduces all undiscovered CIFS attributes to zero. Once the volume has stopped importing any shares, you can do this for all shares in the volume. Then restart this share import with the *enable (gbl-ns-vol-shr)* command.
`Error: Directory name collision detected during import.`	A directory on this share has the same name and path as a file on an already-imported share. For example, share B has a directory named /var/log but share A was already imported with a file named /var/log. To allow the volume to correct this by changing the directory name on import, you can use *modify* on the volume and *import rename-directories* on the share. Alternatively, you can directly access the filer(s) and correct the problem there. Retry this share import (with *enable (gbl-ns-vol-shr)*) after you address the issue.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Error: Dynamic Import operation failed due to a collision.`	Two or more of the volume's shares had common file/directory names that somehow *collided*, causing one of the shares to fail its import. The following collisions can cause this failure: • A file in share A has the same name and path as a file in share B. • A directory in share A has the same name and path as a file in share B. • A directory in share A has the same name and path as a directory in share B, and the directories have different attributes (such as permissions). If they had the same attributes, they would not collide. • A file or directory in share A has a CIFS *case collision* with a file or directory on share B, and the volume is set for no *cifs case-sensitive*. For example, "myDir/MYFILE.doc" on share A could collide with "myDir/myFile.doc" on share B. All duplicate files and naming inconsistencies are recorded in the import reports for the share. These reports are named "import.*job-id.share-name.share-id*.rpt." Use *show reports* type Imp to list all import reports, and use *show reports report-name* to read a report. Using the import report for this share, resolve all file collisions and naming inconsistencies before re-importing. Go to the filers and rename the files, move them, and/or resolve that certain file renames are acceptable. You can also use some share-import options to have the volume automatically rename files, rename directories, or reset directory attributes in this share during import (*import rename-files*, *import rename-directories*, or *import sync-attributes*). If you use any share-import options, use the gbl-ns-vol *reimport-modify* and *modify* commands to allow modification (renames) on import. Then re-import the share with the *enable (gbl-ns-vol-shr)* command.
`Error: Dynamic Import operation failed.`	The share import failed for an undetermined reason. Run the *collect* diag-info CLI command to collect diagnostic information, then contact F5 Support.
`Error: Dynamic Import operation timed out.`	An internal import operation timed out, possibly due to a filer-connectivity issue. Use the *show exports* command and/or *ping* to troubleshoot the connection to the filer.
`Error: Encountered file(s) with more than 1024 hard links during import.`	The managed volume software supports a maximum of 1024 hard links per file. One or more files on this back-end share exceed this limit. These files are recorded in the share's import report, named "import.*job-id.share-name.share-id*.rpt." Use *show reports* type Imp to list all import reports, and use *show reports report-name* to read a particular report. Then access the filer directly to reduce the number of hard links for all of these files. After you correct the problem, use *nsck ... rebuild* volume to reimport all shares in the volume.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Error: Export In Use By Another Switch`	The filer share contains a hidden directory, .acopia, written by a different ARX. Two ARXes cannot import the same share. If the share is not imported by another switch anymore, remove the directory manually. If this happens to all shares after a switch replacement, the old switch's UUID was not properly applied to the replacement switch. Consult the appropriate *Hardware Installation* manual for switch-replacement instructions. After you correct the problem, use *nsck ... rebuild* volume to reimport all shares in the volume.
`Error: Export is read-only, cannot import.`	Check the share configuration at the filer: the ARX requires read/write access throughout the share's directory tree. Use the *show exports* command to examine all permission settings at the filer. Use the following guidelines for permissions problems: • For NFS exports, check your back-end filer configuration: the back-end share should allow *root* access to all of the ARX's proxy IP addresses. Use the *show ip proxy-addresses* command to list all configured proxy IP addresses. • For CIFS shares, the switch uses the namespace's *proxy user* as its identity. The proxy user, created by the *proxy-user* command, must belong to the Administrators group. The *proxy-user (gbl-ns)* command sets the proxy user for a namespace. Once the permissions issue is resolved, you can re-import the share with the *enable (gbl-ns-vol-shr)* command.
`Error: Failed import initialization due to filer full condition.`	The back-end share is full, so the ARX volume could not create the .acopia directory and/or write some small files to mark the share. Go to the filer and clear out some disk space before you retry the import. Once some space is free, you can use the *enable (gbl-ns-vol-shr)* command to restart the share import.
`Error: Failed import initialization due to permissions or access error.`	This indicates permissions problems at the back-end filer. Use the *show exports* command to examine all permission settings at the filer. Use the following guidelines for permissions problems: • For NFS exports, check your back-end filer configuration: the back-end share should allow *root* access to all of the ARX's proxy IP addresses. Use the *show ip proxy-addresses* command to list all configured proxy IP addresses. • For CIFS shares, the switch uses the namespace's *proxy user* as its identity. The proxy user, created by the *proxy-user* command, must belong to the Administrators group. The *proxy-user (gbl-ns)* command sets the proxy user for a namespace. Once the permissions issue is resolved, you can re-import the share with the *enable (gbl-ns-vol-shr)* command.
`Error: Failed to Contact Filer.`	The back-end device could not be located with the IP address configured for the filer. From gbl-ext-filer mode, use the *ip address* command to reset the filer's address. Use the *show exports* command, *expect traceroute*, and/or *ping* to check the connection to the filer. After the filer connection is re-established, you can use the *enable (gbl-ns-vol-shr)* command to restart the share import.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action

`Error: Failed to contact filer for attribute lookup.`
 	
`Error: Failed to contact portmap service during import scan.`
	(NFS) These errors indicate an NFS-server problem at the filer. Once the filer's NFS service is restored, you can use the *enable (gbl-ns-vol-shr)* command to restart the share import.

`Error: Failed to create a file/directory on filer.`	This may indicate a full disk on the back-end filer or permissions problems. Use the *show exports* command to examine all permission settings at the filer. Use the following guidelines for permissions problems: • For NFS exports, check your back-end filer configuration: the back-end share should allow *root* access to all of the ARX's proxy IP addresses. Use the *show ip proxy-addresses* command to list all configured proxy IP addresses. • For CIFS shares, the switch uses the namespace's *proxy user* as its identity. The proxy user, created by the *proxy-user* command, must belong to the Administrators group. The *proxy-user (gbl-ns)* command sets the proxy user for a namespace. Once the permissions issue or space issue is resolved, you can re-import the share with the *enable (gbl-ns-vol-shr)* command.
`Error: Failed to create directory collision due to open CIFS file in path.`	(multi-protocol) A directory on this back-end share has a case collision with an already-imported directory, and a CIFS client has a file open in the already-imported directory. For example, suppose the volume imports /vol/mydir from one share, a client connects and opens a file in that share (/vol/mydir/myfile.exe), and then the volume attempts to import a second share with /vol/MYDIR. The volume must mark the first share as "NFS-only" for the import to proceed, but cannot as long as the client holds "myfile.exe" open. The second share therefore fails its import. You can use *show cifs-service open-files* to find the open file, *close cifs file* to close it, and then retry the share import (with the *enable (gbl-ns-vol-shr)* command).

`Error: Failed to find valid mount version during import scan.`
 	
`Error: Failed to mount filer during import scan.`
 	
`Error: Failed to open NFS connection to filer during import scan.`
	(NFS) These errors indicate an NFS-server problem at the filer. Once the filer's NFS service is fully restored, you can use the *enable (gbl-ns-vol-shr)* command to restart the share import.

`Error: Failed to resolve IP address for filer.`	The back-end device could not be located with the filer name or IP address configured for the filer. From gbl-ext-filer mode, use the *ip address* command to reset the filer's address. Use the *show exports* command and/or *ping* to troubleshoot the connection to the filer. After the filer connection is re-established, you can use the *enable (gbl-ns-vol-shr)* command to restart the share import.
`Error: Failed to retrieve Kerberos information.`	(CIFS) The namespace supports Kerberos authentication (see *cifs authentication*), but the namespace software is unable to confirm that the share is configured to support Kerberos, too. Check the connection to the back-end filer with *show exports* and/or *ping*. Restart the import (with *enable (gbl-ns-vol-shr)*) after you correct the problem.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Error: Failed to take pathlock due to internal operation in progress.`	The share import failed due to an internal-software conflict, possibly due to a timing issue or a brief filer-connectivity issue. Retry the import by using the *enable (gbl-ns-vol-shr)* command on this share. If this error returns, contact F5 Support.
`Error: Failed to unmount filer during import scan.`	(NFS) Persistent errors from a back-end share caused the mount to fail. Check the NFS service at the back-end filer. Once the filer's NFS service is fully restored, you can use the *enable (gbl-ns-vol-shr)* command to restart the share import.
`Error: File collision detected during import.`	A file on this share has the same name and path as a file on an already-imported share. To fix this, you can manually go to the filer and rename the file, or you can set the *modify* flag on this volume. By setting the modify flag, you allow the volume to rename the file on import. (If any other shares are still importing, you must wait for their imports to finish before you can use the modify command.) You must also have the default settings for *import rename-files* and *import rename-directories* on this share. Then restart the import with the *enable (gbl-ns-vol-shr)* command.
`Error: File exists in CIFS, but not in NFS.`	(multi-protocol) The share is configured for both NFS and CIFS, but a directory that is visible in the back-end-CIFS share is not found in the NFS export. The filer probably has two different names for the directory: one for CIFS clients and one for NFS clients. Check the directory at the back-end share, and rename it so that both versions have the same name. Alternatively, you can set the *modify* flag on this volume. By setting the modify flag, you allow the volume to rename the directory on import. (If any other shares are still importing, you must wait for their imports to finish before you can use the modify command.) You may also need to set *import rename-directories* unmapped-unicode for this share; this allows the volume to rename directories whose CIFS names do not map to the character encoding for NFS. Then restart the import with the *enable (gbl-ns-vol-shr)* command.
`Error: filename collision.`	A file on this share has the same name and path as a file on an already-imported share. To fix this, you can manually go to the filer and rename the file, or you can set the *modify* flag on this volume. By setting the modify flag, you allow the volume to rename the file on import. (If any other shares are still importing, you must wait for their imports to finish before you can use the modify command.) You must also have the default settings for *import rename-files* and *import rename-directories* on this share. Then retry the import.
`Error: Filer does not have enough free space to perform import.`	(NFS) The ARX creates a temporary database on each share during the import process: there is not enough room on this share to create the database. The database requires at least 100 MegaBytes of free space. Clear some free space at the filer share, then restart the import with the *enable (gbl-ns-vol-shr)* command.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Error: Filer does not provide an NFS filehandle for a file in its directory listing.`	(multi-protocol) The filer provided a file name in an NFS READDIR call, but did not translate that file name into a filehandle in a subsequent LOOKUP call. Some multi-protocol filers behave this way when a CIFS file name has characters that are not supported by NFS; the filer returns the UTF-8 version of the file name for the READDIR call, which it later rejects in a subsequent LOOKUP call. The ARX can see the file in the directory listing, but cannot get access to the file itself. Rename the file at the filer share, then use *nsck ... rebuild* volume to reimport all shares in the volume.
`Error: Filer does not support specified protocols.`	The protocol(s) configured for the back-end share are not actually supported at the filer. Use the *show exports* command to check the protocols supported by the filer. Use the *show global-config* namespace command to view the required protocols for the namespace. The share must support all of the namespace's protocols. You can remove the share from the volume (with no *share*), or you can add protocol support at the back-end filer. If you add the protocol support to the filer, you can then restart the share import with the *enable (gbl-ns-vol-shr)* command.
`Error: Filer error during import.`	This may indicate a disk failure at the external filer. Correct the filer issue and retry the import (with the *enable (gbl-ns-vol-shr)* command).
`Error: Filer generated name collision is not supported by filer.`	(multi-protocol) The share contains an entry whose filer-generated name (FGN) matches a real name in an already-imported share, or whose real name conflicts with an already-imported FGN. The volume is backed by a mix of Hitachi HNAS (BlueArc) filers and filers from other vendors, and this naming conflict is not supported for this vendor combination. The import report shows the name of the conflicting file or directory. The name appears with an "FC" notation. The report is named "import.*job-id.share-name.share-id*.rpt." Use *show reports* to list all import reports and read their contents. Using the import report, resolve all of these name collisions before re-importing. Go to the filer to rename the files and/or directories. Then use *enable (gbl-ns-vol-shr)* to re-import the share.
`Error: Filer operation returned an unexpected error, see logs for more information.`	The back-end filer behind this share returned an error that the ARX does not recognize. You can use the *show logs* syslog command to view the system log and learn more about the circumstances around the failure. We recommend that you contact F5 Support if you see this import error. You may be requested to run the *collect* command, which assembles diagnostic information for F5 Engineering. After the issue is resolved, use *nsck ... rebuild* volume to reimport all shares in the volume.
`Error: Filer returned invalid link count for a file.`	The import failed because the back-end filer returned a link count of zero or a negative number for a file, which is invalid. A file's link count should be one or more. The ARX syslog contains the file with the invalid link count; use *grep* "link count" logs syslog to find the path. Use this path to help diagnose and correct the filer issue. Then use the *enable (gbl-ns-vol-shr)* command to retry the import.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Error: Hard Links Inconsistent`	(NFS) One or more files in the exported directory have hard links from outside the directory. That is, the switch tried to import a hard link to file X but detected that there was at least one more hard link to file X in some other directory, not included in the NFS export. Remove the hard links from the external directories and retry the share import.
`Error: Hard Links Not Supported`	(CIFS) Hard-links exist in a CIFS share (not supported in CIFS by the ARX).
`Error: I/O error encountered talking to filer.`	Use the *show exports* command and/or *ping* to check the connection to the filer. Restart the import (with *enable (gbl-ns-vol-shr)*) after the connection is restored.
`Error: Import database I/O failure.`	(NFS) The ARX creates a temporary database during the import process. The database was deleted while the import was underway. This may indicate that other clients are bypassing the ARX to access the share, which is an unsupported configuration. You can restart the share import with the *enable (gbl-ns-vol-shr)* command.
`Error: Insufficient files reserved for the volume.`	This share contains more files than the volume can hold. Use the *auto reserve files* command to automatically increase the number of files that this volume can hold as the volume grows. If you prefer to manually set the maximum files for the volume, use *reserve files* to manually increase the maximum. Then restart the share import with the *enable (gbl-ns-vol-shr)* command.
`Error: Insufficient free space to store metadata.`	A volume's metadata share requires gigabytes of free space; the one chosen for this volume has less than one-half gigabyte. Choose a metadata share with more free space: use the *metadata share* command to configure a new metadata share. After you choose another metadata share, or clear up several gigabytes of free space on the current metadata share, you can use the *enable (gbl-ns-vol-shr)* command to restart the import.
`Error: Internal IPC communications failure during import.` `Error: Internal thread error during import.`	These errors each indicate an internal software problem. Run the *collect* diag-info CLI command to collect diagnostic information, then contact F5 Support. Once the issue is understood, you can restart the share import with the *enable (gbl-ns-vol-shr)* command.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Error: Maximum number of namehash collisions for a filename reached in a directory.`	A directory in this share has too many entries (files and/or subdirectories) whose names differ only in case (for example, “thisFile.txt,” “THISfile.txt,” “thisFILE.txt,” and so on). The limit on case collisions for a single entry name is over 16,000. A client tried to create a new entry that would exceed the limit for a particular entry name, and the volume blocked the action with an “access denied” error. You can resolve this issue by accessing the volume as a client and moving as many of the file entries as possible. Create a new directory to hold many of them, or rename them to a non-matching name. For example, you could change the earlier examples to “thatFile.txt,” “THATfile.txt,” and so on to stop them from making a case-insensitive match with “thisfile.txt.” The specific directory and file name appears in a syslog message labeled “ERROR_MAX_HASH_COLLISIONS.” Use *grep* ERROR_MAX_HASH_COLLISIONS logs syslog to search for this error in the syslog. Note that the colliding names are usually case collisions, but not always. The collision occurs after a mathematical conversion of the file names, called a *hash*. After you clear enough colliding files from the directory, you can restart the share import with the *enable (gbl-ns-vol-shr)* command.
`Error: Maximum Path Length Exceeded`	A path to one of the files exceeds the maximum, 1024 characters. You must shorten the path(s) to import this share. Once all of the share’s paths are below 1024 characters, you can restart the share import with the *enable (gbl-ns-vol-shr)* command.
`Error: Metadata database I/O failure.`	There was a database I/O failure for the share. This may be caused by a transient network error, or a filer problem. You can use the *show exports* command to check the filer and connection for common problems. Once the external problem is resolved, use *nsck ... rebuild* volume to reimport all shares in the volume.
`Error: Metadata filer configuration is invalid. Data written has been lost, corrupting the database.`	This is not an import error: it typically occurs while the volume is providing service. This condition disables the volume. Some of this share’s metadata was lost at the volume’s metadata share. This indicates that the metadata share did not synchronously write its data; it wrote the data to a cache, told the ARX that the write succeeded, and then lost the data between the cache and the disk. This is only possible on an NFS export where either • the “async” option was set, or • “async” is the default for the NFS implementation (as with some releases of Linux) CIFS shares do not have this problem. At the filer, set the “sync” option for the NFS export. We also recommend that you specify the “no_wdelay” option. Use *nsck ... rebuild* volume to re-initialize the metadata share and reimport all shares in the volume.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Error: Metadata share unavailable during import.`	The switch could not contact the metadata share. Use the *show exports* command and/or *ping* to check the connection to the metadata share's filer. The show export command also verifies that the share is accessible by *root* (for NFS shares) or the namespace's *proxy-user (gbl-ns)* (for CIFS shares). After the connection is re-established, restart the share import with the *enable (gbl-ns-vol-shr)* command.
`Error: Mismatched NFS filename encoding.`	(multi-protocol) The NFS character-encoding setting for the namespace does not match the character encoding supported at the filer. If this share was imported, lost NFS files could result. Reset the namespace character encoding (using the *character-encoding nfs* command) and retry the import. You can use the *enable (gbl-ns-vol-shr)* command to restart the share import.
`Error: Mount to filer failed.`	(NFS) The switch was unable to NFS-mount the filer share. Check the NFS service at the back-end filer.
`Error: Multi-protocol share is split.`	(multi-protocol) The *filer* command specified an NFS export and a CIFS share over two different directory trees. This is unsupported. Retry the command with the correct share and export names, then retry the import with *enable (gbl-ns-vol-shr)*.
`Error: Multi-protocol user mapping error. (nfs read failure)`	(multi-protocol) During import, the volume creates a test file through CIFS and then attempts to read it through NFS. The volume was unable to read the file as *root*. Check the NFS configuration at the back-end share, correct the problem, and retry the import (*enable (gbl-ns-vol-shr)*).
`Error: Multi-protocol user mapping error. (nfs remove failure)`	(multi-protocol) During import, the volume creates a test file through CIFS and then attempts to delete it through NFS. The volume could read and write the file (as *root*), but was unable to remove it. This may indicate a permissions problem in the top-level directory for the share. Check the NFS configuration at the back-end share, correct the problem, and retry the import (*enable (gbl-ns-vol-shr)*).
`Error: Multi-protocol user mapping error. (nfs write failure)`	(multi-protocol) During import, the volume creates a test file through CIFS and then attempts to write to it through NFS. The volume could read the file, but was unable to write to it (as *root*). Check the NFS configuration at the back-end share, correct the problem, and retry the import (*enable (gbl-ns-vol-shr)*).

`Error: NFS connect failed.`
`Error: NFS error during import scan.`
`Error: NFS error occurred during import scan.`
`Error: NFS Error while scanning storage`
	(NFS) Persistent NFS errors from a back-end share caused the import to fail. Check the NFS service at the back-end filer.
	After the filer's NFS service is fully restored, you can restart the share import with the *enable (gbl-ns-vol-shr)* command.

`Error: NFS Version Mismatch`	(NFS) The back-end share does not support the NFS version(s) configured for the external filer. Use the *show exports* command to check the protocols supported by the filer. Use the *filer* command to change the configured NFS version(s) for the share/export.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Error: NFSv2 error during import scan.` `Error: NFSv3 error during import scan.`	(NFS) Persistent NFS errors from a back-end share caused the import to fail. Check the NFS service at the back-end filer. After the filer's NFS service is fully restored, you can restart the share import with the *enable (gbl-ns-vol-shr)* command.
`Error: Network communications failure during import scan.`	Use the *show exports* command, *expect traceroute*, and/or *ping* to troubleshoot the connection to the filer. After the filer connection is re-established, you can restart the share import with the *enable (gbl-ns-vol-shr)* command.

`Error: Network error or timeout during CIFS attributes check.`
`Error: Network error or timeout during CIFS privilege check.`
`Error: Network error or timeout during CIFS write check.`
	(CIFS) A connection error occurred in the middle of a CIFS-permissions test. Use the *show exports* command, *expect traceroute*, and/or *ping* to troubleshoot the connection to the filer. Once the connection is fully restored, you can restart the share import with the *enable (gbl-ns-vol-shr)* command.

`Error: No CIFS write access.`	The namespace's *proxy-user* does not have adequate privileges to write to this CIFS share, so the import failed. The proxy user must belong to the Administrators group on this filer. You can choose new, more-privileged credentials for your proxy user, or you can go to the filer and add the current proxy user to a more-privileged group. The *probe exports* command can verify that the new proxy-user credentials pass this write test. Then restart the import with the *enable (gbl-ns-vol-shr)* command.
`Error: No space left on share, or share unreachable` `Error: No space on back-end device.`	The ARX got an error writing a database that is created as part of the import process. This database is created on the filer share itself, so the error indicates a problem with the filer. Clear some free space at the filer share, then restart the import with the *enable (gbl-ns-vol-shr)* command.
`Error: Operation aborted at user request.`	An administrator issued the *cancel import* command to stop this share import. You can restart the import with the *enable (gbl-ns-vol-shr)* command.
`Error: Permission Denied`	This indicates that the ARX did not have proper permissions to connect to the back-end filer. For NFS exports, check your back-end filer configuration: the back-end share should allow access to all of the ARX's proxy IP addresses. Use the *show exports* command to check the filer's permissions and configuration. Use the *show ip proxy-addresses* command to list all configured proxy IP addresses. For CIFS shares, the switch uses the proxy user for the namespace; the *proxy-user (gbl-ns)* command sets these credentials. The proxy user must belong to the Administrators group. After the permissions problem is corrected, you can restart the import with the *enable (gbl-ns-vol-shr)* command.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Error: Protocol specific error, see logs for more information.`	(CIFS) The back-end filer returned an unexpected CIFS error during import. The syslog shows the specific error. Use *show logs* syslog to read the syslog, or *grep string* logs syslog to search for a specific string in the syslog. You may need to escalate to F5 Support. After you correct the error, use *nsck ... rebuild* volume to reimport all shares in the volume.
`Error: Proxy User does not map to root user via NFS.`	(multi-protocol) The *proxy user* is a Windows username and password that the volume can use as its identity for share import and for policy operations. In a multi-protocol (CIFS and NFS) namespace, the proxy user on the Windows side must map to the *root* user on the UNIX side. You can select a new proxy user for the namespace with the command. If necessary, map the proxy user to root at the filer itself; the *ARX Site Planning Guide* has instructions for creating this mapping on common multi-protocol filers.
`Error: Remove incomplete: found files on share.`	An administrator failed to remove the share with no *share* because client-visible files are still present on the share. Use the remove-file-entries option to remove all of the file entries from the volume; this produces a client-visible effect, so do this with caution. Alternatively, you can use *remove-share nomigrate*.
`Error: Root Squash is Enabled`	(NFS) This indicates that the ARX did not have proper permissions to create files at the back-end filer. Check your back-end filer configuration: the back-end share should have *no-root-squash* set for all of the ARX's proxy IP addresses. (On some filers, you accomplish this by mapping the *anonymous* user to UID 0 (zero).) Use the *show exports* command to check the filer's permission settings. Use the *show ip proxy-addresses* command to list all configured proxy IP addresses. Once the filer's root-squash option is disabled, you can restart the import with the *enable (gbl-ns-vol-shr)* command.
`Error: RPC error during import scan.`	(NFS) Persistent NFS errors from a back-end share caused the import to fail. Check the NFS service at the back-end filer. After the filer's NFS service is fully restored, you can restart the share import with the *enable (gbl-ns-vol-shr)* command.
`Error: Share attributes incompatible with volume attributes.`	(CIFS) The CIFS attributes set for the volume (with *compressed-files*, *named-streams*, *persistent-acls*, *sparse-files*, and/or *unicode-on-disk*) are not all supported at the back-end share. Use the *show exports* command to check the supported CIFS attributes for the share. You can remove the share from the volume (with no *share*) or disable the conflicting CIFS attribute(s) in the managed volume. If you elect to keep the share in the volume, use the *enable (gbl-ns-vol-shr)* command to restart the share import.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Error: Share does not support Kerberos authentication.`	(CIFS) The namespace supports Kerberos authentication (see *cifs authentication*), but this share does not. The share must support *extended security negotiations* for the import to succeed. Also, the ARX needs the correct service-principal name (*SPN*) for the filer; you can use *show exports* ... capabilities to verify that the ARX has discovered the correct SPN for the filer, or you can use the *spn* command to set it manually. After you ensure that the filer support Kerberos and the ARX has the filer's SPN, you can restart the import (*enable (gbl-ns-vol-shr)*). Alternatively, you can remove the share from the volume with no *share*.
`Error: Share type does not match namespace supported protocols.`	Each back-end share must support *all* of the namespace's configured protocols (any combination of NFSv2, NFSv3(/UDP), NFSv3/TCP, and CIFS). Use the *show global-config* namespace command to view the namespace's protocols. You can remove the share from the volume (with no *share*) or enable the missing service(s) at the filer. If you elect to keep the share in the volume, use the *enable (gbl-ns-vol-shr)* command to restart the share import.

`Error: Specified export is not a directory.`
`Error: Specified export is too long.`
`Error: Specified export not found on filer.`
	The export name is incorrect in the external-filer configuration. Use the *filer* command to change the configured name for the share/export.

`Error: Stat of a file/directory failed during storage init.`	(NFS) The ARX was unable to access the attributes of a file or directory. Check the NFS service at the back-end filer. To check the ARX's connectivity to the filer and perceived permissions at the filer, use *show exports*. You can restart the share import with *enable (gbl-ns-vol-shr)* after you resolve the filer issue.
`Error: Storage is already in use by this switch pair.`	The share is already imported into another managed volume on this switch (or its redundant peer). Evidently, the back-end share has at least one alias and it was previously imported under another name. A share can only be managed by one volume at a time, under a single name. From gbl-ns-vol-shr mode in the CLI, use no *filer* to detach from the back-end share. Then choose another back-end path with the *filer* command, or use no *share* to remove the share from the volume. To restart the import (with or without this share), use *nsck ... rebuild* volume to reimport all shares in the volume.
`Error: Storage remove failed because pending directory operations could not be flushed.`	Someone attempted to remove a share (with no *filer*, no *share*, *remove-share migrate*, *remove-share nomigrate*, or *remove service*), and an internal error caused the removal to fail. Contact F5 Support if you see this message.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Error: Unable To Connect to CIFS Share.`	The connection to the back-end CIFS share failed due to possible configuration errors or a broken connection to the back-end filer. Use the *show exports* command, *expect traceroute*, and/or *ping* to troubleshoot the connection to the filer. Once the connection is fully established, you can restart the share import with the *enable (gbl-ns-vol-shr)* command.
`Error: Unable to read file due to stale filehandle.`	(NFS) The filer provided a filehandle in response to an earlier NFS call, and is now pronouncing it "stale." This is inconsistent filer behavior; it may indicate that NFS clients are bypassing the ARX to access the share. Check the filer share and correct the problem there. Once the filer issue is corrected, use *nsck ... rebuild* volume to reimport all shares in the volume.
`Error: Unexpected CIFS privilege check failure.` `Error: Unexpected CIFS write check failure.`	A CIFS-permission test failed at the filer for an undetermined reason. This may be a filer issue or a connectivity issue. Check the filer and the connection, and retry the share import (with *enable (gbl-ns-vol-shr)*). If this error stops the import a second time, run the *collect* diag-info CLI command to collect diagnostic information and contact F5 Support.
`Error: Unresolvable case-blind name collision.`	(CIFS) A file on this share has the same name and path as a file on an already-imported share, based on a case-blind comparison, and the volume is configured with no *cifs case-sensitive*. That is, some of the characters have differing cases, but the characters match (for example, "index.htm" matches "index.HTM"). If the volume is not case-sensitive, it cannot see the difference between the two names. You have three options to address this error: • Manually go to the filer and rename the file. • Set the *modify* flag on this volume. By setting the modify flag, you allow the volume to rename the file on import. You must wait for all the volume's shares to finish importing before you can use this command. • Use *cifs case-sensitive* to make the volume case-sensitive. Then retry the import with the *enable (gbl-ns-vol-shr)* command.
`Import Interrupted`	An administrator stopped the import with the *cancel import* command. You may be able to restart the import with no *enable (gbl-ns-vol-shr)* and then enable. If the import was stopped too far in the process, you must first use *nsck ... destage* to shut down the volume, remove and re-add the share, then enable the volume again.
`Internal Error (`*number*`)`	Internal problem; contact F5 personnel.
`Metadata Only`	The share is designated to store namespace metadata only, so there are no client-accessible files to import. This serves as an explanation; it is not an import error.
`Offline: Volume Disabled`	No imports are possible unless the volume is enabled. Use the *enable (gbl-ns, gbl-ns-vol)* command to enable the volume.
`Offline: Volume Failed`	The metadata share for the volume failed to import. Use *metadata share* to designate a new dedicated share for metadata.

Table 10.1 *Share Status Conditions (Continued)*

Status Condition	Description/Action
`Remove Interrupted`	An administrator stopped the share removal with the *cancel remove* command. To restart the removal process, use *remove-share migrate*, *remove-share nomigrate*, no *share*, or no *filer*.
`Storage job aborted.`	The import process was interrupted by an *nsck ... rebuild* force. The rebuild operation will re-import the share.
`Storage job aborted: DNAS is shutting down.`	DNAS, the internal name for volume software, is shutting down. This may be the result of administrative action, such as a *remove service*, during the import. The volume has stopped running, so the import is canceled.
`Uninitialized (No filer assigned)`	Use the *filer* command to assign a filer to the share, then retry the import.

Table 10.1 *Share Status Conditions (Continued)*

Restarting the Import

After you correct the problem, you can restart the import for the share. To remove the share from the managed volume, skip ahead to the next section.

In most cases, you can restart a failed import by re-enabling it. For details on enabling a share in a managed volume, refer to *Enabling the Share*, on page 9-45 of the *ARX® CLI Storage-Management Guide*.

For example, this command sequence successfully restarts an import for the wwmed~/acct~budget share:

```
bstnA(gbl)# namespace wwmed
bstnA(gbl-ns[wwmed])# volume /acct
bstnA(gbl-ns-vol[wwmed~/acct])# share budget
bstnA(gbl-ns-vol-shr[wwmed~/acct~budget])# enable
bstnA(gbl-ns-vol-shr[wwmed~/acct~budget])# ...
```

The alternatives below are for situations where you have reconsidered using this share in the volume, or where the above table indicates that you must rebuild the entire volume.

Removing the Share from the Volume

You can remove the volume from the share as a first step to retrying the import, or as a final step.

Any imported directories and files are visible to the volume's clients, so you begin by choosing how to handle them. You can migrate all files and directories to other shares in the volume, migrate only the directories, or remove all of the share's files and directories. The sections below describe each of these options.

The entire volume fails if its metadata share fails or its only share fails. In either of these cases, you must rebuild the entire volume; skip to the next section for this situation.

Retaining Both Directories and Files in the Volume

If any files were imported from the share, you can migrate them elsewhere in the volume before the share is removed. The files move along with the share's master directories. No files remain on the share after this process is over.

Use the remove-files migrate command to migrate all files and directories to another share before removing it from the volume. This command was described above; see *Removing an Imported Share*, on page 10-18.

Note

The share with the master instance of a directory (or master directory) is considered the "home" share for the directory. (Recall that a volume can have a copy of any directory in multiple back-end shares.) This is the first-imported instance of a the directory, or possibly the only instance. By default, the volume places any new files for the directory into its master instance.

Removing the share ensures that none of its directories are master any longer. If the share never returns to the volume, this is irrelevant. If you add the share back later, the volume tends to avoid putting new files onto the share. This does not affect clients, but may create an unexpected file distribution at the back end.

Suppose share A has two directories, "/usr" and "/log," and you remove it from the volume. The masters for "/usr" and "/log" migrate to another share in the volume, share B. When share A comes back later, all new files that clients create in "/usr" and "/log" will gravitate to share B, not share A. Only policy can cause files to move into share A. As stated above, clients see no difference.

For example, this command sequence removes the budget share from the /acct volume, migrating all of its files and master directories into the "bills" share:

```
bstnA(gbl)# end
bstnA# remove-share migrate wwmed /acct budget bills

WARNING !! Share 'budget' will be removed from volume '/acct' in
namespace 'wwmed' after migrating files to 'bills'
Proceed? [yes/no] yes

bstnA# ...
```

Retaining Imported Directories (Not Files) in the Volume

To keep imported directories in the volume but remove any of the share's imported files, use the remove-files nomigrate command. This was described in *Removing Shares Without File Migration*, on page 10-24. This moves all of the share's master directories into another share in the same volume, then removes the share from the volume. Any imported files disappear from client view.

Note

As above, the migration of the master directories places the "home" for this share's directories on the other share.

For example, this command sequence removes the budget share from the /acct volume, moving all of its (empty) master directories into the "bills" share:

```
bstnA(gbl)# end
bstnA# remove-share nomigrate wwmed /acct budget bills

WARNING !! Share 'budget' will be removed from volume '/acct' in
namespace 'wwmed' WITHOUT migrating files.
Proceed? [yes/no] yes

bstnA# ...
```

Retaining Nothing from the Share

To remove all traces of the share, you must remove all of the share's directories from the volume's directory tree. This is a serious change in the volume's metadata, and it requires that you take the volume offline. Use nsck ... destage to take the volume offline (recall *De-Staging a Single Volume*, on page 7-38), remove the share from the volume, then re-enable all of the volume's remaining shares (as shown in *Enabling All Shares in the Volume*, on page 9-54 of the storage-management manual).

Important

Clients cannot access the volume while it is destaged. NFS clients must unmount and remount after the volume comes back online. Additionally, all NFS and CIFS services that host the volume experience a brief (roughly 10-second) outage.

For example, the following command sequence shuts down the "wwmed~/acct" volume, removes the "budget" share, and then enables all of the remaining shares at once:

```
bstnA(gbl)# end
bstnA# nsck wwmed destage volume /acct

Volume /acct is in use by NFS global service 'ac1.medarch.org'.

This operation will remove entries for all shares in the volume
from the namespace metadata.  To reimport a destaged share,
enable the share.
```

```
% WARNING: Volume /acct in namespace wwmed is in use by global services.

Destaging the volume will disrupt all clients using this volume.

Destage the volume anyway? [yes/no] yes
bstnA# global
bstnA(gbl)# namespace wwmed
bstnA(gbl-ns[wwmed])# volume /acct
bstnA(gbl-ns-vol[wwmed~/acct])# no share budget
bstnA(gbl-ns-vol[wwmed~/acct])# enable shares
bstnA(gbl-ns-vol[wwmed~/acct])# ...
```

Reinstating the Share

If you want to bring the share back into the volume, repeat its original configuration. When you enable the share in the already-enabled volume, the volume imports the share's files and directories.

For example, the following command sequence returns the "budget" share to the "/acct" volume:

```
bstnA(gbl)# namespace wwmed
bstnA(gbl-ns[wwmed])# volume /acct
bstnA(gbl-ns-vol[wwmed~/acct])# share budget
This will create a new share.

Create share 'budget'? [yes/no] yes
bstnA(gbl-ns-vol-shr[wwmed~/acct~budget])# filer das1 nfs3 /exports/budget
bstnA(gbl-ns-vol-shr[wwmed~/acct~budget])# enable
bstnA(gbl-ns-vol-shr[wwmed~/acct~budget])# exit
bstnA(gbl-ns-vol[wwmed~/acct])# ...
```

Reimporting the Entire Volume

The entire volume fails if its metadata-share fails or if its only share fails. In either case, you use nsck ... rebuild volume to delete the volume's metadata and completely rebuild it (as described in *Rebuilding a Volume*, on page 7-33).

Important

This is service-affecting in the same way as nsck ... destage*, described above.*

For example, this command sequence rebuilds the wwmed~/it volume:

```
bstnA(gbl)# end
bstnA# nsck wwmed rebuild volume /it
Reimport all shares from volume /it in namespace wwmed? [yes/no] yes
bstnA# ...
```

Removing a Full Namespace Service

You can use a single command to remove a namespace, all of its volumes, all the global servers and services that export its volumes, and any other configuration objects that are used only by this namespace. All of these components together comprise a single namespace *service*. From priv-exec mode, use remove service command to remove the full namespace service:

remove service *namespace* [timeout *seconds*] [sync]

where:

namespace (1-30 characters) is the namespace to remove,

seconds (optional, 300-36,000) sets a time limit for the removal of each namespace component, and

sync (optional) waits for the removal to finish before returning. With this option, the CLI lists the namespace and global-server components as it removes them.

The CLI prompts for confirmation before removing the namespace and all of its services. Enter **yes** to continue.

This operation generates a report, "remService_*namespace_date*.rpt," which catalogs all of the actions that it took. The *namespace* in the file name identifies the removed namespace, and the *date* is the date and time when the command started. The CLI shows the report name after you invoke the command. Use show reports to see the file listing; use show, tail, or grep to read the file. To save the report off to an external site, use the copy ... ftp or copy ... scp command from priv-exec mode.

The command does not create the report if you use the sync option; it shows its actions at the command line instead.

For example, this command sequence removes the medco namespace and later reviews the report:

```
bstnA(gbl)# end
bstnA# remove service medco

Remove service 'medco'? [yes/no] yes
Scheduling report: remService_medco_201002240751.rpt

bstnA# ...
bstnA# show reports remService_medco_201002240751.rpt
**** Remove Namespace Report: Started at Wed Feb 24 02:51:30 2010 ****
**** Software Version: 5.02.000.12538 (Feb 16 2010 20:14:13) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:
% COMMAND: Remove Service medco
% INFO: nsck medco destage force
% INFO: wait-for nsck medco 300
% INFO: no enable; namespace medco
% INFO: wait-for volume-disable medco 300
% INFO: Checking service configuration for namespace medco
% INFO: Checking CIFS browsing for namespace medco
% INFO: Checking NFS services for namespace medco
% INFO: no nfs acopiaFiler
% INFO: Checking CIFS services for namespace medco
% INFO: Checking global servers for namespace medco
```

```
% INFO: no global server acopiaFiler
% INFO: Checking volume policies for namespace medco
% INFO: Removing Policy medco /vol
% INFO: Checking CIFS browsing for namespace medco
% INFO: Checking shares for namespace medco
% INFO: no share corporate
% INFO: no share sales
% INFO: no share generic
% INFO: Checking volume metadata shares for namespace medco
% INFO: Checking volumes for namespace medco
% INFO: no volume /vol
% INFO: Checking namespace metadata shares for namespace medco
% INFO: Checking namespace medco
% INFO: no namespace medco
% INFO: Checking NFS ACLs for namespace medco
% INFO: Checking external-filers for namespace medco
% INFO: no external-filer nas2
% INFO: no external-filer nas3
**** Total processed:              0
**** Elapsed time:          00:00:08
**** Remove Namespace Report: DONE at Wed Feb 24 02:51:38 2010 ****
bstnA# ...
```

Removing All Policy Objects from a Namespace

You can use a single command to remove all rules, share farms, and other policy objects from a namespace. The remove namespace command has been described in the *ARX® CLI Storage-Management Guide*: see *Removing a Direct Volume*, on page 8-31, *Removing a Managed Volume*, on page 9-68, or *Removing a Namespace*, on page 7-28. Add the optional policy-only keyword to remove only the policy objects:

`remove namespace` *`name`* `policy-only [timeout` *`seconds`*`] [sync]`

where:

name (1-30 characters) is the name of the namespace,

policy-only is the option to remove only the policy objects,

seconds (optional, 300-10,000) sets a time limit on each of the removal's component operations, and

sync (optional) waits for the removal to finish before returning. With this option, the CLI lists the policy objects as it removes them.

The CLI prompts for confirmation before removing the policy objects. Enter **yes** to continue. This operation generates a report named "removeNs_*namespace_date*.rpt" if you omit the sync option.

For example, this command sequence exits to priv-exec mode and then synchronously removes all policy objects from the "insur_bkup" namespace:

```
prtlndA(gbl)# end
prtlndA# remove namespace insur_bkup policy-only timeout 600 sync

Remove components from namespace 'insur_bkup'? [yes/no] yes
% INFO: Removing service configuration for namespace insur_bkup
```

```
% INFO: Removing volume policies for namespace insur_bkup
% INFO: destroy policy insur_bkup /insurShdw
prtlndA# ...
```

Removing All Policy Objects from a Volume

The optional volume argument focuses the remove namespace command on one volume:

remove namespace *name* volume *volume* policy-only [timeout *seconds*] [sync]

where:

name (1-30 characters) is the name of the namespace,

volume (1-1024 characters) is the path name of the volume,

policy-only is the option to remove only the policy objects, and

seconds (optional, 300-10,000) sets a time limit on each of the removal's component operations, and

sync (optional) waits for the removal to finish before returning, as described above.

For example, this command sequence exits to priv-exec mode and then removes all policy objects from the "insur~/claims" volume:

```
bstnA(gbl)# end
bstnA# remove namespace insur volume /claims policy-only

Remove policy components from volume '/claims' in namespace 'insur'? [yes/no] yes
...
```

Managing Namespace Collisions

When a managed volume is enabled, the ARX takes inventory from all of the volume's back-end shares and includes their files into the volume. An important part of this process is to ensure that there is only a single instance of any given path or file name. If you have a path or file name that is identical on more than one back-end share, there is an import *collision*; only one of the files can be imported into the managed volume. The redundant file is renamed as follows:

file-name_share-jobid.ext

where

file-name is the file's original name, without its extension (for example, "myfile" from "myfile.doc"),

share is the name of the namespace share,

jobid is an integer identifying the job that imported the share, and

.ext is the files original extension (for example, "doc"), if there is one.

If there is more than one redundant file, an index is added:

file-name_share-jobid-index.ext

where *index* is a number starting at 1.

The same naming conventions are applied to a directory whose name matches an already-imported file, or to a directory that has different file attributes than an already-imported directory. In addition, any tilde (~) characters in a directory name are replaced with commas (,).

Finding the Renamed Files

To find any files that have collided and been renamed, you can view the report that was generated during import. Each share generates one import report each time it imports (import occurs when the share and volume are first enabled, or if you use nsck ... rebuild to rebuild the volume). Use the show reports type Imp command to see all import reports.

For example, this switch has several import reports:

```
bstnA(cfg)# show reports type Imp


  reports
    Codes: Imp=Import
      import.1.budget.5.rpt   Feb 24 00:57  1.9 kB        Imp  DONE: 713 in 00:00:04
      import.10.backlots.17.rpt Feb 24 01:00  2.0 kB        Imp  DONE: 1 in 00:00:04
      import.11.scanners.18.rpt Feb 24 01:00  2.2 kB        Imp  DONE: 5 in 00:00:03
      import.12.shr1-old.26.rpt Feb 24 01:14  5.6 kB        Imp  DONE: 161 in 00:00:08
      import.13.shr1-next.27.rpt Feb 24 01:14  2.5 kB        Imp  DONE: 14 in 00:00:08
      import.2.bills.6.rpt    Feb 24 00:57  1.9 kB        Imp  DONE: 122 in 00:00:03
      import.3.bills2.7.rpt   Feb 24 00:57  2.5 kB        Imp  DONE: 415 in 00:00:03
      import.4.it5.8.rpt      Feb 24 00:58  1.9 kB        Imp  DONE: 131 in 00:01:01
```

```
      import.5.rx.11.rpt      Feb 24 00:59  1.8 kB        Imp  DONE: 31 in 00:00:06
      import.6.charts.12.rpt  Feb 24 00:59  2.1 kB        Imp  DONE: 160 in 00:00:06
      import.7.bulk.13.rpt    Feb 24 00:59  1.9 kB        Imp  DONE: 4 in 00:00:06
      import.8.equip.15.rpt   Feb 24 00:59  1.8 kB        Imp  DONE: 19 in 00:00:04
      import.9.leased.16.rpt  Feb 24 01:00  1.9 kB        Imp  DONE: 42 in 00:00:04

bstnA(cfg)#
```

Use show reports *file-name* to view the import report. A collision appears in the File Scan Phase section, with a Type of "R" (Rename) and "NC" (name collision). For example, the following report shows three collided files (highlighted in bold):

```
bstnA(cfg)# show reports import.3.bills2.7.rpt
**** Share Import Report: Started at Wed Feb 24 00:57:49 2010 ****
**** Software Version: 5.02.000.12538 (Feb 16 2010 20:14:13) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:

**** Namespace: wwmed
**** Volume:    /acct
**** Share:     bills2
**** IP Addr:   192.168.25.23
**** Export:    /exports/acct2
**** Options:
****            modify:                          yes
****            rename-directories:              yes
****            rename-non-mappable-directories: no
****            rename-files:                    yes
****            sync-attributes:                 yes
****            skip-managed-check:              no
****            import protection:               yes
****            import treewalk-threads:         8
****            import priority:                 65535 (Lowest)

**** NOTE: Since both sync-attributes and rename-directories were specified,
****       only directories that collide with existing filenames will be
****       renamed.  Directories with colliding attributes will have their
****       attributes synchronized to the namespace.

**** Ignore List:
****            .snapshot


Share                Physical Filer
-------------------  ------------------------------------------------------------
[bills2           ]  192.168.25.23:/exports/acct2

**** LEGEND
****
****  Actions
****   R   : Entry renamed.
****   S   : Directory attributes synchronized.
****   ?   : Indicates additional settings or operations required for full
****         operation.  See specific issues for SD and IN below.
****   !   : Unable to resolve given specified options or configuration.
****
****  Entry Type
****   F   : Entry is a file.
****   D   : Entry is a directory.
****
****  Issue
```

```
****   NC  : Name collision.
****   AC  : Attribute collision.
****   RA  : Attributes of share root are inconsistent.
****   MG  : Subdirectory of this share is already imported as managed share.
****   CC  : Case-blind collision (MPNS and CIFS-only).
****   MC  : Maximum name collisions for this name in a directory
****   ER  : Entry removed directly from filer during import.
****   AE  : Error accessing entry.
****   RV  : Reserved name on filer not imported.
****   DF  : DFS link found during import.
****   TC  : Trailing character: space and period are not supported by all filer vendors.
****   HL  : Exceeded limit of 1024 hard links to a file.
****   @@  : Other error.


Import Scan:
================================================================================

------------------------------------------------------------
Import Scan Start Time:   Wed Feb 24 00:57:51 2010
------------------------------------------------------------

Type      Path
--------  ------------------------------------------------------------------------
[S D AC]  /payable/
       Differing Attributes: NFS Mode Bits
[R F NC]  /index.html -> index_bills2-3.html
[R F NC]  /payable/FrontMatter.fm -> FrontMatter_bills2-3.fm
[R F NC]  /payable/variables.fm -> variables_bills2-3.fm

Directories found:                      31
Files found:                           384
Directories Scanned/Second:             31
Files Scanned/Second:                  384
Total Entries Scanned/Second:          415

------------------------------------------------------------
Import Scan Stop Time:    Wed Feb 24 00:57:52 2010
Import Scan Elapsed Time: 00:00:01
------------------------------------------------------------

Directories imported by scan:                                    31
Directories imported dynamically:                                 0
Files imported by scan:                                         384
Files imported dynamically:                                       0
Directories renamed due to name/attribute conflict:               0
Files renamed due to name conflict:                               3
Directory attributes synchronized:                                1

**** Elapsed time:           00:00:03
**** Share Import Report: DONE at Wed Feb 24 00:57:52 2010 ****
```

Correcting the Collisions

To rename a duplicate file, access the virtual IP (VIP) for the volume as a client would, and rename or move the file as desired.

Managing Collisions With CIFS 8.3 Names

Volumes that support CIFS have an additional possible naming-collision obstacle, caused by back-end servers keeping an extra name for some files and directories. Windows once ran on file systems (such as FAT12 and FAT16) that supported only short file names, sometimes called *8.3 names*. An 8.3 name uses the following format:

base-name[*.ext*]

where

- ***base-name*** is 1-8 characters,
- **.** is optional, and
- ***ext*** is 1-3 characters.

Newer Windows servers run NTFS, which supports file and directory names of any length. Other file systems that support CIFS also support names of any length. For backwards compatibility, these contemporary file systems continue to support an *alternate* 8.3 name for any file or directory that does not fit the above pattern. The alternate name is sometimes called a filer-generated name (*FGN*) since it is created by the filer. The longer name is called the *primary* name. Contemporary Windows clients see the primary name while older Windows applications see only the alternate name, the 8.3 FGN. Contemporary applications can use either name to access the file or directory, though, by default, they do not see the alternate name.

The 8.3-FGN Pattern

Filers generate alternate 8.3 names with specific patterns, usually including a tilde (~) character to indicate that the name is filer-generated. Any 8.3 FGN falls into one of two patterns:

name-with-tilde[*.ext*]

is the most-common pattern, where

- ***name-with-tilde*** (2-8 characters) has one tilde (~) character (such as “RANDOM~2”), and
- ***ext*** is optional and has 1-3 characters.

name-without-tilde[*.ext*]

is a pattern that EMC file servers use in some cases, where

- ***name-without-tilde*** (exactly 8 characters) has no tilde (~) character (such as “0000D773”), and
- ***ext*** is optional and has 1-3 characters.

8.3 FGNs Do Not Exist in an ARX Volume

Any 8.3 FGNs that appear when you access the filer directly (such as "KMO_ME~1.DAT") do not appear in the client view of the managed volume. The managed volume keeps only the primary name for the file ("kmo_medical_record.dat" in this example); files and directories are not accessible by their filer's alternate names.

Best Practice: Avoid Primary Names That Match The 8.3 Pattern

F5 Networks recommends strongly that you disable the creation of 8.3 FGNs on your back-end filers. There is no penalty for disabling this, since the 8.3 FGNs are not accessible through the ARX, anyway. Windows and Samba both are able to disable altname generation, and Microsoft itself recommends the practice.

If a primary name matches the pattern for an 8.3 FGN, it collides with any matching alternate name on another share. For example, consider a file on share A with a primary name of "MYFILE~1" and a different file on share B named "myFileForYouToRead." If the latter file has an alternate name of "MYFILE~1," it collides with the file on share A. For the same reason, migration of such a file from share A to B will fail.

8.3 FGN Collision Avoidance

If a client attempts to create a file that matches an 8.3 FGN on a back-end filer, the ARX will execute collision prevention measures on the target share automatically. If this is successful, the file creation request will be processed normally, and the result from the file server will be returned to the client. If the collision prevention behavior fails, a relevant error code will be returned instead.

If an 8.3 FGN collision is recognized on the back-end filer, the ARX will:

- Attempt to clear the colliding 8.3 FGN on the back-end filer. This works only on newer versions of Windows filers. The primary file name will remain the same.
- Change the colliding 8.3 FGN explicitly by instructing the filer to change it. (This works only on Windows.)
- Change the 8.3 FGN implicitly by renaming the colliding file (thus indirectly causing the file server to change its 8.3 FGN). (This works on all filers.) This change is temporary and is never visible via the ARX VIP.

The ARX metadata is not affected by any attempt to clear or change the 8.3 FGN.

Identifying 8.3 FGN Collisions

Run an nsck-inconsistencies report (recall *Finding Metadata Inconsistencies*, on page 7-16) to identify 8.3 name collisions. Each file with this issue is flagged with "F8," and each directory is flagged with "D8."

For example, this command sequence generates and shows an inconsistencies report for the "medarcv" namespace, which reveals one file collision and one directory collision (highlighted in bold text):

```
bstnA# nsck medarcv report inconsistencies /rcrds outputfile rcrdsIssues
Scheduling report: rcrdsIssues.rpt on switch bstnA
bstnA# show reports rcrdsIssues.rpt
**** Inconsistencies Report: Started at Wed Feb 24 01:58:00 2010 ****
**** Software Version: 5.02.000.12538 (Feb 16 2010 20:14:13) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:
**** Namespace: medarcv
**** Volume: /rcrds
**** Path: /rcrds

Share                Physical Filer
-------------------  ------------------------------------------------------------
[rx                ]  192.168.25.29:prescriptions
[charts            ]  192.168.25.20:histories
[bulk              ]  192.168.25.27:bulkstorage

**** Legend:
****   LF = File exists in the metadata, but is missing from the physical filer.
****   LD = Directory exists in the metadata, but is missing from the physical filer.
****   FF = File exists on the physical filer, but is missing from the metadata.
****   FD = Directory exists on the physical filer, but is missing from the metadata.
****   LL = File is a symlink in the metadata, but is a regular file on the filer.
****   FL = File is a symlink on the filer, but is a regular file in the metadata.
****   IF = Filehandles in the metadata do not match the filehandles on the physical filer.
****   MF = The file is currently being migrated.
****   NL = Unable to lock parent directory during report.
****   FE = Error contacting filer during report.
****   FO = Filer Offline: The filer is offline or disabled.
****   F8 = A file name matches a CIFS alternate "8.3" name on another share.
****   D8 = A directory name matches a CIFS alternate "8.3" name; its contents will be skipped.
****   DC = A client has the file or directory open for delete-on-close, but the filer has already
deleted it.
****   SD = Striped leaf directory found on filer, expected on other shares.
****   SL = File is a symlink.
****   UT = Name contains characters that are invalid UTF-8; must solve issue directly on the filer

Type                     Share                Path
-----------------------  -------------------  ------------------------------
[LF                    ]  [charts            ]  /copyRandomx.exe
[   F8                 ]  [charts            ]  /KMO_ME~1.DAT -> kmo_medical_record.dat
[   D8                 ]  [charts            ]  /RECORD~1/ -> records_predating_y2k

**** Total Found Items:                  0
**** Total Lost Items:                   1
**** Total Invalid Filehandles:          0
**** Total Migrating Files:              0
**** Total Deleted Before Close:         0
**** Total Locking Errors:               0
**** Total Filer Errors:                 0
**** Total 8.3 Errors:                   2
**** Total Found Stripes:                0
```

```
**** Total processed:             188
**** Elapsed time:           00:00:00
**** Inconsistencies Report: DONE at Wed Feb 24 01:58:00 2010 ****
bstnA# ...
```

Correcting 8.3 Name Collisions Manually

In the event that the ARX's 8.3 name collision avoidance behavior does not resolve an 8.3 name collision automatically, you can use the same method to correct 8.3 name collisions that you use with standard collisions: access the VIP as a client and rename the offending file or directory. The entry to rename is the one with a primary name that matches the 8.3-FGN pattern. This is the first name shown in the inconsistencies report ("KMO_ME~1.DAT" and "RECORD~1" in the above example).

Fixing File/Directory Names with Trailing Periods

This section applies only to managed volumes that support CIFS. You can skip this section if you are using NFS-only volumes and/or direct volumes.

CIFS-filer vendors are inconsistent in their handling of two illegal trailing characters in file/directory names, period (.) and space ( ). Some CIFS implementations strip the trailing character from the end of the name at creation time, some reject the name with an error, and others replace any trailing-period name with an FGN. This inconsistent handling poses a problem for a volume backed by multiple CIFS vendors: a file on one back-end share may have its name changed to an FGN if it migrates to another. To avoid this situation, a CIFS-only volume identifies filers that create FGNs for trailing-period names, and does not allow clients to create these names on those filers.

CIFS-only volumes use special processing for trailing-period names. (The period, not the space, is the trailing character that triggers an FGN on some filers.) The following subsections describe this special processing.

◆ Note

This special processing occurs only in volumes where at least one share converts trailing-period names into FGNs.

Illegal Client Operations

CIFS-only volumes prevent the following client operations if (and only if) they are backed by any filer that converts trailing-period names to FGNs:

- Clients cannot create any new file or directory with a trailing period in these volumes.
- Clients cannot rename an existing file or directory so that the new name has a trailing period.
- The volume also blocks some client operations *inside* a directory with a trailing-period name. For example, creating the file, “\myvol\mydir.\newFile.txt”, could fail because of the period at the end of the parent directory’s name, “\mydir.”. The volume would block this if it meant that the volume would have to replicate “\mydir.” on a filer that would change it into an FGN.

The final case is subtle, and may require a search through the syslog file (recall *Accessing the Syslog*, on page 8-7) to prove that a trailing-period is at fault. Search for the string “TRAIL” for syslog messages about illegal trailing periods. Use grep TRAIL logs syslog to search for messages about trailing periods, or grep *ip-address* logs syslog to find all messages concerning a client at *ip-address*.

Finding Imported Files with Illegal Trailing Characters

Files or directories with trailing-period names may have existed before import. These files and directories are flagged with the letters “TC” (for Trailing Character) in a CIFS share’s import report. For example, this import report finds two directories and one file that end with a period. The file and directories are highlighted below in bold text:

```
bstnA# show reports import.6.charts.12.rpt
**** Share Import Report: Started at Wed Feb 24 00:59:42 2010 ****
**** Software Version: 5.02.000.12538 (Feb 16 2010 20:14:13) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:

**** Namespace: medarcv
**** Volume:    /rcrds
**** Share:     charts
**** IP Addr:   192.168.25.20
**** Export:    histories
**** Options:
****            modify:                          yes
****            rename-directories:              yes
****            rename-non-mappable-directories: no
****            rename-files:                    yes
****            sync-attributes:                 yes
****            skip-managed-check:              no
****            import protection:               yes
****            import treewalk-threads:         8
****            import priority:                 65535 (Lowest)

**** NOTE: Since both sync-attributes and rename-directories were specified,
****       only directories that collide with existing filenames will be
****       renamed.  Directories with colliding attributes will have their
****       attributes synchronized to the namespace.

**** Ignore List:
****            .snapshot


Share                Physical Filer
-------------------  ------------------------------------------------------------
[charts            ]  192.168.25.20:histories

**** LEGEND
****
****  Actions
****   R   : Entry renamed.
****   S   : Directory attributes synchronized.
****   ?   : Indicates additional settings or operations required for full
****         operation.  See specific issues for SD and IN below.
****   !   : Unable to resolve given specified options or configuration.
****
****  Entry Type
****   F   : Entry is a file.
****   D   : Entry is a directory.
****
****  Issue
****   NC  : Name collision.
****   AC  : Attribute collision.
****   RA  : Attributes of share root are inconsistent.
****   MG  : Subdirectory of this share is already imported as managed share.
```

```
****   CC  : Case-blind collision (MPNS and CIFS-only).
****   MC  : Maximum name collisions for this name in a directory
****   ER  : Entry removed directly from filer during import.
****   AE  : Error accessing entry.
****   RV  : Reserved name on filer not imported.
****   DF  : DFS link found during import.
****   TC  : Trailing character: space and period are not supported by all filer vendors.
****   HL  : Exceeded limit of 1024 hard links to a file.
****   @@  : Other error.


Import Scan:
================================================================================

------------------------------------------------------------
Import Scan Start Time:   Wed Feb 24 00:59:48 2010
------------------------------------------------------------

Type      Path
--------  ------------------------------------------------------------------------
[  D TC]  /2007/late_test_results.
[  D TC]  /2004/planR.
[  F TC]  /2007/hlo_medical_record.dat.

Directories found:                      19
Files found:                           141
Directories Scanned/Second:             19
Files Scanned/Second:                  141
Total Entries Scanned/Second:          160

------------------------------------------------------------
Import Scan Stop Time:    Wed Feb 24 00:59:48 2010
Import Scan Elapsed Time: 00:00:01
------------------------------------------------------------

Directories imported by scan:                                  19
Directories imported dynamically:                               0
Files imported by scan:                                       141
Files imported dynamically:                                     0
Directories renamed due to name/attribute conflict:             0
Files renamed due to name conflict:                             0
Directory attributes synchronized:                              0

**** Elapsed time:          00:00:06
**** Share Import Report: DONE at Wed Feb 24 00:59:48 2010 ****
bstnA# ...
```

Finding Shares That Convert Trailing-Period Names to FGNs

You can use the show namespace command to find if a particular share converts trailing-period names into FGNs: the no-trail-period flag appears in the share's Features field.

For example, this command shows that the "swic~/users~samba1" share performs the FGN conversion:

```
minturnA# show namespace swic volume /users share samba1

Namespace "swic" Configuration
Description: (none)
```

```
Metadata Cache Size: 512 MB
Proxy User: swi_admin
Filer SMB Signatures: Enabled
SAM Reference Filer: win-1 (192.168.81.30)

Supported Protocols
-------------------
  cifs

CIFS Authentication
-------------------
Protocols:
  NTLM
  NTLMv2
  Anonymous (IPC$ only)

Participating Switches
----------------------
  bstnA (Volume Group 1) [Current Switch]


Windows Management Authorization Policies
-----------------------------------------
  mmcAdmin

Volumes
-------
  /users
    CIFS : compressed files: yes; named streams: yes; persistent ACLs: yes
           sparse files: yes; Unicode on disk: yes; case sensitive: no

             Volume freespace: 10 GB (automatic)
           Volume total space: 19 GB
                  CIFS quotas: Not Enabled
              Auto Sync Files: Enabled
                Metadata size: 104 kB
         Metadata free space: 46 GB
             Filer Subshares: Enabled
              Oplock support: Enabled
          Notify-change mode: Normal
             CIFS path cache: Enabled
      CIFS access based enum: Not Enabled
                  Snapshots: Not Enabled
            Migration method: Staged
                       State: Enabled

                 Host Switch: minturnA
                    Instance: 1
                Volume Group: 1
                   Processor: 1.1
                       Files: 70 used (35 dirs), 3.9 M free, 252 M max (automatic)
    Metadata shares:

      Filer          Backend Path           Contains Metadata  Status
      ------------------------------------------------------------------
      swi-nas        /vol/datavol1/exports/aco_meta Yes               Online

    Share samba1
      Description                home dirs for Windows users
      Filer                      smb-1 [192.168.81.3]
      CIFS Share                 wusers
      Features                   cifs-acls cifs-case-blind no-trail-period
```

```
      SID Translation             No
      Ignore SID errors           No
      Status                      Online
      Volume Root Backing         Yes
      Import Sync Attributes      Yes
      Import Skip Managed Check   Yes
      Import Priority             65535 (Lowest)
      Free space on storage       10 GB (11,700,883,456 B)
      Total space on storage      19 GB (21,472,735,232 B)
      Policy Maintain Freespace:  1.0 GB
      Policy Resume Freespace:    2.0 GB
      Transitions                 1
      Last Transition             Mon 23 Aug 2010 03:32:14 AM EDT

minturnA# ...
```

Disallowed Migrations and Replications

A managed volume never migrates a file or directory with a trailing period to a filer that would convert it to an FGN. Specifically, volumes avoid the following autonomous operations:

- file migrations through file-placement rules (see *Placing Files on Particular Shares*, on page 14-4 of the *ARX® CLI Storage-Management Guide*),
- directory striping (to facilitate a file migration), or
- file or directory replications to a shadow volume (see *Configuring a Shadow-Copy Rule (Source Switch)*, on page 16-9 of the same manual).

Stopped migrations and replications are enumerated in file-placement reports and shadow-copy reports.

Migrating Storage Between Filer Vendors

Some sites use the ARX to move storage from filer to filer. If the source filer allows these illegal trailing characters but the destination filer does not, you must manually change these names before the migration. The best practice for this migration varies from site to site. You can contact F5 Support to help choose the best method for your site.

Finding NFS-Only Entries in a Multi-Protocol Volume

This section applies to volumes in a multi-protocol (CIFS and NFS) namespace only, where the CIFS name and the NFS name for a file may differ.

Some file/directory names are legal in NFS but illegal in CIFS, or are legal in CIFS but illegal in NFS. Most multi-protocol filers use the original name for the clients of one protocol, then create a new filer-generated name (FGN) for the clients of the other protocol. The ARX aggregates the naming solutions from several multi-protocol vendors, each following a different standard for FGNs. Every file or directory must have an NFS-side name that matches its CIFS-side name *on every share*, or it is marked as *NFS-only*.

Best Practice: Avoid NFS-Only Entries Where Practical

CIFS clients cannot access an NFS-only entry. This is especially a problem for directories, which obscure all of their contents including all child directories.

Performance Issues

NFS-only entries, under rare circumstances, can also impede volume performance. The volume does not record any FGNs from its filers; instead, it probes for FGNs *only when* a collision with an FGN is possible. An FGN collision can only occur in a directory that has one of two problems: either it contains NFS-only entries, or it contains a name that matches an FGN pattern (like “FILE~2.txt”).

A directory with any NFS-only entries is certain to have at least one FGN on its filers. If a client creates a name there that might match an FGN pattern (such as “myfile~1.doc” or “dir~1”), the volume first probes all its back-end shares to verify that the FGN is not already taken. If the FGN is taken, the client’s new file is marked “NFS-only.”

A directory with any FGN-patterned names introduces the opposite problem: a new NFS-only entry can create a back-end FGN that collides with an existing entry. In this case, the pre-existing entry with the FGN-patterned name becomes “NFS-only.” For example, if “MYFILE~1” already exists in a directory and a client creates “myFile?,” the volume must probe the back-end shares to see if any of them created a “MYFILE~1” FGN. If the FGN is found, the original “MYFILE~1” entry becomes “NFS-only.”

Evidence from the field demonstrates that these collision checks are rarely necessary, though they do affect performance when they occur.

The policy engine is subject to this collision check like any external client.

Identifying Problematic Entries

If CIFS clients complain about access issues or slow performance, you should identify the NFS-only entries in the volume and work toward changing their names. This is especially true for directory names, which can obscure multiple files and directories from CIFS users. Run an nsck-inconsistencies report with the multi-protocol flag (recall *Focusing on Multi-Protocol Issues*, on page 7-18) to reveal all NFS-only file names and directory names. Each of them is flagged with an "NF." For example, this command sequence generates and shows an inconsistencies report for the "insur" namespace, which reveals several NFS-only files and directories:

```
bstnA# nsck insur report inconsistencies multi-protocol outputfile insur_fgns
Scheduling report: insur_fgns._claims.rpt on switch bstnA
bstnA# show reports insur_fgns._claims.rpt
**** Inconsistencies Report: Started at Wed Feb 24 01:58:49 2010 ****
**** Software Version: 5.02.000.12538 (Feb 16 2010 20:14:13) [nbuilds]
**** Hardware Platform: ARX-4000
**** Report Destination:
**** Namespace: insur
**** Volume: /claims
**** Path: /claims
**** multiprotocol flag specified

Share                Physical Filer
-------------------  ------------------------------------------------------------
[shr1-old          ]  192.168.25.21 NFS:/vol/vol1/NTFS_QTREE/insurance, CIFS:insurance
[shr1-next         ]  192.168.25.51 NFS:/root_vdm_4/patient_records, CIFS:patient_records

**** Legend:
****   LF = File exists in the metadata, but is missing from the physical filer.
****   LD = Directory exists in the metadata, but is missing from the physical filer.
****   FF = File exists on the physical filer, but is missing from the metadata.
****   FD = Directory exists on the physical filer, but is missing from the metadata.
****   LL = File is a symlink in the metadata, but is a regular file on the filer.
****   FL = File is a symlink on the filer, but is a regular file in the metadata.
****   IF = Filehandles in the metadata do not match the filehandles on the physical filer.
****   MF = The file is currently being migrated.
****   NL = Unable to lock parent directory during report.
****   FE = Error contacting filer during report.
****   FO = Filer Offline: The filer is offline or disabled.
****   F8 = A file name matches a CIFS alternate "8.3" name on another share.
****   D8 = A directory name matches a CIFS alternate "8.3" name; its contents will be skipped.
****   DC = A client has the file or directory open for delete-on-close, but the filer has already
deleted it.
****   SD = Striped leaf directory found on filer, expected on other shares.
****   SL = File is a symlink.
****   UT = Name contains characters that are invalid UTF-8; must solve issue directly on the filer
****
****  Multi-Protocol Only:
****   NF = Name is accessible only to NFS clients.
****   NI = Name does not exist via both NFS and CIFS.
****   CC = Case-blind name collision.
****   IC = CIFS invalid characters found in NFS name.
****   FN = A portion of the name contains a filer-generated pattern.
****   NM = CIFS name has characters that are not mappable to the NFS encoding.
****   SP = A NFS name in the path's ancestry will prevent CIFS access.

Type                      Share                Path
------------------------  -------------------  -------------------------------
[              NF CC   ]  [shr1-old          ]  /INDEX.html
```

```
[               NF CC   ]  [shr1-old          ]  /index.html
[               NF CC   ]  [shr1-old          ]  /Tools
[               NF CC   ]  [shr1-old          ]  /tools
[               NF IC   ]  [shr1-old          ]  /Claims:2001
[               NF IC   ]  [shr1-old          ]  /claims:2005
[               NF IC   ]  [shr1-old          ]  /draft_proposals.
[   F8                  ]  [shr1-old          ]  /OVERDU~1.DOC -> overdueclaimsmemo.doc
[   D8                  ]  [shr1-old          ]  /Y2KCLA~1/ -> y2kclaims
[                     SP]  [shr1-old          ]  /draft_proposals./FY06challenges.ppt
[                     SP]  [shr1-old          ]  /draft_proposals./claim2389_rebut2.doc
[                     SP]  [shr1-old          ]  /draft_proposals./FY2007propB.ppt
[                     SP]  [shr1-old          ]  /draft_proposals./claim2389_rebut1.doc
[               NF IC   ]  [shr1-old          ]  /images/FRU replace wrongway.tif
[               NF IC   ]  [shr1-old          ]  /images/:D1JL100
[               NI NM   ]  [shr1-old          ]  /images/file012bÄ«/ (Characters: U+012b)
[               NF CC   ]  [shr1-old          ]  /stats/piechart.ppt
[               NF CC   ]  [shr1-old          ]  /stats/PieChart.ppt
[               NF IC   ]  [shr1-old          ]  /stats/on_the_job:2004.cnv
[               NF IC   ]  [shr1-old          ]  /stats/on_the_job:2003.cnv
[               NF IC   ]  [shr1-old          ]  /stats/in_home:2005
[                     SP]  [shr1-old          ]  /tools/extractCdDocs.pl
[                     SP]  [shr1-old          ]  /tools/cleanBU.csh
[                     SP]  [shr1-old          ]  /tools/checkinAll.pl
[                     SP]  [shr1-old          ]  /tools/docSet.pl
[                     SP]  [shr1-old          ]  /tools/updateCvs.csh
[                     SP]  [shr1-old          ]  /tools/books.xml
[                     SP]  [shr1-old          ]  /tools/blowAway.pl
[                     SP]  [shr1-old          ]  /tools/makeCd.pl
[               NF IC   ]  [shr1-old          ]  /stats/in_home:2005/age:11-21yrs.csv
[               NF IC   ]  [shr1-old          ]  /stats/in_home:2005/age:>21yrs.csv
[               NF IC   ]  [shr1-old          ]  /stats/in_home:2005/age:<10yrs.csv
[               NF IC   ]  [shr1-old          ]  /stats/in_home:2005/age:>21yrs.csv
[               NF IC   ]  [shr1-old          ]  /stats/in_home:2005/age:<10yrs.csv
[               NF IC   ]  [shr1-old          ]  /stats/in_home:2005/age:11-21yrs.csv
[                     SP]  [shr1-old          ]  /stats/in_home:2005/age:<10yrs.csv
[                     SP]  [shr1-old          ]  /stats/in_home:2005/age:>21yrs.csv
[                     SP]  [shr1-old          ]  /stats/in_home:2005/age:11-21yrs.csv

**** Total Found Items:                   0
**** Total Lost Items:                    0
**** Total Invalid Filehandles:           0
**** Total Migrating Files:               0
**** Total Deleted Before Close:          0
**** Total Locking Errors:                0
**** Total Filer Errors:                  0
**** Total 8.3 Errors:                    2
**** Total Found Stripes:                 0

**** Total processed:             171
**** Elapsed time:           00:00:01
**** Inconsistencies Report: DONE at Wed Feb 24 01:58:50 2010 ****
```

The remaining subsections describe the various types of NFS-only files and directories, and discuss methods for finding and fixing them.

Case-Blind Collisions (CC)

A *case-blind collision* occurs when two or more entries in the same parent directory have names that differ only in case (for example, “index.html” and “INDEX.html”). CIFS does not support case collisions, but NFS does. These are created by NFS clients, either before import or while the volume is in service.

A file with a case-blind collision is marked with a “CC” in the inconsistencies report. For example (from the inconsistencies report above):

```
Type                      Share                Path
------------------------  -------------------  ------------------------------
[              NF CC   ]  [shr1-old          ]  /index.html
[              NF CC   ]  [shr1-old          ]  /INDEX.html
...
```

Note

If a directory contains more than 1,024 case collisions, its parent directory incurs an additional performance penalty if it checks for FGN collisions during metadata-intensive operations (share import, restore data*, all* nsck *operations, and* sync*). The FGN-collision check was described in* *Performance Issues, on page 10-64.*

Names with Characters that are Illegal in CIFS (IC)

The following characters are legal in NFS file names but illegal in their CIFS counterparts: any control character, /, \, :, *, ?, >, <, ”, or |. Additionally, a trailing period (.) or a trailing space character is illegal in a CIFS name; recall *Fixing File/Directory Names with Trailing Periods*, on page 10-59. Files and directories with any of these illegal CIFS characters are NFS-only. Like case collisions, these are created by NFS clients.

Each entry with illegal CIFS characters is flagged with an “IC” (for “Illegal Character”) in the inconsistencies report. For example, this file is NFS-only because of the colon (:) in its name:

```
Type                      Share                Path
------------------------  -------------------  ------------------------------
...
[              NF IC   ]  [shr1-old          ]  /stats/on_the_job:2004.cnv
```

Client View: ARX-Generated Names

A managed volume creates its own ARX-generated names (AGNs) for files or directories with illegal CIFS characters. The volume presents the AGN to its CIFS clients. The AGN has a unique format, easily distinguishable from the formats of any FGNs:

(!) *nnn_changed-file-name*

Where

"**(!)** " (with the trailing space) is the prefix for all AGNs.

nnn is a different number for every AGN in this directory. This number is padded so that it is always at least three digits long.

changed-file-name is the original file name (or directory name), with all illegal characters replaced with an underscore character (_).

For example, the above NFS-side file name ("on_the_job:2004.csv") may have the following AGN for CIFS clients:

(!) 001_on_the_job_2004.cnv

The ARX only generates CIFS-side AGNs. NFS-side AGNs are not possible in the current implementation.

Unicode Characters in CIFS Names: Non-Mappable in NFS (NM)

Some illegal file or directory names may have been created by CIFS clients before import. CIFS natively supports all Unicode characters in its file or directory names, but NFS servers must specifically have their character encoding configured to support them. Otherwise, many Unicode characters are *non-mappable* to NFS. A multi-protocol filer set for Latin1 character encoding may have allowed CIFS clients to use Unicode characters (such as Korean or Japanese characters) in entry names. If so, the filer accepted the CIFS name and created an FGN for its NFS clients.

CIFS clients cannot create this problem through the volume; the volume only allows CIFS-side names that are fully mappable to NFS. This is an import-only issue.

When the volume imports the share, it treats non-mappable names differently for files and directories.

Non-Mappable File Names

For files, the volume uses the NFS-side FGN for the file name. If the FGN contains any illegal CIFS characters (as they often do), the volume also creates a CIFS-side AGN from that FGN. For example, consider a site that supports Latin1, a single-byte character encoding, for its NFS clients. Suppose (before import) a CIFS user creates a file with the following name:

file012b*C*

where *C* is a character that is not supported by the filer's NFS character encoding.

The filer may create an NFS-side FGN with no apparent relationship to the original file name:

:I1A3410

The volume imports this file and uses the above name for its NFS clients. For its CIFS clients, it creates the following AGN (since the colon character cannot be used in a CIFS file name):

(!) 001__I1A3410

An NSCK inconsistencies report for the volume shows these files with an "NM" flag. For example:

```
Type                      Share                Path
------------------------  -------------------  ------------------------------
...
[              NI NM   ]  [shr1-old          ]  /images/file012bÄ«/ (Characters: U+012b)
```

The CIFS-side AGN is inaccessible, but an NFS user can change the NFS-side name (":I1A3410" in the above example) to something that is legal and recognizable in both protocols (for example, "file012").

Non-Mappable Directory Names

A directory may contain a large tree of subdirectories and files, so the import process takes greater care in keeping it accessible and easily recognizable from both CIFS and NFS. If the volume is allowed to modify its imported shares and rename directories in the particular share, it renames the directory based on its CIFS name. The import for the share fails if the volume is not allowed to both modify and rename directories; this is discussed in *Preventing Directory Renames During Import*, on page 9-38 of the *ARX® CLI Storage-Management Guide*.

The directory renames follow the pattern below. The volume replaces each unmappable CIFS character with its numeric equivalent:

new-dirname_share-jobid[*-index*][*.ext*]

where *new-dirname* is the original CIFS-side name, but with "(U+*nnnn*)" in place of each unmappable character. The *nnnn* is the Unicode number for the character, shown in hexadecimal format. The name is truncated if it exceeds 256 characters. For example, "dir(U+30D2)(U+30AA)_myshare-2."

The share's import report includes the original name and the new name for each renamed directory. The directory does not appear in the inconsistencies report, as its new name is legal in both protocols.

Split Directories (SP)

Any NFS-only directory is called a *split directory*. All descendents of split directories are flagged with an "SP" in the inconsistencies report; they are flagged because they are inaccessible to CIFS clients. For example, here is a split directory and its descendents:

```
Type                      Share                Path
------------------------  -------------------  ------------------------------
[              NF CC   ]  [shr1-old          ]  /tools
```

```
...
[                    SP]  [shr1-old          ]  /tools/extractCdDocs.pl
[                    SP]  [shr1-old          ]  /tools/cleanBU.csh
...
```

"Marked" Directories

Some multi-protocol directories are *marked* in the metadata for FGN-collision checking. A directory is marked for this processing if it contains entries with either of the following name issues:

- NFS-only (such as "my?dir," or "MYDIR" and "mydir" together), and/or
- matching an FGN pattern (such as "my~dir").

If the directory contains either of these name types, the volume must check for FGN collisions every time a client creates a name of the *other* type. The FGN-collision check was described earlier, in *Performance Issues*, on page 10-64. This check occurs for all types of clients, including the policy engine; that is, the volume also performs extra checking before *migrating* any FGN-related files or subdirectories from one back-end share to another.

A directory remains "marked" for this checking even if clients remove all NFS-only and FGN-matching entries from it. You can remove the mark by emptying the directory completely, or by cleaning out all entries with FGN issues and running sync files on it. The latter option is described later in this section.

Preventative Striping

Another disadvantage for a marked directory can be extra back-end copies (called *stripes*) for some of its NFS-only subdirectories. Before the policy engine migrates any NFS-only file (such as "newFile" in a directory that already contains "NEWFILE"), it stripes all of its subdirectories that fit an FGN pattern (such as "DIR~3") to the target share. Then it migrates the file. This ensures that the subdirectories remain open to both CIFS and NFS. It also creates extra, unexpected directory stripes on the back-end target share.

Correcting NFS-Only Entries

To remove the "NFS-only" flag from a file or directory, an NFS client can access the volume's front-end service and rename the offending entry from there. Choose a name that is valid in both protocols. For example, here are some NFS-only issues and possible solutions:

- Case collision ("index.html" and "INDEX.HTML" in the same directory)

  Rename "INDEX.HTML" to "index2.html."
- Illegal CIFS characters ("on_the_job:2004.csv")

  Rename to "on_the_job_2004.csv," to eliminate the illegal colon (:).
- Name with an FGN pattern ("OVERDU~1.DOC") that collides with an actual FGN.

Rename to “overdue_1.doc” to remove the tilde (~).

You can correct a marked directory by correcting all of the NFS-only entries within it.

After the rename(s), you must run the sync files ... recurse command to remove the NFS-only flags from the volume’s renamed files and directories. The sync utility probes all of the volume’s shares to confirm that no FGN collisions exist. This also removes the marks from any directories that no longer contain any NFS-only entries. The sync files command was described in an earlier chapter: recall *Synchronizing Metadata with Actual Files*, on page 7-22.

For example, the following command sequence exits to priv-exec mode and synchronizes the metadata in the “insur~/claims” volume:

```
bstnA(gbl)# end
bstnA# sync files insur volume /claims path / recurse
Scheduling sync files operation on switch bstnA, report name: sync.7._rcrds.rpt
bstnA# ...
```

Recursively Correcting Split Directories

In any split directories that have been corrected, the sync utility probes all child entries, including child directories, for the first time. This may expose more NFS-only entries and/or FGN-patterned names. If so, repeat the renaming and syncing process as needed.

Running Periodic Checks for NFS-Only Entries

Best practices dictate that you occasionally run an inconsistencies report on your multi-protocol volumes to monitor for NFS-only entries. Clients can create these by using bad characters in file/directory names, making a pair of entries with a case-collision, or by making an entry name that happens to match an FGN on one of the volume’s back-end filers. As described above, once a directory contains any NFS-only or FGN-patterned entry, it performs extra processing for certain (rare) client operations.

Migrations in a Multi-Protocol Namespace

Migrations in a multi-protocol (NFS and CIFS) volume have subtle side effects. These side effects compensate for limitations in back-end filers; clients are unaffected in most cases, but this section is offered so that you are aware of any potential issues.

Skip to the next section if your namespace is CIFS-only or NFS-only.

File-Attribute Migrations

All file migrations are concerned with file attributes in addition to the files themselves. *File attributes* are permission settings, the name or ID of the user who owns the file, the group or groups who have access to the file, last-modified times, named streams, and other external data associated with the file. File-attribute migrations require special semantics in a multi-protocol namespace (that is, a namespace whose shares support both NFS and CIFS access). If your namespace is NFS-only or CIFS-only, you can skip this section.

A multi-protocol namespace can be backed by a heterogeneous mix of multi-protocol filers (MPFs), possibly from multiple vendors. The ARX passes client requests to these filers, and passes filer responses back to the clients. File attributes, such as file ownership and permission settings, are managed by each filer.

NFS and CIFS have fundamentally different file attributes, and each MPF vendor has unique semantics for reconciling the two. An NFS client can change a file's attributes on Filer X, and Filer X uses its semantics to change the CIFS attributes accordingly. This translation is manifestly inexact, so an MPF from another vendor typically gets a slightly different CIFS translation for the same NFS attributes.

If the ARX migrates the filer from Filer X to Filer Y (made by two different vendors), it takes these different semantics into account. This ensures that the NFS and CIFS file attributes are preserved *as much as possible* as they are migrated from one vendor to another. In some cases, file attributes are interpreted differently on the destination filer. The sections below share the details of attribute migration.

Use these tables along with vendor documentation to determine any implications for your clients.

From a NetApp Filer, UNIX Qtree

The following table shows how file attributes are migrated from a NetApp filer with a UNIX-based Qtree:

Vendor for Destination Filer	NFS		CIFS	
	Permission Bits	UID, GID, time stamps, ...	DOS Attributes	Security Descriptor (SD)
NetApp, UNIX Qtree	copied, then reset *after* file is transferred through CIFS	copied	copied (as dictated by destination Qtree)	derived by destination (NetApp) filer
NetApp, UNIX Qtree with cifs.preserve_unix_security				
NetApp, NTFS Qtree	Migrates between these two types of Qtrees are not supported.			
Samba	copied		not copied	
EMC	all copied *after* SD		derived by NetApp, then copied (overriding any volume ACLs)	
EMC, MIXED				

From a NetApp Filer, UNIX Qtree with cifs.preserve_unix_security

NetApp's cifs.preserve_unix_security flag causes a NetApp volume to mimic its NFS file permissions with a special CIFS ACL. This special ACL is ignored when migrated to most filers, but it requires some special treatment when migrated from one NetApp volume to another. The following table shows how file attributes are migrated from a NetApp filer with a UNIX-based Qtree and this special attribute:

Vendor for Destination Filer	NFS		CIFS	
	Permission Bits	UID, GID, time stamps, ...	DOS Attributes	Security Descriptor (SD)
NetApp, UNIX Qtree	copied, then reset *after* file is transferred through CIFS	copied	copied (as dictated by destination Qtree)	derived by destination (NetApp) filer
NetApp, UNIX Qtree with cifs.preserve_unix_security	copied		copied *after* UNIX attributes	
NetApp, NTFS Qtree	Migrates between these two types of Qtrees are not supported.			
Samba	copied		not copied	
EMC	all copied *after* SD		derived by NetApp, then copied (overriding any volume ACLs)	
EMC, MIXED				

From a NetApp Filer, NTFS Qtree

The following table shows how file attributes are migrated from a NetApp filer with an NTFS-based Qtree:

Vendor for Destination Filer	NFS		CIFS	
	Permission Bits	UID, GID, time stamps, ...	DOS Attributes	Security Descriptor (SD)
NetApp, UNIX Qtree	Migrates between NTFS and UNIX Qtrees are not supported.			
NetApp, UNIX Qtree with cifs.preserve_unix_security				
NetApp, NTFS Qtree	only high (setuid, setgid, and sticky) bits copied; remaining attributes derived by destination (NetApp) filer.		copied	
Samba	Migrates from NetApp/NTFS to Samba (no ACLs) are not supported.			
EMC	copied, then reapplied after SD		copied after initial copy of UNIX attributes	
EMC, MIXED	copied		copied after UNIX attributes	

From an EMC File Server (MIXED or not)

The following table shows how file attributes are migrated from an EMC file server. These rules apply whether or not the source-EMC file server is MIXED mode (that is, whether or not the source file system has its "accesspolicy" set to "MIXED"):

Vendor for Destination Filer	NFS		CIFS	
	Permission Bits	UID, GID, time stamps, ...	DOS Attributes	Security Descriptor (SD)
NetApp, UNIX Qtree	copied		copied (as dictated by destination Qtree)	derived by destination (NetApp) filer
NetApp, UNIX Qtree with cifs.preserve_unix_security				
NetApp, NTFS Qtree	only high (setuid, setgid, and sticky) bits copied; remaining attributes derived by destination (NetApp) filer.		copied after initial copy of UNIX high bits	
Samba	copied		derived by destination (Samba) filer	
EMC	copied		copied *after* UNIX attributes	
EMC, MIXED				

From a Unix Filer with Samba

The following table shows how file attributes are migrated from an Samba filer that does not support CIFS ACLs:

Vendor for Destination Filer	NFS		CIFS	
	Permission Bits	UID, GID, time stamps, ...	DOS Attributes	Security Descriptor (SD)
NetApp, UNIX Qtree	copied		copied (as dictated by destination Qtree)	derived by destination (NetApp) filer
NetApp, UNIX Qtree with cifs.preserve_unix_security				
NetApp, NTFS Qtree	Migrates from Samba (no ACLs) to NetApp/NTFS are not supported.			
Samba	copied		derived by destination (Samba) filer	
EMC	copied		derived by destination (EMC) filer	
EMC, MIXED	copied, then reapplied *after* the file is transferred through CIFS		not copied; derived by destination (EMC) filer	

Showing Policy History for a Volume

From any mode, you can use the show policy history command to show all of the policy transactions for a given managed volume:

show policy history namespace *namespace* volume *vol-path*

where

namespace (1-30 characters) is the namespace, and

vol-path (1-1024 characters) identifies the volume to examine.

The output lists all policy-related events that occurred in the volume, in chronological order. For example, the following command sequence shows the policy history for the "wwmed~/acct" volume:

```
bstnA(gbl)# show policy history namespace wwmed volume /acct

Namespace:        wwmed
Volume:           /acct
Rule
---------------------------------------
/acct

Time                  Message
--------------------  ------------------------------------------------------------
2012-03-26T04:51:49   Rule [wwmed:/acct:docs2das8]: Created.
2012-03-26T04:51:49   Rule [wwmed:/acct:docs2das8]: Enabled.
2012-03-26T04:51:50   Rule [wwmed:/acct:fm1]: Created.
2012-03-26T04:51:50   The configuration change conflicted with another change. The operation will
retry.
2012-03-26T04:51:51   Rule [wwmed:/acct:fm1]: Enabled.
2012-03-26T04:52:00   Rule [wwmed:/acct:fm1]: The initial scan has been disabled for this rule.
2012-03-26T04:52:04   Rule [wwmed:/acct:fm1]: The initial scan has been disabled for this rule.
```

Showing Detailed Status for a Particular Rule

If you add the name of a rule or share-farm, detailed status for the object appears:

show policy history namespace *namespace* volume *vol-path* rule *rl-name*

where

namespace (1-30 characters) is the namespace,

vol-path (1-1024 characters) identifies the volume to examine, and

rl-name (1-1024 characters) is one rule or share farm.

The output shows detailed events for the given policy object. For example, the following command sequence shows the policy history for the "dos2das8" rule:

```
bstnA(gbl)# show policy history namespace wwmed volume /acct rule docs2das8

Namespace:        wwmed
Volume:           /acct
Rule
---------------------------------------
```

```
docs2das8

Time                 Message
-------------------  ------------------------------------------------------------
2009-04-22T04:55:44  Enabled.
2009-04-22T04:56:01  Starting scan.
2009-04-22T04:56:04  Scan complete.
bstnA(gbl)# ...
```

This example shows the history for the "fm1" share farm in the same volume:

```
bstnA(gbl)# show policy history namespace wwmed volume /acct rule fm1

Namespace:        wwmed
Volume:           /acct
Rule
---------------------------------------
fm1

Time                 Message
-------------------  ------------------------------------------------------------
2009-04-22T05:47:20  Enabled.
2009-04-22T05:47:53  The initial scan has been disabled for this rule.
bstnA(gbl)# ...
```

Showing Any or All Pending Migrations

Any files that are waiting to be migrated appear in the *policy queue*. Use the show policy queue command to see if any migrations are pending:

show policy queue namespace *namespace* volume *vol-path*

where

namespace (1-30 characters) is the namespace, and

vol-path (1-1024 characters) identifies the volume to examine.

The output shows a table with three rows per file. The first line has the virtual-file path (starting with the managed-volume path). The second line has the rule name and storage target. The third line summarizes the current status of the migrated file. If files appear in the queue for long, the files may be very large, or there may be a problem with the policy engine or filers.

For example, this shows several files in the migration queue:

```
bstnA# show policy queue namespace medarcv volume /rcrds
File
        Rule                  Target
Status
------------------------------------------------------------------------------
/rcrds/2000/planA/fd_roosevelt.dat (Migrate)
        dailyArchive          bulk
        In Progress (Auto-Close Enabled) (Client Access Denied)
/rcrds/2000/planA/jq_adams.dat (Migrate)
        dailyArchive          bulk
        In Progress (Auto-Close Enabled) (Client Access Denied)
/rcrds/2000/planA/s_adams.dat (Migrate)
        dailyArchive          bulk
        In Progress (Auto-Close Enabled) (Client Access Denied)
/rcrds/2000/planA/t_jefferson.dat (Migrate)
```

```
        dailyArchive          bulk
        In Progress (Auto-Close Enabled) (Client Access Denied)
/rcrds/2000/planA/t_roosevelt.dat (Migrate)
        dailyArchive          bulk
        In Progress (Auto-Close Enabled) (Client Access Denied)
/rcrds/2007/late_test_results./june.xls (Migrate)
        dailyArchive          bulk
        In Progress (Auto-Close Enabled) (Client Access Denied)
bstnA# ...
```

Showing Auto-Closed Files (CIFS)

This section only applies to files and rules in a volume that supports CIFS.

You can configure a file-placement rule to automatically close any open files and hold them closed until the file is migrated. This makes it possible for the rule to migrate the file without interference from client writes. This feature is discussed in of the *Automatically Closing All Open Files (CIFS)*, on page 14-14 of the *ARX® CLI Storage-Management Guide*.

To see all files that are currently held closed, add the auto-close option to the end of the show policy queue command:

show policy queue namespace *namespace* volume *vol-path* auto-close

The output is the same as above, with additional lines for each file that the volume is holding closed. For example:

```
bstnA# show policy queue namespace medarcv volume /rcrds auto-close
File
        Rule                  Target
Status
-------------------------------------------------------------------------------
/rcrds/flu_vaccine.list (Migrate)
        dailyArchive          bulk
        In Progress (Auto-Close Enabled)
/rcrds/gw_medical_record.dat (Migrate)
        dailyArchive          bulk
        In Progress (Auto-Close Enabled) (Client Access Denied)
/rcrds/hst_medical_record.dat (Migrate)
        dailyArchive          bulk
        In Progress (Auto-Close Enabled)
/rcrds/kmo_medical_record.dat (Migrate)
        dailyArchive          bulk
        In Progress (Auto-Close Enabled) (Client Access Denied)
/rcrds/KMO_ME~1.DAT (Migrate)
        dailyArchive          bulk
        In Progress (Auto-Close Enabled) (Client Access Denied)
bstnA# ...
```

Canceling All Pending Migrations

To cancel a volume's pending migrations, go to priv-exec mode and use the cancel migration command:

cancel migration *namespace* volume *vol-path*

where

namespace (1-30 characters) identifies the namespace, and

vol-path (1-1024 characters) is the managed volume with pending migrations.

A prompt requests confirmation before the CLI cancels the migrations; enter **yes** to confirm.

For example, this command cancels all queued migrations in the "insur~/claims" volume:

```
bstnA(gbl)# end
bstnA# cancel migration insur volume /claims
Are you sure you want to cancel all the queued migrates in volume
'/claims'? [yes/no] yes
bstnA# ...
```

Canceling the Migration of a Single File

You can identify a single file to cancel its migration:

cancel migration file *file-name*

where ***file-name*** (1-1024 characters) is a full virtual path to the file, including the path name of the managed volume that contains it. This is the file path that clients see.

Unlike the command to stop all migrations in a volume, this does not prompt for confirmation.

For example, this command cancels the migration for a single file:

```
bstnA(gbl)# end
bstnA# cancel migration file /claims/y2kclaims/artwork/bigpict.tif
bstnA# ...
```

Addressing Inconsistent Directory Attributes

Directory attributes, such as access privileges, can become inconsistent in a namespace if those attributes are changed on the filer directly, outside of the control of the ARX. Directory attributes can become inconsistent also in special cases in which the actual disk on a filer is full, or when a quota for disk capacity usage by a specific user is met, and there is not enough space available on the disk to update the directory attribute.

In most cases, the ARX will recognize and address any inconsistencies that it encounters transparently, without any need for intervention by the user. However, in cases in which attributes have been made to be inconsistent by changes outside of the ARX, it may be expedient to address those inconsistencies explicitly. Mostly, this section describes the procedures that are available for use in the case of attributes having been changed outside of the ARX's management.

It is not necessary for directory attributes to be consistent across all striped directories. When a backout is unsuccessful or is not possible, the system continues to operate with inconsistent attributes and continues to provide access to the directory. Full access to a directory is restored automatically when an out-of-space condition is removed, without any administrative action executed on the ARX.

Metadata Inconsistencies

In addition to quota and disk space issues, metadata inconsistencies can occur in multi-protocol volumes in which both CIFS and NFS are used. In this case, inconsistencies can occur without being caused by the client; for example, a file name or directory name simply may be different between CIFS and NFS.

The nsck report metadata-only command displays the metadata for a specified namespace or volume. The resulting reports include information about directories with inconsistent attributes, if any are found. A directory is flagged as having inconsistent attributes when a discrepancy is identified in the attributes between a master directory and one or more of its stripes.

Attribute Inconsistency Use Cases

Attribute inconsistency encompasses two distinct use cases:

- Attributes becoming inconsistent as a result of storage depletion.
- Attributes becoming inconsistent because file attributes were changed explicitly outside of the ARX's management.

In most cases, the ARX will recognize and address any inconsistencies that it encounters transparently, without any need for intervention by the user. This section mostly describes the procedures that are available for use in the case of attributes having been changed outside of the ARX's management.

Finding and Showing Inconsistent Attributes

If inconsistent directory attributes are present in a particular namespace, they will be visible in the contents of namespace check (nsck) reports that are executed against that namespace.

Finding Attribute Inconsistencies

The nsck... report inconsistencies command compares the attributes of the master directory to the attributes of each secondary directory and reports any discrepancies.

There are three types of inconsistency that can be reported for directories; one is for secondary directory attributes, and the other two are for directory metadata:

- IS – The IS inconsistency type is reported for any secondary directory for which the attributes do not match the attributes of the master directory. The inconsistency is reported only for filer shares that have been detected as storing all security descriptor information.
- MI – The MI inconsistency type is reported on any master directory that is marked with the inconsistent attribute flag, but where no shares are detected as inconsistent.
- SI – The SI inconsistency type is reported on any master directory that is not marked with the inconsistent attribute flag, but where the shares are detected as inconsistent. This inconsistency will be reported only for secondary directories that have been detected as storing all security descriptor information.

A brief description of each of these types is included in the report's legend.

The nsck... report inconsistencies command also compares the state of striped directory attributes to the inconsistent-attributes flag in the metadata. Any discrepancies that are found are reported.

Synchronizing Directory Attributes

The synchronization process checks stripe directory attributes, reports inconsistencies found, synchronizes directory attributes, and corrects metadata as necessary.

Use the nsck... sync directories command to synchronize inconsistency directory attributes.

The synchronization process identifies a number of different cases of attribute inconsistency and reports them if they're present:

- IS – striped directory attributes inconsistent.
- SS – striped directory attributes synchronized.
- IA – inconsistent attributes.
- CA – consistent attributes.
- SY – directory metadata synchronized.

Resolving Directory Attribute Inconsistency

Each time that a directory is accessed, the ARX software attempts to re-synchronize the directory attributes. This will fail as long as the filer reports that it is out of disk space. The inconsistent attributes can be cleared using one of these methods:

- reducing the disk usage
- increasing the disk space

Disk usage can be reduced either by removing files that reside on the full filer share, or by migrating them to another filer share in the volume. The SNMP trap that is sent when directory attributes are found to be inconsistent contains the information necessary for identifying which file system needs attention.

Increasing disk space can be performed only by the filer administrator, but modifying quotas or resizing file systems can be performed by other authorized users.

File removal is performed by the end user, through the management interface.

File migration can be performed by an administrator by modifying policy rules to place less data on full filer shares, or a user can change the time stamps on a file to cause a migration rule to move it to a different tier. Linux and UNIX users can use the standard shell command “touch” to accomplish this. Windows Explorer and Windows command line interfaces do not offer direct ways of manipulating file time stamps, but there is a “touch” command available in Windows Services for UNIX and Subsystem for UNIX Applications. Alternatively, the following command, executed at a command prompt, emulates the “touch” command:

copy /b filename +”

Per Operation Restriction Summary

Operations that can affect the consistency of a directory structure across filer shares cannot be allowed, and are restricted. Operations that could potentially grant access to file data in violation of the desired directory attributes also are disallowed.

Note the following operation restrictions:

- For files:
    - Open-file is always allowed, since access is controlled primarily by the object permissions, not by the parent directory permissions.
    - Get-attributes is always allowed. Directory attributes are always returned from the directory master, and are always correct. File attributes are returned from the file, which is always correct.
    - Set-attributes on a file is unrestricted.

- If a set-attributes operation is attempted on a directory that is marked as inconsistent-attributes, the operation is refused with a disk-full error.
- Create-file is allowed if the target share is on the same filer share as the master directory of the parent directory, since the inherited attributes and permission checking will be correct. bstnA(gbl)#All other cases of create-file will be refused with a disk-full error.
- Remove-file is allowed if the file is on the same filer share as the master of the parent directory. Otherwise, it will fail with an access-denied error.
- The set-file-info request to set the disposition flag to delete-on-close is treated identically to the remove-file operation.
- If either the source or destination directory of a rename-file or remake-directory operation is marked as inconsistent-attributes, the operation is refused with a disk-full error.

- For directories:
  - Readdir (find-files in CIFS) is always allowed.
  - Create-directory is allowed if the target filer share is the same filer share as the directory master of the parent directory. All other cases of create-directory will be refused with a disk-full error.
  - Remove-directory is allowed if the directory master is located on the same filer share as the parent directory master.

11

Troubleshooting CIFS Services

Overview

This chapter describes how to run various show commands that monitor client traffic to a front-end CIFS service. It also contains commands to break a CIFS-client connection or close an open file. This extends the previous chapters, which described tools for collecting diagnostic information and troubleshooting network connections.

Showing Client-Connection Statistics

A CIFS service keeps client-access statistics for each of its exports/shares. Use the show cifs-service exports command to show these statistics:

show cifs-service exports {*fqdn* | all}

where ***fqdn*** **|** **all** is a required choice:

fqdn (1-128 characters) identifies a single CIFS service by its fully-qualified domain name (for example, www.myorg.org).

all selects all CIFS services on the current switch.

Each NSM processor exports all of the exports/shares from a CIFS service. The output from this command therefore contains a set of rows for each processor, one per export/share. Each row describes the processor and export, and shows the statistics for client connections to that export on that processor.

For example, this shows all client connections to the CIFS service at "ac1.medarch.org:"

```
bstnA> show cifs-service exports ac1.medarch.org


Global Server: ac1.MEDARCH.ORG            [192.168.25.15]

                              Namespace                                 Tree Connects
Proc  Export                  Virtual Path                        Curr   Peak   Total
----  ----------------------  ----------------------------------  ----   ----   -----
2.5   acopia#ns3_lab_equipment$   medarcv                         0      0      0
                              /lab_equipment
2.5   acopia#ns3_rcrds$       medarcv                             0      0      0
                              /rcrds
2.5   acopia#ns4_claims$      insur                               0      0      0
                              /claims
2.5   ARCHIVES                medarcv                             0      0      0
                              /rcrds
2.5   bulkstorage             medarcv                             0      0      0
                              /rcrds
2.5   CELEBS                  medarcv                             0      0      0
                              /rcrds/VIP_wing
2.5   chem_results            medarcv                             0      0      0
                              /test_results
2.5   CLAIMS                  insur                               0      0      0
                              /claims
2.5   E$                      medarcv                             0      0      0
                              /acopia$ns3
```

```
2.5   F$                       insur                              0      0      0
                               /acopia$ns4
2.5   labs                     medarcv                            0      0      0
                               /lab_equipment
2.5   MP3S                     medarcv                            0      0      0
                               /rcrds/2011/mp3downloads
2.5   SPECS                    insur                              0      0      0
                               /claims/specs
2.5   STATS                    insur                              0      0      0
                               /claims/stats
2.5   xraysScanners            medarcv                            0      0      0
                               /lab_equipment
2.5   Y2004                    medarcv                            0      0      0
                               /rcrds/2004
2.5   Y2005                    medarcv                            0      0      0
                               /rcrds/2005
2.5   Y2010                    medarcv                            0      0      0
                               /rcrds/2010
2.5   Z$                       medarcv                            0      0      0
                               /test_results
2.6   acopia#ns3_lab_equipment$   medarcv                         0      0      0
                               /lab_equipment
2.6   acopia#ns3_rcrds$        medarcv                            0      0      0
                               /rcrds
2.6   acopia#ns4_claims$       insur                              0      0      0
                               /claims
2.6   ARCHIVES                 medarcv                            0      0      0
                               /rcrds
2.6   bulkstorage              medarcv                            0      0      0
                               /rcrds
2.6   CELEBS                   medarcv                            0      0      0
                               /rcrds/VIP_wing
2.6   chem_results             medarcv                            0      0      0
                               /test_results
2.6   CLAIMS                   insur                              0      0      0
                               /claims
2.6   E$                       medarcv                            0      0      0
                               /acopia$ns3
2.6   F$                       insur                              0      0      0
                               /acopia$ns4
2.6   labs                     medarcv                            0      0      0
                               /lab_equipment
2.6   MP3S                     medarcv                            0      0      0
                               /rcrds/2011/mp3downloads
2.6   SPECS                    insur                              0      0      0
                               /claims/specs
2.6   STATS                    insur                              0      0      0
                               /claims/stats
2.6   xraysScanners            medarcv                            0      0      0
                               /lab_equipment
2.6   Y2004                    medarcv                            0      0      0
                               /rcrds/2004
2.6   Y2005                    medarcv                            0      0      0
                               /rcrds/2005
2.6   Y2010                    medarcv                            0      0      0
                               /rcrds/2010
2.6   Z$                       medarcv                            0      0      0
                               /test_results
2.7   acopia#ns3_lab_equipment$   medarcv                         0      0      0
                               /lab_equipment
2.7   acopia#ns3_rcrds$        medarcv                            0      0      0
                               /rcrds
```

```
2.7   acopia#ns4_claims$       insur                              0      0      0
                               /claims
2.7   ARCHIVES                 medarcv                            0      0      0
                               /rcrds
2.7   bulkstorage              medarcv                            0      0      0
                               /rcrds
2.7   CELEBS                   medarcv                            0      1      4
                               /rcrds/VIP_wing
2.7   chem_results             medarcv                            0      0      0
                               /test_results
2.7   CLAIMS                   insur                              0      0      0
                               /claims
2.7   E$                       medarcv                            0      0      0
                               /acopia$ns3
2.7   F$                       insur                              0      0      0
                               /acopia$ns4
2.7   labs                     medarcv                            0      1      1
                               /lab_equipment
2.7   MP3S                     medarcv                            0      0      0
                               /rcrds/2011/mp3downloads
2.7   SPECS                    insur                              0      0      0
                               /claims/specs
2.7   STATS                    insur                              0      0      0
                               /claims/stats
2.7   xraysScanners            medarcv                            0      0      0
                               /lab_equipment
2.7   Y2004                    medarcv                            0      0      0
                               /rcrds/2004
2.7   Y2005                    medarcv                            0      0      0
                               /rcrds/2005
2.7   Y2010                    medarcv                            0      0      0
                               /rcrds/2010
2.7   Z$                       medarcv                            0      0      0
                               /test_results
2.8   acopia#ns3_lab_equipment$   medarcv                         0      0      0
                               /lab_equipment
2.8   acopia#ns3_rcrds$        medarcv                            0      0      0
                               /rcrds
2.8   acopia#ns4_claims$       insur                              0      0      0
                               /claims
2.8   ARCHIVES                 medarcv                            0      0      0
                               /rcrds
2.8   bulkstorage              medarcv                            0      0      0
                               /rcrds
2.8   CELEBS                   medarcv                            0      0      0
                               /rcrds/VIP_wing
2.8   chem_results             medarcv                            0      0      0
                               /test_results
2.8   CLAIMS                   insur                              0      0      0
                               /claims
2.8   E$                       medarcv                            0      0      0
                               /acopia$ns3
2.8   F$                       insur                              0      0      0
                               /acopia$ns4
2.8   labs                     medarcv                            0      0      0
                               /lab_equipment
2.8   MP3S                     medarcv                            0      0      0
                               /rcrds/2011/mp3downloads
2.8   SPECS                    insur                              0      0      0
                               /claims/specs
2.8   STATS                    insur                              0      0      0
                               /claims/stats
```

```
2.8   xraysScanners             medarcv                          0      0      0
                                /lab_equipment
2.8   Y2004                    medarcv                          1      1      3
                                /rcrds/2004
2.8   Y2005                    medarcv                          0      0      0
                                /rcrds/2005
2.8   Y2010                    medarcv                          0      0      0
                                /rcrds/2010
2.8   Z$                       medarcv                          0      0      0
                                /test_results
2.9   acopia#ns3_lab_equipment$   medarcv                       0      0      0
                                /lab_equipment
2.9   acopia#ns3_rcrds$        medarcv                          0      0      0
                                /rcrds
2.9   acopia#ns4_claims$       insur                            0      0      0
                                /claims
2.9   ARCHIVES                 medarcv                          0      1      6
                                /rcrds
2.9   bulkstorage              medarcv                          0      0      0
                                /rcrds
2.9   CELEBS                   medarcv                          0      0      0
                                /rcrds/VIP_wing
2.9   chem_results             medarcv                          0      0      0
                                /test_results
2.9   CLAIMS                   insur                            0      0      0
                                /claims
2.9   E$                       medarcv                          0      0      0
                                /acopia$ns3
2.9   F$                       insur                            0      0      0
                                /acopia$ns4
2.9   labs                     medarcv                          0      0      0
                                /lab_equipment
2.9   MP3S                     medarcv                          1      1      3
                                /rcrds/2011/mp3downloads
2.9   SPECS                    insur                            0      0      0
                                /claims/specs
2.9   STATS                    insur                            0      0      0
                                /claims/stats
2.9   xraysScanners            medarcv                          0      0      0
                                /lab_equipment
2.9   Y2004                    medarcv                          0      0      0
                                /rcrds/2004
2.9   Y2005                    medarcv                          1      1      2
                                /rcrds/2005
2.9   Y2010                    medarcv                          0      0      0
                                /rcrds/2010
2.9   Z$                       medarcv                          0      0      0
                                /test_results
2.10  acopia#ns3_lab_equipment$   medarcv                       0      0      0
                                /lab_equipment
2.10  acopia#ns3_rcrds$        medarcv                          0      0      0
                                /rcrds
2.10  acopia#ns4_claims$       insur                            0      0      0
                                /claims
2.10  ARCHIVES                 medarcv                          1      1      2
                                /rcrds
2.10  bulkstorage              medarcv                          0      0      0
                                /rcrds
2.10  CELEBS                   medarcv                          0      0      0
                                /rcrds/VIP_wing
2.10  chem_results             medarcv                          0      0      0
                                /test_results
```

```
2.10  CLAIMS                  insur                              0      0      0
                              /claims
2.10  E$                      medarcv                            0      0      0
                              /acopia$ns3
2.10  F$                      insur                              0      0      0
                              /acopia$ns4
2.10  labs                    medarcv                            0      0      0
                              /lab_equipment
2.10  MP3S                    medarcv                            0      0      0
                              /rcrds/2011/mp3downloads
2.10  SPECS                   insur                              0      0      0
                              /claims/specs
2.10  STATS                   insur                              0      0      0
                              /claims/stats
2.10  xraysScanners           medarcv                            0      0      0
                              /lab_equipment
2.10  Y2004                   medarcv                            0      0      0
                              /rcrds/2004
2.10  Y2005                   medarcv                            0      0      0
                              /rcrds/2005
2.10  Y2010                   medarcv                            0      0      0
                              /rcrds/2010
2.10  Z$                      medarcv                            0      0      0
                              /test_results
2.11  acopia#ns3_lab_equipment$   medarcv                       0      0      0
                              /lab_equipment
2.11  acopia#ns3_rcrds$       medarcv                            0      0      0
                              /rcrds
2.11  acopia#ns4_claims$      insur                              0      0      0
                              /claims
2.11  ARCHIVES                medarcv                            0      0      0
                              /rcrds
2.11  bulkstorage             medarcv                            0      0      0
                              /rcrds
2.11  CELEBS                  medarcv                            0      0      0
                              /rcrds/VIP_wing
2.11  chem_results            medarcv                            0      0      0
                              /test_results
2.11  CLAIMS                  insur                              0      0      0
                              /claims
2.11  E$                      medarcv                            0      0      0
                              /acopia$ns3
2.11  F$                      insur                              0      0      0
                              /acopia$ns4
2.11  labs                    medarcv                            0      0      0
                              /lab_equipment
2.11  MP3S                    medarcv                            0      0      0
                              /rcrds/2011/mp3downloads
2.11  SPECS                   insur                              0      0      0
                              /claims/specs
2.11  STATS                   insur                              0      0      0
                              /claims/stats
2.11  xraysScanners           medarcv                            0      0      0
                              /lab_equipment
2.11  Y2004                   medarcv                            0      0      0
                              /rcrds/2004
2.11  Y2005                   medarcv                            0      0      0
                              /rcrds/2005
2.11  Y2010                   medarcv                            0      0      0
                              /rcrds/2010
2.11  Z$                      medarcv                            0      0      0
                              /test_results
```

```
2.12  acopia#ns3_lab_equipment$   medarcv                          0      0      0
                                /lab_equipment
2.12  acopia#ns3_rcrds$         medarcv                          0      0      0
                                /rcrds
2.12  acopia#ns4_claims$        insur                            0      0      0
                                /claims
2.12  ARCHIVES                  medarcv                          0      0      0
                                /rcrds
2.12  bulkstorage               medarcv                          0      0      0
                                /rcrds
2.12  CELEBS                    medarcv                          0      0      0
                                /rcrds/VIP_wing
2.12  chem_results              medarcv                          0      0      0
                                /test_results
2.12  CLAIMS                    insur                            0      1      1
                                /claims
2.12  E$                        medarcv                          0      0      0
                                /acopia$ns3
2.12  F$                        insur                            0      0      0
                                /acopia$ns4
2.12  labs                      medarcv                          0      0      0
                                /lab_equipment
2.12  MP3S                      medarcv                          0      0      0
                                /rcrds/2011/mp3downloads
2.12  SPECS                     insur                            0      0      0
                                /claims/specs
2.12  STATS                     insur                            0      0      0
                                /claims/stats
2.12  xraysScanners             medarcv                          0      0      0
                                /lab_equipment
2.12  Y2004                     medarcv                          0      0      0
                                /rcrds/2004
2.12  Y2005                     medarcv                          0      0      0
                                /rcrds/2005
2.12  Y2010                     medarcv                          0      0      0
                                /rcrds/2010
2.12  Z$                        medarcv                          0      0      0
                                /test_results
                                                                 ----   ----   -----
                                                    Totals:       4      8     22

bstnA> ...
```

Showing Statistics from One NSM Processor

Each CIFS-client connection is managed by one NSM processor. If you show the connection statistics for a single CIFS service (as shown above), you can narrow the focus further to a single NSM processor. To accomplish this, identify the processor after the FQDN:

show cifs-service exports *fqdn slot.processor*

where

fqdn (1-128 characters) identifies the CIFS service,

slot (2 in the ARX-4000, 1 in all other platforms) is the slot number of the desired NSM, and

processor (1-12) is the NSM-processor number. Use show processors for a complete list of processors (and their modules and slots) on the ARX.

The output is formatted the same way as shown above. For example, this shows the connection statistics for the "ac1.medarch.org" service at processor 2.5:

```
bstnA> show cifs-service exports ac1.medarch.org 2.5


Global Server: ac1.MEDARCH.ORG            [192.168.25.15]

                              Namespace                                 Tree Connects
Proc  Export                  Virtual Path                          Curr   Peak   Total
----  ----------------------  ------------------------------------  ----   ----   -----
2.5   acopia#ns3_lab_equipment$   medarcv                          0      0      0
                              /lab_equipment
2.5   acopia#ns3_rcrds$       medarcv                             0      0      0
                              /rcrds
2.5   acopia#ns4_claims$      insur                               0      0      0
                              /claims
2.5   ARCHIVES                medarcv                             0      0      0
                              /rcrds
2.5   bulkstorage             medarcv                             0      0      0
                              /rcrds
2.5   CELEBS                  medarcv                             0      0      0
                              /rcrds/VIP_wing
2.5   chem_results            medarcv                             0      0      0
                              /test_results
2.5   CLAIMS                  insur                               0      0      0
                              /claims
2.5   E$                      medarcv                             0      0      0
                              /acopia$ns3
2.5   F$                      insur                               0      0      0
                              /acopia$ns4
2.5   labs                    medarcv                             0      0      0
                              /lab_equipment
2.5   MP3S                    medarcv                             0      0      0
                              /rcrds/2011/mp3downloads
2.5   SPECS                   insur                               0      0      0
                              /claims/specs
2.5   STATS                   insur                               0      0      0
                              /claims/stats
2.5   xraysScanners           medarcv                             0      0      0
                              /lab_equipment
2.5   Y2004                   medarcv                             0      0      0
                              /rcrds/2004
2.5   Y2005                   medarcv                             0      0      0
                              /rcrds/2005
2.5   Y2010                   medarcv                             0      0      0
                              /rcrds/2010
2.5   Z$                      medarcv                             0      0      0
                              /test_results
                                                                  ----   ----   -----
                                                        Totals:      0      0      0
```

Showing Client Sessions

If MMC is allowed for this CIFS service (see *Supporting MMC Browsing*, on page 11-21 of the *ARX® CLI Storage-Management Guide*), authorized clients can use MMC to list all client connections to the CIFS service. You can perform the same operation from the CLI. Use the show cifs-service user-sessions command to list the client connections to one or all CIFS services:

show cifs-service user-sessions {*fqdn* | all}

where ***fqdn*** | **all** is a required choice:

fqdn (1-128 characters) identifies a single CIFS service by its fully-qualified domain name (for example, www.company.com).

all selects all CIFS services on the current switch.

The output is a table with one row per client session. Each row shows the NSM processor to which the client is connected, the client's IP address, the CIFS protocol used (SMB or SMB2), the authentication method that the client used, whether or not SMB signing was used, the age of the connection (in seconds), and the user name that the client entered.

For example, this finds the client sessions for all CIFS services on the switch:

```
bstnA# show cifs-service user-sessions all


CIFS Service ac1.MEDARCH.ORG

Proc  IP Address       Proto  Auth      Sign  Age            Username
----  ---------------  -----  --------  ----  -------------  ----------------------------
2.7   172.16.100.20    SMB    Kerberos  Yes   00:00:47       juser@MEDARCH.ORG
2.7   172.16.100.20    SMB    NTLM      Yes   00:00:49       juser@MEDARCH.ORG
2.8   172.16.108.112   SMB    Kerberos  Yes   00:00:21       Administrator@MEDARCH.ORG
2.9   172.16.100.68    SMB    Kerberos  Yes   00:00:47       lfine_md@MEDARCH.ORG
2.10  172.16.100.209   SMB    Kerberos  Yes   00:00:47       choward_md@MEDARCH.ORG


Total number of users displayed is 5

bstnA#
```

Showing Client Sessions at One NSM Processor

Each CIFS-client session is managed by one NSM processor. If you show the connections to a single CIFS service (as shown above), you can narrow the focus further to a single NSM processor. To accomplish this, identify the processor after the FQDN:

show cifs-service user-sessions *fqdn slot.processor*

where

fqdn (1-128 characters) identifies a single CIFS service,

slot (2 on the ARX-4000, 1 in all other platforms) is the slot number of the desired NSM, and

processor (1-12) is the NSM-processor number. Use show processors for a complete list of processors (and their modules and slots) on the ARX.

The output is formatted the same way as shown above.

For example, this shows the session hosted by the "ac1.medarch.org" service at NSM processor 2.7:

```
bstnA> show cifs-service user-sessions ac1.medarch.org 2.7

CIFS Service ac1.MEDARCH.ORG

Proc  IP Address       Auth      Sign  Age            Username
----  ---------------  --------  ----  -------------  ----------------------------
2.7   172.16.100.20    Kerberos  Yes   00:00:27       Administrator@MEDARCH.ORG
2.7   172.16.100.20    NTLM      Yes   00:00:35       juser@MEDARCH.ORG

Total number of users displayed is 2


bstnA> ...
```

Showing CIFS Work Queue Statistics

The show statistics cifs work-queues command enables you to display time statistics for various CIFS-related tasks that transit the CIFS work queues. These statistics are useful for diagnosing problems with CIFS performance.

The command syntax is:

```
show statistics cifs work-queues volume-group vg-id
```

or

```
show statistics cifs work-queues instance instance-id
```

where *vg-id* or *instance-id* specifies the volume group or namespace instance identifier for which you want to display CIFS work queue statistics.

The command clear statistics cifs work-queues enables you to reset those statistics. The clear statistics filer command resets some CIFS statistics that are not covered by clear statistics cifs work-queues.

Showing CIFS Fastpath Statistics

The command show statistics cifs fastpath enables you to display a variety of statistics related to CIFS servers. These statistics include the numbers of front end and back end connections, the number of transactions handled, and others.

The command syntax is:

show statistics cifs fastpath [all | *slot.processor*]

where:

slot.processor indicates a specific NSM processor against which to execute the command.

For example:

```
bstnA# show statistics cifs fastpath


Proc Transactions FrontEnd    BackEnd     File File
     Handled      Connections Connections Info Handles
---- ------------ ----------- ----------- ---- -------
2.1  0            0           0           0    0
2.2  0            0           0           0    0
2.3  182          0           0           0    0
2.4  1840         6           0           0    0
2.5  128          1           0           0    0
2.6  0            0           0           0    0
2.7  71           2           2           0    0
2.8  46           1           2           0    0
2.9  97           2           4           0    0
2.10  44          1           2           0    0
2.11  0           0           0           0    0
2.12  27          0           0           0    0

bstnA#
```

Dropping a CIFS Client

From priv-exec mode, you can use drop cifs-service user-session to close a session with a CIFS client:

drop cifs-service user-session *fqdn slot.processor* ipaddress *client*

where

fqdn (1-128 characters) identifies a single CIFS service,

slot.processor identifies a network processor, and

client identifies the client session by its source-IP address.

A CLI prompt announces the results of the command.

For example, this command sequence shows all CIFS clients, drops a client session, then shows that the session's Age has restarted (indicating that the client application re-connected to the CIFS service):

```
bstnA> show cifs-service user-sessions all


CIFS Service ac1.MEDARCH.ORG

Proc  IP Address       Proto  Auth      Sign  Age           Username
----  ---------------  -----  --------  ----  ------------  ----------------------------
2.7   172.16.100.20    SMB    NTLM      Yes   00:00:31      juser@MEDARCH.ORG
2.7   172.16.100.20    SMB    Kerberos  Yes   00:00:32      juser@MEDARCH.ORG
2.8   172.16.108.112   SMB    Kerberos  Yes   00:00:17      Administrator@MEDARCH.ORG
2.9   172.16.100.68    SMB    Kerberos  Yes   00:12:38      lfine_md@MEDARCH.ORG
2.10  172.16.100.209   SMB    Kerberos  Yes   00:00:31      choward_md@MEDARCH.ORG


Total number of users displayed is 5

bstnA> enable
bstnA# drop cifs-service user-session ac1.medarch.org 2.9 ipaddress 172.16.100.68

% INFO: All CIFS user sessions for the specified IP address were dropped.

bstnA# show cifs-service user-sessions all


CIFS Service ac1.MEDARCH.ORG

Proc  IP Address       Proto  Auth      Sign  Age           Username
----  ---------------  -----  --------  ----  ------------  ----------------------------
2.7   172.16.100.20    SMB    NTLM      Yes   00:00:31      juser@MEDARCH.ORG
2.7   172.16.100.20    SMB    Kerberos  Yes   00:00:32      juser@MEDARCH.ORG
2.8   172.16.108.112   SMB    Kerberos  Yes   00:00:17      Administrator@MEDARCH.ORG
2.9   172.16.100.68    SMB    Kerberos  Yes   00:00:04      lfine_md@MEDARCH.ORG
2.10  172.16.100.209   SMB    Kerberos  Yes   00:00:31      choward_md@MEDARCH.ORG


Total number of users displayed is 5

bstnA# ...
```

Listing Open Files in a CIFS Service

To list the files currently opened by CIFS clients, use the show cifs-service open-files command:

show cifs-service open-files {*fqdn* | all}

where ***fqdn*** **|** **all** is a required choice:

fqdn (1-128 characters) is the fully-qualified domain name (for example, www.company.com) for the CIFS service's global server.

all selects all CIFS services on the current switch.

If remote-Windows management is allowed for this CIFS service (see *Supporting MMC Browsing*, on page 11-21 of the *ARX® CLI Storage-Management Guide*), authorized clients can use MMC to list these files.

For each open file, this shows the NSM processor that is serving the file to the client, the client's IP address and identity, the "Virtual" IP address and share that the client application sees, the "Virtual" location of the file (from the client's perspective), the filer's IP address and share name, and the location of the file from the filer's perspective.

For example, this finds some files that are opened in the "ac1.medarch.org" CIFS service:

```
bstnA> show cifs-service open-files ac1.MEDARCH.ORG
Cifs Open Files
---------------

Proc:  2.7
    User IP:                172.16.100.20
    User Name:              juser@MEDARCH.ORG
    Namespace:              medarcv
    Virtual IP:             192.168.25.15
    Virtual Path:           \rcrds\VIP_wing\fractures\examSchedule.doc
    Virtual Share:          CELEBS
    Virtual FID:            589
    Filer IP:               192.168.25.29
    Filer Share:            CELEBS$
    Path on Filer:          \fractures\examSchedule.doc
    Open Mode:              Read+Write
    Locks:                  0

...

Proc:  2.7
    User IP:                172.16.100.20
    User Name:              juser@MEDARCH.ORG
    Namespace:              medarcv
    Virtual IP:             192.168.25.15
    Virtual Path:           \rcrds\VIP_wing\holdback\orSched.doc
    Virtual Share:          CELEBS
    Virtual FID:            604
    Filer IP:               192.168.25.20
    Filer Share:            CELEBS$
    Path on Filer:          \holdback\orSched.doc
    Open Mode:              Read+Write
    Locks:                  0
```

```
Proc:  2.9
    User IP:                172.16.100.68
    User Name:              lfine_md@MEDARCH.ORG
    Namespace:              medarcv
    Virtual IP:             192.168.25.15
    Virtual Path:           \rcrds\2010\holdback\examSchedule.doc
    Virtual Share:          Y2010
    Virtual FID:            536
    Filer IP:               192.168.25.29
    Filer Share:            Y2010
    Path on Filer:          \holdback\examSchedule.doc
    Open Mode:              Read+Write
    Locks:                  0

Proc:  2.9
    User IP:                172.16.100.68
    User Name:              lfine_md@MEDARCH.ORG
    Namespace:              medarcv
    Virtual IP:             192.168.25.15
    Virtual Path:           \rcrds\2010\holdback\stdProcedure.doc
    Virtual Share:          Y2010
    Virtual FID:            537
    Filer IP:               192.168.25.20
    Filer Share:            Y2010
    Path on Filer:          \holdback\stdProcedure.doc
    Open Mode:              Read+Write
    Locks:                  0

...

Proc:  2.10
    User IP:                172.16.100.209
    User Name:              choward_md@MEDARCH.ORG
    Namespace:              medarcv
    Virtual IP:             192.168.25.15
    Virtual Path:           \rcrds\holdback\memo.doc
    Virtual Share:          ARCHIVES
    Virtual FID:            535
    Filer IP:               192.168.25.20
    Filer Share:            histories
    Path on Filer:          \holdback\memo.doc
    Open Mode:              Read+Write
    Locks:                  0

Total number of files displayed is 22
bstnA> ...
```

Focusing on One NSM Processor

As mentioned above, each CIFS session is managed by a single NSM processor. If you show the open files at a single CIFS service, you can further narrow the focus to a single NSM processor. To accomplish this, identify the processor after the FQDN for the CIFS service:

show cifs-service open-files *fqdn slot.processor*

where

fqdn (1-128 characters) identifies a single CIFS service,

slot (2 on the ARX-4000, 1 in all other platforms) is the slot number of the desired NSM, and

processor (1-12) is the NSM-processor number. Use show processors for a complete list of processors (and their modules and slots) on the ARX.

For example, this shows all open files hosted by the "ac1.medarch.org" service at NSM processor 2.9:

```
bstnA> show cifs-service open-files ac1.medarch.org 2.9
Cifs Open Files
---------------

Proc:  2.9
    User IP:                172.16.100.68
    User Name:              lfine_md@MEDARCH.ORG
    Namespace:              medarcv
    Virtual IP:             192.168.25.15
    Virtual Path:           \rcrds\2010\holdback\examSchedule.doc
    Virtual Share:          Y2010
    Virtual FID:            536
    Filer IP:               192.168.25.29
    Filer Share:            Y2010
    Path on Filer:          \holdback\examSchedule.doc
    Open Mode:              Read+Write
    Locks:                  0

Proc:  2.9
    User IP:                172.16.100.68
    User Name:              lfine_md@MEDARCH.ORG
    Namespace:              medarcv
    Virtual IP:             192.168.25.15
    Virtual Path:           \rcrds\2010\holdback\stdProcedure.doc
    Virtual Share:          Y2010
    Virtual FID:            537
    Filer IP:               192.168.25.20
    Filer Share:            Y2010
    Path on Filer:          \holdback\stdProcedure.doc
    Open Mode:              Read+Write
    Locks:                  0

Total number of files displayed is 2
bstnA> ...
```

Closing an Open File

For situations where a client needs access to a long-opened file, you can use the close cifs file command to forcibly close the file. Invoke this command from priv-exec mode:

close cifs file *fqdn slot.processor* fid *file-id*

where

fqdn (1-128 characters) identifies a single CIFS service,

slot.processor identifies a network processor, and

file-id (0-65535) identifies the file. This ID is shown in the output of show cifs-service open-files, described above.

The CLI indicates a successful close by showing the processor and FID of the closed file.

For example, this command sequence shows all open files hosted by the "ac1.medarch.org" service at processor 2.10, closes one, and shows that the file is no-longer open:

```
bstnA> show cifs-service open-files ac1.medarch.org 2.10
Cifs Open Files
---------------

Proc:  2.10
    User IP:                172.16.100.209
    User Name:              choward_md@MEDARCH.ORG
    Namespace:              medarcv
    Virtual IP:             192.168.25.15
    Virtual Path:           \rcrds\holdback\examSchedule.doc
    Virtual Share:          ARCHIVES
    Virtual FID:            532
    Filer IP:               192.168.25.29
    Filer Share:            prescriptions
    Path on Filer:          \holdback\examSchedule.doc
    Open Mode:              Read+Write
    Locks:                  0

Proc:  2.10
    User IP:                172.16.100.209
    User Name:              choward_md@MEDARCH.ORG
    Namespace:              medarcv
    Virtual IP:             192.168.25.15
    Virtual Path:           \rcrds\holdback\stdProcedure.doc
    Virtual Share:          ARCHIVES
    Virtual FID:            533
    Filer IP:               192.168.25.20
    Filer Share:            histories
    Path on Filer:          \holdback\stdProcedure.doc
    Open Mode:              Read+Write
    Locks:                  0

Proc:  2.10
    User IP:                172.16.100.209
    User Name:              choward_md@MEDARCH.ORG
    Namespace:              medarcv
    Virtual IP:             192.168.25.15
    Virtual Path:           \rcrds\holdback\recoveryStats.xls
```

```
    Virtual Share:            ARCHIVES
    Virtual FID:              534
    Filer IP:                 192.168.25.29
    Filer Share:              prescriptions
    Path on Filer:            \holdback\recoveryStats.xls
    Open Mode:                Read+Write
    Locks:                    0

Proc:  2.10
    User IP:                  172.16.100.209
    User Name:                choward_md@MEDARCH.ORG
    Namespace:                medarcv
    Virtual IP:               192.168.25.15
    Virtual Path:             \rcrds\holdback\memo.doc
    Virtual Share:            ARCHIVES
    Virtual FID:              535
    Filer IP:                 192.168.25.20
    Filer Share:              histories
    Path on Filer:            \holdback\memo.doc
    Open Mode:                Read+Write
    Locks:                    0

Total number of files displayed is 4
bstnA> enable
bstnA# close cifs file ac1.medarch.org 2.10 fid 533
slot:2.2  CIFS open file with FID:533  closed.
bstnA# show cifs-service open-files ac1.medarch.org 2.10
Cifs Open Files
---------------

Proc:  2.10
    User IP:                  172.16.100.209
    User Name:                choward_md@MEDARCH.ORG
    Namespace:                medarcv
    Virtual IP:               192.168.25.15
    Virtual Path:             \rcrds\holdback\examSchedule.doc
    Virtual Share:            ARCHIVES
    Virtual FID:              532
    Filer IP:                 192.168.25.29
    Filer Share:              prescriptions
    Path on Filer:            \holdback\examSchedule.doc
    Open Mode:                Read+Write
    Locks:                    0

Proc:  2.10
    User IP:                  172.16.100.209
    User Name:                choward_md@MEDARCH.ORG
    Namespace:                medarcv
    Virtual IP:               192.168.25.15
    Virtual Path:             \rcrds\holdback\recoveryStats.xls
    Virtual Share:            ARCHIVES
    Virtual FID:              534
    Filer IP:                 192.168.25.29
    Filer Share:              prescriptions
...

Total number of files displayed is 3
bstnA# ...
```

Listing Kerberos Tickets Granted to Clients

If Kerberos is active for this CIFS service, the service passes Kerberos *tickets* to clients who successfully authenticate. The CIFS service caches the Kerberos tickets on behalf of its clients. These tickets have an expiration time. To view all cached tickets, the clients who hold the tickets, and the expiration times, use the show cifs-service kerberos-tickets command:

show cifs-service kerberos-tickets {*fqdn* | all}

where ***fqdn*** **|** **all** is a required choice:

fqdn (1-128 characters) is the fully-qualified domain name (for example, www.company.com) for the CIFS service's global server.

all selects all CIFS services on the current switch.

The output refers to a client as a *Principal* and a server or Ticket-Granting Ticket as a *Service Principal*. For each principal with a Kerberos ticket, this shows the contents of all tickets (including grant times and expiration times). The total number of principals, ticket-granting tickets, and service tickets appears at the end. If you select **all**, each CIFS service has a separate section with the totals at the end.

For example, this finds that the "ac1.medarch.org" service has granted tickets to two principals, "lfine_md@MEDARCH.ORG" and "juser@MEDARCH.ORG:"

```
bstnA> show cifs-service kerberos-tickets ac1.medarch.org


Service:        ac1.medarch.org

  Start Time(UTC)    Expiry Time(UTC)   Service Principal
  ------------------------------------------------------------------------------

Principal:  lfine_md@MEDARCH.ORG

  10/21/2009 06:08:18 PM 10/22/2009 04:08:18 AM krbtgt/MEDARCH.ORG@MEDARCH.ORG
        Renew Till:   10/28/2009 06:08:18 PM

Principal:  juser@MEDARCH.ORG

  10/21/2009 06:02:30 PM 10/22/2009 04:02:30 AM krbtgt/MEDARCH.ORG@MEDARCH.ORG
        Renew Till:   10/28/2009 06:02:30 PM

  Total number of principals displayed is 2
  Total number of ticket-granting-tickets cached for this selection is 2
  Total number of service tickets cached for this selection is 0
bstnA> ...
```

Listing Tickets Granted to a Particular Principal

You can search for a particular principal by adding the user clause to the end of the command:

show cifs-service kerberos-tickets {*fqdn* | all} user *username*

where ***username*** (1-128 characters) is a search string for one or more principals. The output includes all principles whose first characters match this string. This is a case-blind comparison, so “jpub” matches “jpublic,” “jpublisher,” and “JPUBS.”

For example, this shows all tickets that “ac1.medarch.org” granted to any principal whose name starts with “juser:”

```
bstnA> show cifs-service kerberos-tickets ac1.medarch.org user juser


Service:        ac1.medarch.org

  Start Time(UTC)    Expiry Time(UTC)   Service Principal
  ------------------------------------------------------------------------------

Principal:  juser@MEDARCH.ORG

  10/21/2009 06:02:30 PM 10/22/2009 04:02:30 AM krbtgt/MEDARCH.ORG@MEDARCH.ORG
        Renew Till:   10/28/2009 06:02:30 PM

  Total number of principals displayed is 1
  Total number of ticket-granting-tickets cached for this selection is 1
  Total number of service tickets cached for this selection is 0
bstnA> ...
```

Leaving and Rejoining an AD Domain

Some AD policies set an expiration period for machine-account passwords; if a CIFS service's password expires, its clients can no longer use Kerberos authentication. Before the password expires, you can use a CLI command to reset it (see *Changing the ARX's Machine-Account Keys (Kerberos)*, on page 11-38 of the *ARX® CLI Storage-Management Guide*). If a service's password expires before you reset it with that command, you must remove and rebuild the CIFS-service configuration to get an entirely new machine account password. The CIFS service cannot use its expired password to get a new one.

Saving the CIFS-Service Configuration

You begin to remove a CIFS service by recording its configuration. Use the show global-config cifs *fqdn* command for an ordered list of the CLI commands required to build the *fqdn* service (recall *Focusing On Named Configurations*, on page 5-16). For example, this shows the configuration for the ac1.medarch.org service:

```
bstnA> show global-config cifs ac1.medarch.org
;================================ cifs ==================================
cifs ac1.MEDARCH.ORG
  description "insurance-claim records"
  kerberos-creds ac1.MEDARCH.ORG MEDARCH.ORG eHaPWO/6TGWZdDtL15dstg== ac1$ HOST/ac1.MEDARCH.ORG 2
  signatures
  dynamic-dns ac1
  dynamic-dns fs1
  dynamic-dns fs2
  dynamic-dns fs5
  dynamic-dns insur
  browsing medarcv
  browsing insur
  export medarcv /rcrds as ARCHIVES description "2 year-old medical records"
  export medarcv /rcrds/2005 as Y2005 filer-subshare
  export medarcv /lab_equipment as labs description "lab equipment"
  export medarcv /rcrds as bulkstorage description "big share, now merged thru ARX"
  export medarcv /lab_equipment as xraysScanners description "scanners and xray machines"
  export medarcv /test_results as chem_results description "chem-lab records"
  export medarcv /rcrds/VIP_wing as CELEBS filer-subshare hidden
  export medarcv /rcrds/2011/mp3downloads as MP3S filer-subshare
  export medarcv /rcrds/2004 as Y2004 filer-subshare
  export medarcv /rcrds/2010 as Y2010 filer-subshare
  export insur /claims as CLAIMS description "insurance claims"
  export insur /claims/specs as SPECS
  export insur /claims/stats as STATS description "claim stats"
  enable
  exit
bstnA> ...
```

Save this to a text file.

Remove the kerberos-creds command from the file. The CLI uses this command only for global-config playback. That command preserves the now-expired machine-account password, and precludes the domain-join command that you must execute later.

If there are any dynamic-dns commands, as above, remove all of them. If there are more than one, you may want to save them to a separate file. Each of these commands registers a DNS name with a dynamic-DNS server; this is only possible after the CIFS service rejoins the domain.

Also remove the enable command. You enable the service manually, after you rejoin the CIFS service to the AD domain.

For example, this is the correct configuration (without the kerberos-creds command, the enable command, or any dynamic-dns commands) to store for the above CIFS service:

```
cifs ac1.medarch.org
  description "insurance-claim records"
  signatures
  browsing medarcv
  browsing insur
  export medarcv /rcrds as ARCHIVES description "2 year-old medical records"
  export medarcv /rcrds/2005 as Y2005 filer-subshare
  export medarcv /lab_equipment as labs description "lab equipment"
  export medarcv /rcrds as bulkstorage description "big share, now merged thru ARX"
  export medarcv /lab_equipment as xraysScanners description "scanners and xray machines"
  export medarcv /test_results as chem_results description "chem-lab records"
  export medarcv /rcrds/VIP_wing as CELEBS filer-subshare hidden
  export medarcv /rcrds/2011/mp3downloads as MP3S filer-subshare
  export medarcv /rcrds/2004 as Y2004 filer-subshare
  export medarcv /rcrds/2010 as Y2010 filer-subshare
  export insur /claims as CLAIMS description "insurance claims"
  export insur /claims/specs as SPECS
  export insur /claims/stats as STATS description "claim stats"
  exit
```

Dropping Terminal-Confirmation Prompts

At the ARX CLI, enter no terminal confirmation to avoid any confirmation prompts. This makes it easier to copy and paste the above configuration into the CLI later. You can enter this command from any mode. For example:

```
bstnA(gbl)# no terminal confirmation
bstnA(gbl)# ...
```

Clearing Dynamic-DNS Hostnames (If Necessary)

If the CIFS service uses dynamic DNS, you must remove the service's hostname(s) from your local DNS before you remove the service. This is evident in the global-config file you created above; each dynamic-dns command in the file (if there are any) represents one DNS hostname. You can skip this section if your CIFS service does not use any dynamic-DNS hostnames.

Use no dynamic dns for each of this service's DNS hostnames. This sends hostname-removal notifications to local dynamic-DNS servers (typically DCs; see *Removing a Host Name*, on page 11-36 of the *ARX® CLI Storage-Management Guide*). For each dynamic-dns command in the

global-config file, go to gbl-cifs mode and enter the no form of that exact command. To continue the above example, this removes DNS support for five hostnames:

```
bstnA(gbl)# cifs ac1.medarch.org
bstnA(gbl-cifs[ac1.medarch.org])# no dynamic-dns ac1
bstnA(gbl-cifs[ac1.medarch.org])# no dynamic-dns fs1
bstnA(gbl-cifs[ac1.medarch.org])# no dynamic-dns fs2
bstnA(gbl-cifs[ac1.medarch.org])# no dynamic-dns fs5
bstnA(gbl-cifs[ac1.medarch.org])# no dynamic-dns insur
bstnA(gbl-cifs[ac1.medarch.org])# ...
```

Wait several seconds after you remove the final hostname, so that the DNS server(s) have time to remove the hostname(s) from service. You can use the show dynamic-dns command to monitor the operation (see *Showing Dynamic-DNS Status*, on page 11-36 of the *ARX® CLI Storage-Management Guide*).

Removing the CIFS Service

Once all dynamic-DNS hostnames are removed, you can remove the entire CIFS service. Use the no cifs *fqdn* command from gbl mode (as described in *Removing a CIFS Service*, on page 11-47 of the *ARX® CLI Storage-Management Guide*). For example:

```
bstnA(gbl-cifs[ac1.medarch.org])# exit
bstnA(gbl)# no cifs ac1.medarch.org
bstnA(gbl)# ...
```

Restoring the CIFS-Service Configuration

Add the service back by copying the configuration from the file above and pasting it into the CLI. Remember to omit the kerberos-creds command. For example:

```
bstnA(gbl)# cifs ac1.medarch.org
bstnA(gbl-cifs[ac1.medarch.org])#  description "insurance-claim records"
bstnA(gbl-cifs[ac1.medarch.org])#  signatures
bstnA(gbl-cifs[ac1.medarch.org])#  browsing medarcv
bstnA(gbl-cifs[ac1.medarch.org])#  browsing insur
bstnA(gbl-cifs[ac1.medarch.org])#  export medarcv /rcrds as ARCHIVES description "2 year-old
medical records"
bstnA(gbl-cifs[ac1.medarch.org])#  export medarcv /rcrds/2005 as Y2005 filer-subshare
bstnA(gbl-cifs[ac1.medarch.org])#  export medarcv /lab_equipment as labs description "lab
equipment"
bstnA(gbl-cifs[ac1.medarch.org])#  export medarcv /rcrds as bulkstorage description "big share,
now merged thru ARX"
bstnA(gbl-cifs[ac1.medarch.org])#  export medarcv /lab_equipment as xraysScanners
bstnA(gbl-cifs[ac1.medarch.org])#  export medarcv /test_results as chem_results
bstnA(gbl-cifs[ac1.medarch.org])#  export medarcv /rcrds/VIP_wing as CELEBS filer-subshare hidden
bstnA(gbl-cifs[ac1.medarch.org])#  export medarcv /rcrds/2011/mp3downloads as MP3S filer-subshare
bstnA(gbl-cifs[ac1.medarch.org])#  export medarcv /rcrds/2004 as Y2004 filer-subshare
bstnA(gbl-cifs[ac1.medarch.org])#  export medarcv /rcrds/2010 as Y2010 filer-subshare
bstnA(gbl-cifs[ac1.medarch.org])#  export insur /claims as CLAIMS description "insurance claims"
bstnA(gbl-cifs[ac1.medarch.org])#  export insur /claims/specs as SPECS
bstnA(gbl-cifs[ac1.medarch.org])#  export insur /claims/stats as STATS description "claim stats"
bstnA(gbl-cifs[ac1.medarch.org])#  exit
```

```
bstnA(gbl)# ...
```

Rejoining the AD Domain and Enabling the Service

The next step is to rejoin the CIFS service to the AD domain. This generates a new secret password for the CIFS service. Use the domain-join command from gbl-cifs mode, as described in *Joining a CIFS Service to an Active Directory Domain*, on page 11-26 of the *ARX® CLI Storage-Management Guide*. Then use the enable command from the same mode. For example, this joins the "ac1.medarch.org" service to its AD domain, "MEDARCH.ORG:"

```
bstnA(gbl)# cifs ac1.medarch.org
bstnA(gbl-cifs[ac1.medarch.org])# domain-join MEDARCH.ORG
Username: acoadmin
Password: ******

% INFO: Service 'ac1' successfully joined the domain using a
pre-created computer account.


% INFO: Service 'ac1' joined the domain with delegation type set to
Unconstrained. The delegation of services on 'ac1' is allowed only
with the Kerberos protocol.

bstnA(gbl-cifs[ac1.medarch.org])# enable
bstnA(gbl-cifs[ac1.medarch.org])# ...
```

After these commands, clients can use Kerberos to access the CIFS service.

Restoring Dynamic-DNS Hostnames (If Necessary)

If you removed any dynamic-DNS hostnames earlier, replay the dynamic-dns commands now. For example, this restores four hostnames to the "ac1.medarch.org" service:

```
bstnA(gbl)# cifs ac1.medarch.org
bstnA(gbl-cifs[ac1.medarch.org])# dynamic-dns ac1
bstnA(gbl-cifs[ac1.medarch.org])# dynamic-dns fs1
bstnA(gbl-cifs[ac1.medarch.org])# dynamic-dns fs2
bstnA(gbl-cifs[ac1.medarch.org])# dynamic-dns fs5
bstnA(gbl-cifs[ac1.medarch.org])# dynamic-dns insur
bstnA(gbl-cifs[ac1.medarch.org])# ...
```

Turning Confirmation Prompts Back On

Best practices dictate that terminal confirmation is enabled for everyday use of the CLI. As a final step, turn terminal confirmation back on. For example:

```
bstnA(gbl-cifs[ac1.medarch.org])# terminal confirmation
bstnA(gbl-cifs[ac1.medarch.org])# ...
```

Changing the Proxy User Password

For security reasons, you may choose to change the password for the proxy user account from time to time. For a Windows proxy user, you must make this change in two places: in Active Directory, where the corresponding Windows user is defined, and in the ARX configuration.

Follow this procedure to make the change:

1. Change the password on the Domain Controller (DC).
2. Wait for all of the DCs in the proxy user's Windows Domain to synchronize their databases.
3. After the DCs are synchronized with the new password, re-run the *user (gbl-proxy-user)* command to change the password on the ARX.

During the time that the proxy-user password on the ARX does not match the one in the external Active Directory, the volumes that use the proxy user cannot access any of their back-end-CIFS storage. This prevents the managed volumes from performing autonomous operations, such as policy-driven migrations, and causes the CIFS shares to go offline. F5 recommends that you perform this procedure during non-busy hours.

12

GUI Maintenance

Overview

The GUI has an unsigned SSL key that it uses to authenticate with HTTPS clients. This key is kept in a *keystore* file in the configs directory, "acopia.keystore." When your browser first connects to the GUI and encounters this key, it issues a challenge similar to the following:

By clicking Yes, you accept the ARX's SSL key. Your client then recognizes the SSL key in future HTTPS connections.

The SSL key expires in one year. After the key expires, browsers warn that the key has expired. Use the instructions in this chapter to regenerate the SSL key.

Removing the SSL Key

The SSL key in stored in the "acopia.keystore" file in the configs directory. You can use the show configs command to view the contents of the configs directory, and you can use delete configs acopia.keystore to remove it. For example:

```
bstnA# show configs

  configs
      acopia.keystore         Aug  9 11:20  1.2 kB
      active.license          Sep 15 15:19  3.4 kB
      api-ssl.crt             Aug  9 12:49  1009 B
      api-ssl.key             Aug  9 12:49  891 B
      arx.dossier             Sep 15 15:19  3.0 kB
      boot-config             Sep 16 00:46  1.5 kB
      eeprom.dat              Sep 16 00:13  119 B
      encoded.license         Sep 15 15:19  3.5 kB
      nlad_smb.conf           Sep 16 00:13  44 B
      nlad.conf               Sep 16 00:43  24 B
      oem-menu-log.txt        Sep 16 00:07  3.0 kB
      omDbVersion.info        Sep 16 00:11  280 B
      startup-config          Sep 16 01:31  7.3 MB
      unparsed.license        Sep 15 15:19  3.4 kB

bstnA# delete configs acopia.keystore
bstnA# show configs


  configs
```

```
      boot-config             May 18 03:25  1.3k
      omDbVersion.info        May 18 03:00   279
      running                 May 17 10:25  7.5k
      startup-config          May 18 11:28  6.2M

bstnA# ...
```

Restarting the GUI

You must stop and restart the GUI to create a new SSL key. You can accomplish this by stopping any management access to HTTPS, then re-opening access. Specifically, use the management access https command to work with HTTPS, then use no permit all followed by permit [vlan | mgmt | all]. For detailed instructions, refer to *Permitting Access*, on page 8-5 of the *ARX® CLI Network-Management Guide*.

For example, the following command sequence stops and restarts the GUI, then shows that the GUI recreated the "acopia.keystore" file:

```
bstnA# config
bstnA(cfg)# management access https
bstnA(cfg-mgmt-access[HTTPS])# no permit all
bstnA(cfg-mgmt-access[HTTPS])# permit all
bstnA(cfg-mgmt-access[HTTPS])# show configs


 configs
      wwmed.keystore         Aug  9 11:20  1.2 kB
      active.license          Sep 15 15:19  3.4 kB
      api-ssl.crt             Aug  9 12:49  1009 B
      api-ssl.key             Aug  9 12:49  891 B
      arx.dossier             Sep 15 15:19  3.0 kB
      boot-config             Sep 16 00:46  1.5 kB
      eeprom.dat              Sep 16 00:13  119 B
      encoded.license         Sep 15 15:19  3.5 kB
      nlad_smb.conf           Sep 16 00:13  44 B
      nlad.conf               Sep 16 00:43  24 B
      oem-menu-log.txt        Sep 16 00:07  3.0 kB
      omDbVersion.info        Sep 16 00:11  280 B
      startup-config          Sep 16 01:31  7.3 MB
      unparsed.license        Sep 15 15:19  3.4 kB

bstnA(cfg-mgmt-access[HTTPS])# ...
```

13

Powering Down the ARX

Overview

You can power down the ARX-500, ARX-2000, or ARX-4000 with a CLI command. This cuts power from all systems except the Non-Volatile RAM (NVRAM). The NVRAM contains namespace metadata information that managed volumes are currently using. Other platforms require a manual power-off sequence, which cuts power from all systems including the NVRAM. Once the NVRAM loses power from an external source, it uses a battery backup for up to 72 hours.

This chapter describes how to power down a single ARX and a redundant pair of them. This prepares the ARX pair for a planned power outage.

For power outages of greater than 72 hours, contact F5 Support.

◆ Note

The ARX-VE is a software-only virtual appliance, and the instructions provided here for turning off power, turning on power, and checking the NVRAM battery are not relevant to its operation. Instead, you should refer to the user documentation for your third-party hardware for relevant instructions related to power-down and power-up. However, the software practices described here, including saving configuration parameters and verifying services and overall system health after power has been restored, are relevant to all versions of the ARX, including ARX-VE.

Saving Configuration Parameters

Save a copy of the ARX configuration. The configuration parameters are divided into two classes: *running-config* and *global-config*. The running-config has all of the chassis, network, and redundancy parameters; all of the CLI commands for running-config are under “cfg” mode. The global-config contains all namespaces, volumes, global servers, policy rules, and other storage parameters. The CLI commands for global-config are under “gbl” mode. Save a copy of both the running-config and the global-config. An earlier chapter described all of this: recall *Backing Up the Running Configuration*, on page 5-1.

Use the copy running-config and copy global-config commands to copy the files off of the switch. For example, the following command sequence copies the running-config and global-config off of the “bstnA” switch to a remote FTP server:

```
bstnA# copy running-config ftp://juser:jpasswd@172.16.100.183//var/a6k-running.cfg
% INFO: The copy command completed successfully.
bstnA# copy global-config ftp://juser:jpasswd@172.16.100.183//var/a6k-global.cfg
% INFO: The copy command completed successfully.
bstnA# ...
```

If the ARX has a redundant peer, copy the running-config off of the peer, too. You do not need to copy the global-config, which is shared between the peers. For example,

```
bstnB# copy running-config ftp://juser:jpasswd@172.16.100.183//var/a6kB-running.cfg
% INFO: The copy command completed successfully.
bstnB# ...
```

Checking the NVRAM Battery

The Non-Volatile RAM (NVRAM) memory contains namespace metadata information that managed volumes are currently using. If this memory is lost, all managed volumes must re-import all of their back-end shares. To protect against this time-consuming process, the NVRAM chip has a separate battery backup that lasts up to 72 hours.

Before you shut down any services, use the show chassis nvr command to verify that the battery is in good working order. The output must indicate that the battery is in a “Good” state and that the Error-Correction Circuitry (ECC) indicates that the NVRAM has “No Error.” Run this on both peers if you have redundancy configured.

For example, this output indicates that the “bstnA” chassis has a battery that is ready for power-off:

```
bstnA# show chassis nvr

 NVR:
 NVR Battery  ECC State               NVR Size (MB)
 -----------  ----------------------  -------------
 Good         No Error                2048
bstnA#
```

This example shows that the redundant peer, “bstnB,” also has a fully-charged battery:

```
bstnB# show chassis nvr

 NVR:
 NVR Battery  ECC State               NVR Size (MB)
 -----------  ----------------------  -------------
 Good         No Error                2048
bstnB#
```

Important

Contact F5 Support if your output does not match the above NVR Battery *and* ECC State.

Turning Off the Power

By this time, all ARX clients should be aware of the impending service outage. In a redundant pair, turn the power off at the backup switch first, then the active switch. On an ARX-500, ARX-2000, or ARX-4000, you can go to priv-exec mode and use the shutdown command:

```
shutdown
```

The CLI prompts for confirmation before shutting down the power; enter **yes** to proceed with the shutdown.

For example, this shuts down all power on the "bstnA" switch:

```
bstnA# shutdown

This command turns off the chassis and powers it down.
You will need to manually restore power to return the chassis to
service.
Are you sure? [yes/no] yes
...
```

After you turn off the power, the battery for the NVRAM lasts up to 72 hours.

Manually Turning Off the Power

If you are on site, you can shut down the ARX manually. Press the power button or flip the power switch(es) on the chassis. These are located in different places on each platform:

- ARX-500 - a button on the right side of the front panel (see the *ARX®–500 Hardware Installation Guide*).
- ARX-1500 - a switch toward the left side of the rear panel (see the *ARX®–1500 Hardware Installation Guide*).
- ARX-2000 - a button on the upper-left of the front panel (see the *ARX®–2000 Hardware Installation Guide*).
- ARX-2500 - a switch toward the left side of the rear panel (see the *ARX®–2500 Hardware Installation Guide*).
- ARX-4000 - a button on the upper-left of the front panel (see the *ARX®–4000 Hardware Installation Guide*).

Limited Down Time

As mentioned above, the NVRAM battery drains in 72 hours. If you do not restore power in that time, all managed volumes will re-import their back-end shares after you power on. Depending on the size of the directory trees in these shares, import can be a time-consuming process.

Restoring Power

To watch the boot-up messages as the system powers up, you can connect a serial line to the ARX's Console port. Each Hardware Installation guide shows the location of this port, a 9600-baud, 8-N-1 terminal.

For any chassis type, you turn the power back on with the power button or switch(es). The boot sequence takes more than one minute to complete. After the boot sequence finishes, the Console prompts for authentication:

```
...

User Access Authentication

Username:
```

Verifying Successful Power-Up

Use the CLI to verify that all services are back online.

Showing Overall Health

The show health command should not indicate any open alarms. For example, this system is in good overall health:

```
bstnA(cfg)# show health

System Health Information
Date                  ID      Event                       Description
-------------------------------------------------------------------------------
Fri Jul 21 03:54:38   (0)   - No active alarms.

bstnA(cfg)# ...
```

Repeat this command on the redundant switch if you have one. This example shows that the peer switch is also in good overall health:

```
bstnB(cfg)# show health

System Health Information
Date                  ID      Event                       Description
-------------------------------------------------------------------------------
Fri Jul 21 03:55:08   (0)   - No active alarms.

bstnB(cfg)# ...
```

Showing Namespace Status

The show namespace status all command shows the state of all namespaces, volumes, and shares. This is described in the *ARX® CLI Storage-Management Guide*; see *Monitoring the Import*, on page 9-56. The "Status" for each volume should be "Enabled." Each share should have a Status of "Online;" if the switch or switches were powered off for more than 72 hours, they may be "Pending" or "Importing" for some time before they transition to an "Online" state.

For example, the namespaces on “bstnA” have all volumes enabled and all shares online:

```
bstnA(cfg)# show namespace status all

Namespace: medco
Description:

     Share                     Filer                               Status
         NFS Export
    ------------------------ ----------------------------------- -----------

  Volume: /vol                                                   Enabled
     corporate                 nas1                                Online
         NFS: /vol/vol1

     sales                     nas2                                Online
         NFS: /vol/datavol1/direct

     generic                   nas3                                Online
         NFS: /vol/vol2/direct

Namespace: wwmed
Description:

     Share                     Filer                               Status
         NFS Export
    ------------------------ ----------------------------------- -----------

  Volume: /acct                                                  Enabled
     budget                    das1                                Online
         NFS: /exports/budget

     bills                     das8                                Online
         NFS: /work1/accting

     bills2                    das3                                Online
         NFS: /exports/acct2

     it5                       das7                                Online
         NFS: /lhome/it5

     metadata-share            nas1                                Online
         NFS: /vol/vol1/meta1

Namespace: medarcv
Description:

     Share                     Filer                               Status
        CIFS Share
    ------------------------ ----------------------------------- -----------

  Volume: /lab_equipment                                         Enabled
     equip                     nas10                               Online
        CIFS: equipment

     backlots                  fs2                                 Online
        CIFS: backlot_records

     leased                    nas10                               Online
        CIFS: for_lease
```

```
     scanners                  fs5                                  Online
        CIFS: xraysScanners

     metadata-share            nas1                                 Online
         NFS: /vol/vol1/meta6

     [rs] equipSnap            nas11                                Online
        CIFS: equipBkup

     [rs] leasedSnap           nas11                                Online
        CIFS: leasedBkup

[rs] : replica-snap share

  Volume: /rcrds                                                  Enabled
     rx                        fs4                                  Online
        CIFS: prescriptions

     charts                    fs1                                  Online
        CIFS: histories

     bulk                      fs2                                  Online
        CIFS: bulkstorage

     metadata-share            nas1                                 Online
         NFS: /vol/vol1/meta3

  Volume: /test_results                                           Enabled
     chemistry                 fs1                                  Online
        CIFS: chem_results

     hematology                fs3                                  Online
        CIFS: hematology_results

     2005_charts               medarcv                              Online
        Path: /rcrds

Namespace: insur
Description:

     Share                     Filer                                Status
        CIFS Share
         NFS Export
    ------------------------ ------------------------------------ -----------

  Volume: /claims                                                 Enabled
     shr1-old                  nas1                                 Online
        CIFS: insurance
         NFS: /vol/vol1/NTFS_QTREE/insurance

     metadata-share            nas1                                 Online
         NFS: /vol/vol1/meta2

     shr1-next                 nasE1                                Online
        CIFS: patient_records
         NFS: /root_vdm_4/patient_records

bstnA(cfg)# ...
```

Showing Front-End Service Health

Use show virtual service to verify that all NFS and/or CIFS services are running properly. In a redundant pair, these services only run on the active peer. All services should have a state of “Ready.” For example, this ARX has all of its services running:

```
bstnA(cfg)# show virtual service
Switch:  bstnA
------------------------

  Global Server                                Virtual IP Address   Service    State
  ------------------------------------------------------------------------------------------
  acopiaFiler                                  192.168.25.12        NFS        Ready
  ac1.MEDARCH.ORG                              192.168.25.15        NFS        Ready
  ac1.MEDARCH.ORG                              192.168.25.15        CIFS       Ready

bstnA(cfg)# ...
```

Showing a Healthy Redundancy Configuration

For a pair of redundant switches, check the redundancy configuration from either peer. Use show redundancy all, and check for the status shown in the example below (important status is highlighted in bold):

```
bstnA(cfg)# show redundancy all
                                  Transitions
Node Switch/Quorum Disk   Status    Role    Total Last (UTC)
---- -------------------- --------- ------- ----- -------------------
*1   bstnA                Up        Active  Never -
 2   bstnB                Up        Backup  1     07:37:08 07/21/2006
 QD  192.168.25.234       Up        Quorum  1     07:36:57 07/21/2006

Date/Time(UTC) Recent History
-------------------------------------------------------------------------
07-21 07:37:13 Quorum disk is online, system is ready for failover
07-21 07:37:08 Pair status changed from JoinWait to Formed
07-21 07:37:08 Peer switch 'bstnB' is now online
...
```

14

ARX Disaster Recovery

Overview

An ARX cluster comprises two ARXes that are paired in a redundant configuration for high-availability. The ARX supports disaster recovery for ARX clusters that reside at geographically separate sites. Disaster recovery works by failing over services, volumes, and shares from an active ARX cluster that is unable to continue operating to a backup ARX cluster at a different site that has been prepared to take over for the active ARX cluster and waits passively until it is activated explicitly. The failover process executes efficiently to ensure a short recovery time, enabling ARX clients to continue accessing their volumes and shares with only a minimal interruption of service. The feature requires that file servers or other back-end storage arrays provide the replication of user data from the active site to the backup site.

All ARXes used in a disaster recovery configuration must be the same model and must be running the same revision of the ARX software.

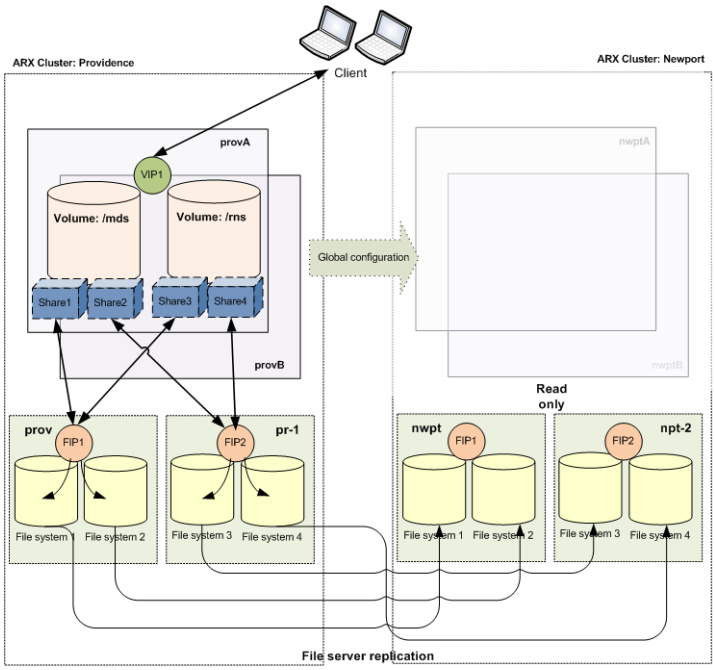
ARX disaster recovery consists of the following high level steps:

1. Ensure that both the active and the backup ARX clusters are configured for networking (via the running-config), and that the active ARX cluster is configured with your current storage configuration (via the global-config).
2. Define a name for both the active and backup ARX clusters; this is simply a name given to the ARX HA pair at the corresponding site.
3. Set site-specific parameters in the active ARX cluster's global-config. The active ARX cluster's global-config will contain many common configuration elements, but it contains also a small number of cluster-specific configuration elements that accommodate site-specific variations, such as differences in the IP addresses at the two sites. This global-config that has been set with both active and backup site parameters is a "shared" global-config.
4. Set up a configuration replication task for replicating the active ARX cluster's global-config to the ARX cluster at the backup site. The configuration replication occurs according to a user-configured schedule. This ensures that a recent copy of the shared global-config resides on the backup ARX cluster.
5. Execute the procedure for performing a failover from the active cluster to the backup cluster. Failover from the active cluster to the backup cluster occurs when you initiate the failover process explicitly, and load and activate the shared global-config on the backup cluster. These steps must be coordinated with other data center failover steps, such as failing over between file servers in the active and backup data centers.

This chapter describes the process of configuring two ARX clusters for disaster recovery. The examples provided throughout this chapter describe the configuration of an active cluster named "Providence", comprising the

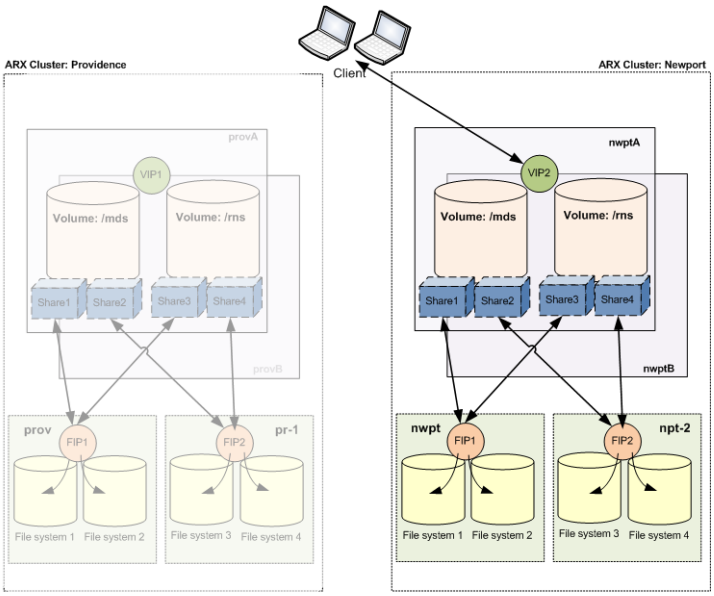
ARXes “provA” and “provB”, and a backup cluster named “Newport”, comprising the ARXes “nwptA” and “nwptB”. The global configuration defines two managed volumes, /mds and /rns, which are exported from the filers “prov” and “pr-1” at the Providence site and from the filers “nwpt” and “npt-2” at the Newport site.

The first figure below illustrates the disaster recovery configuration, with the Providence cluster active and the Newport cluster configured as its backup. Note the replication of client data from the file servers at the active site to those at the backup site, and also the replication of the ARX shared global-config.



The second figure illustrates an actual disaster recovery situation, in which the Providence cluster is unable to continue operating, and the ARX administrator has executed the failover process for both the filer servers and

the ARX clusters. The shared global-config has been loaded and activated on the Newport cluster, and the Newport ARXes now provide access to the client data.



Throughout the examples in this chapter, substitute host names and IP addresses as appropriate to mimic the example configuration in your own network.

Note

This chapter uses the terms "active" and "backup" to indicate which cluster or site is being described, and uses the term "current" when referring to the cluster or site for which you are executing actions to carry out a given instruction. However, be aware that the ARX product interface may refer to a cluster or site as being "local" (the active cluster or site, for which you execute most actions) or "remote" (the backup cluster or site, which waits ready to take over for the active cluster or site).

Before You Begin

This section describes several preliminary tasks that must be completed to prepare ARX clusters for disaster recovery.

Preparing Back-End Filers

Begin by using filer-replication applications (such as NetApp's SnapMirror, EMC's SRDF, or RoboCopy) to duplicate the storage from an active site to a backup site. Continuously replicate all of the back-end shares behind the ARX. Also create back-end shares for managed-volume metadata at the backup site, but do not replicate the source site's metadata. The ARX software at the backup site will recreate the metadata there.

Verifying That Master Keys Are Identical

Ensure that all of the ARXes at both sites use the same master key. The ARXs' master keys must be identical in order for the ARXes to share a global-config. The best practice is to synchronize the master keys at installation time, as described in the ARX Hardware Installation guides.

Execute the show master-key command on each ARX to display an encrypted version of its master key. If the backup switches have the wrong master key, reset them to their factory default configurations and change the master keys to the appropriate values.

An example of the output from the show master-key command follows; note that you must type the system password and the wrapping password for the chassis when prompted to do so:

```
provA(cfg)# show master-key
System Password: %uper$ecretpw
Wrapping Password: an0ther$ecretpw
Validate Wrapping Password: an0ther$ecretpw
Encrypted master key:
2oftVCwAAAAgAAAApwazSRFd2ww/H1pi7R7JMDZ9SoIg4WGA/XsZP+HcXjsIAAAADDRbM
CxE/bc=
provA(cfg)#
```

For detailed information about using this command, refer to the show master-key description in the *ARX CLI Reference*.

You must clear your entire configuration to reset your master key.

Follow these steps to reset an ARX back to its factory defaults and reset its master key:

1. Use the copy running-config command to copy the entire running configuration (network-level parameters, not storage parameters) into a file on the chassis.
2. The ARX should not be running any storage services if it is designated as a backup; a backup is designed to take over services from the active switch or cluster after a failover. Use the remove

service command to clear each service that is running currently, if any. This cleanly de-couples the ARX software from all back-end filers.

3. Connect a serial cable to the Console port. Only the Console port is available after you execute the next step.
4. Use the delete startup-config and delete configs boot-config commands to delete the entire configuration, and then execute the reload command. This resets the ARX to its factory defaults, disables all management IP interfaces, and reruns the initial-boot script at the Console port.
5. Use the initial-boot script to reset the master key. For detailed instructions on the initial-boot process, refer to the hardware installation guide for your chassis.
6. Invoke the run command on the running-config file that you saved onto the chassis earlier. This re-establishes all of your network parameters.

Removing Existing Global Configuration From the Backup ARXes

At the beginning of the disaster recovery configuration process, the ARXes at the backup site should have only their master keys and their network parameters configured (i.e., the running-config); no global-config should be present. This is not typically an issue when configuring brand new ARXes for the first time, but may be a necessary consideration if you are using ARXes that have been deployed and configured previously.

The privileged-exec mode CLI command clear global-config removes the global-config.

For example, the following command deletes the global-config for the current ARX cluster:

```
provA# clear global-config
```

Configuring a RON Mesh

All ARXes in both clusters must be configured with RON tunnels to all of the other ARXes, creating a full mesh. Use the ron tunnel CLI command to connect both ARXes at the active site to both ARXes at the backup site.

For example, the following series of commands creates a RON tunnel named “toNewport” from the ARX named ProvA to the ARX with an IP address of 192.168.8.106 at the Newport site.

```
provA# config
provA(cfg)# interface vlan 103
provA(cfg-if-vlan[103])# ron tunnel toNewport
provA(cfg-if-vlan-ron-tnl[103~toNewport])# peer address 192.168.8.106
provA(cfg-if-vlan-ron-tnl[103~toNewport])# no shutdown
provA(cfg-if-vlan-ron-tnl[103~toNewport])# exit
provA(cfg-if-vlan[103])#
```

Repeat this command sequence for the other side of the tunnel, from the Newport ARXes to the Providence ARXes.

Disaster Recovery Configuration

This section describes the tasks you need to perform to configure the ARXes for disaster recovery.

Disaster Recovery Configuration Summary

Configuring ARX clusters and preparing for disaster recovery comprises the following tasks, each of which is described in its own section:

- Defining Cluster Names
- Adding the Backup Cluster's Filers to the Active Cluster's Configuration
- Assigning SAM-Reference Filers
- Assigning Metadata Shares
- Assigning Volume Shares
- Assigning VIPs For Each Cluster
- Adding the Backup Cluster's Filers to the CIFS Service's Delegate-To List
- Replicating the Global Config to the Backup Cluster
- Manual Failover Tasks
- Loading a Configuration on the Backup Cluster
- Activating a Configuration on the Backup Cluster

Defining Cluster Names

Use the cluster-name CLI command to specify a name for each cluster and the ARXes that belong to it. Each cluster must be named uniquely.

The command syntax is:

[no] cluster-name ***name*** {member ***switch-name*** member ***switch-name***}

Where *name* is the cluster you are defining, and *switch-name* is the name of an individual ARX that will belong to the cluster.

For example, the following command creates an ARX cluster named "providence" that comprises the ARXes "provA" and "provB":

```
provA(gbl)# cluster-name providence member provA member provB
```

The configuration supports the use of two clusters at a time.

Changing a Cluster Name

To change an existing cluster name, you must remove the cluster name from the global configuration first. Use the remove cluster-config command in privileged exec mode to do this.

The command syntax is:

remove cluster-config *clustername*

where *clustername* is the name of the cluster for which you want to remove the name from the global configuration. If you specify "localcluster" as the cluster name, the name of the current cluster is removed from the global configuration.

For a remote cluster, this removes the following objects from the system:

- any filer
- any metadata share
- any virtual server that resides in the cluster

For the local cluster, this removes only the cluster name from the objects listed above. They remain in the configuration, but without the cluster name as part of their definitions.

The CLI prompts for confirmation before removing or changing any configuration objects; type "yes" to proceed.

After the cluster name has been cleared, you can define a new cluster name as described in the previous section.

Adding the Backup Cluster's Filers to the Active Cluster's Configuration

In order for all client data to be accessed reliably and transparently by the ARXes at both sites, the filers at the backup cluster's site must be included in the active cluster's configuration. (This is part of the global configuration that is in turn replicated from the active site to the backup site.) For complete instructions for doing this, refer to the *ARX® CLI Storage-Management Guide*, *Appendix 6, Adding an External Filer*.

For example, the following commands show the addition to the Providence configuration of two filers at the Newport site, named "nwpt" and "npt2". The name and IP address of each is specified, along with a brief description, the management protocol to be used, and the proxy user ID for each. In addition, snapshot support is configured for each. The exit command at the end of each command sequence quits gbl-filer mode.

For example, the following commands configure the addition of the filer "nwpt":

```
provA(gbl)# external-filer nwpt
provA(gbl-filer[nwpt])# ip address 172.16.200.1
provA(gbl-filer[nwpt])# description "big filer in Newport's front lab"
provA(gbl-filer[nwpt])# filer-type network-appliance management-protocol rsh
provA(gbl-filer[nwpt])# proxy-user unxAdmin
provA(gbl-filer[nwpt])# manage snapshots
provA(gbl-filer[nwpt])# exit
```

The following commands configure the addition of the filer "npt2":

```
provA(gbl)# external-filer npt2
provA(gbl-filer[npt2])# ip address 172.16.200.2
provA(gbl-filer[npt2])# description "New filer, rack 4 (Newport)"
provA(gbl-filer[npt2])# filer-type network-appliance management-protocol rsh
```

```
provA(gbl-filer[npt2])# proxy-user unxAdmin
provA(gbl-filer[npt2])# manage snapshots
provA(gbl-filer[npt2])# exit
```

Assigning SAM-Reference Filers

If you are using CIFS local groups, you may want to assign a security account management (SAM) reference filer per site. Use the sam-reference command with its cluster argument to specify the cluster with which the external filer is associated. The command syntax is:

sam-reference *filer* cluster *clustername*

where *clustername* specifies the cluster with which *filer* is associated.

◆ Note

Execute the sam reference *command twice when configuring for disaster recovery: once to specify the SAM reference filer at the active cluster, and again to specify it at the backup cluster.*

Refer to *Selecting a SAM-Reference Filer* in the *ARX® CLI Storage-Management Guide* for complete information about SAM-reference filers and the use of this command.

For example, the following commands specify pr-1 as the SAM-reference filer for the Providence cluster, and npt2 as the SAM-reference filer for the Newport cluster:

```
provA(gbl-ns[provMed])# sam-reference pr-1 cluster providence
provA(gbl-ns[provMed])# sam-reference npt2 cluster newport
```

Assigning Metadata Shares

For each managed volume in the configuration, you need to define metadata shares for each site. Use the metadata share command to specify a metadata share for use with a particular cluster. The command syntax is:

metadata share *filer* {nfs3 | nfs3tcp | cifs} *path* [cluster *cl-name*]

where *filer* is the name of the external filer, *path* is the share path on the filer, and *cl-name* is cluster that will use filer.

◆ Note

Execute the metadata share *command twice per volume when configuring for disaster recovery: once to specify the metadata share's host at the active cluster, and again to specify the metadata host at the backup cluster. The two metadata shares need to use the same protocol.*

Refer to *Storing Volume Metadata on a Dedicated Share* in the *ARX® CLI Storage-Management Guide* for complete information about metadata shares.

For example, the following commands specify the path to the metadata share on filer pr-1 for the Providence cluster, and the path to the metadata share on filer npt2 for the Newport cluster.

```
provA(gbl-ns[provMed])# volume /mds
provA(gbl-ns-vol[provMed~mds])# metadata share pr-1 nfs3 /vol/vol3/arx_meta cluster providence
provA(gbl-ns-vol[provMed~mds])# metadata share npt2 nfs3 /vol/vol3/arxMeta cluster newport
```

Assigning Volume Shares

For each managed volume share configuration, you need to define per site configuration. For each ARX share, a filer and share can be configured per cluster. Assign the shares to a filer and cluster using the filer command. The command syntax is:

filer *filer* {nfs *sharename* | cifs *sharename*} [access-list *listname*] [cluster *clustername*]

where *filer* is the name of the filer that hosts the share in question, nfs *sharename* or cifs *sharename* is the share to be associated with the cluster, *listname* is the ACL associated with the share, and *clustername* is the cluster to which this configuration is relevant.

◆ Note

Execute the filer *command twice per share when configuring for disaster recovery: once to specify the share's filer at the active cluster, and again to specify the share's filer at the backup cluster.*

For example:

```
provA(gbl-ns-vol[provMed~mds])# share mds
provA(gbl-ns-vol-shr[provMed~mds~mds])# filer pr-1 cifs MDS cluster providence
provA(gbl-ns-vol-shr[provMed~mds~mds])# filer npt2 cifs MDS cluster newport
provA(gbl-ns-vol-shr[provMed~mds~mds])# exit
provA(gbl-ns-vol[provMed~mds])# share rns
provA(gbl-ns-vol-shr[provMed~mds~rns])# filer prov cifs RNS cluster providence
provA(gbl-ns-vol-shr[provMed~mds~rns])# filer nwpt cifs RNS cluster newport
provA(gbl-ns-vol-shr[provMed~mds~rns])# exit
```

Assigning VIPs For Each Cluster

Each ARX cluster may have a unique virtual IP address (VIP) assigned to it to be accessed by ARX clients. In some disaster recovery configurations, a per site VIP may be required. The command syntax is:

virtual server *switchname vip mask* [vlan *vlan-id*] [cluster *clustername*]

Where *switchname* is the name of the network switch, *vip* is the IP address assigned to the virtual server, *mask* is the network subnet mask associated with that IP address, *vlan-id* is the VLAN on which virtual server is a node, and *clustername* is the cluster to which this configuration is relevant.

For example, the following command specifies the cluster name "providence" for the virtual server 192.168.103.55, with VLAN 103, on the ARX named "provA":

```
provA(gbl-gs[provmed.MEDARCH.ORG])# virtual server provA
192.168.103.55 255.255.255.0 vlan 103 cluster providence
```

Note

All configuration in global server mode is shared between virtual servers.

Adding the Backup Cluster's Filers to the CIFS Service's Delegate-To List

If you are running CIFS services with constrained delegation, some additional configuration of the domain controller will be necessary to support your CIFS clients. The back-end CIFS servers at the backup cluster must be on the "delegate-to" list for each front-end CIFS service. For each CIFS service, execute the probe delegate-to command to find any back-end servers that need to be added to the list, and add them at the domain controller. When complete, the CIFS service's "delegate-to" list includes the back-end servers behind both clusters. This configuration is performed at the DC, and persists after the "load-configs" test.

Replicating the Global Config to the Backup Cluster

Replication of the global configuration from the active cluster to a backup cluster is controlled by a user-configured replication task. Define the replication task using the global-config mode CLI command config-replication. This command creates a task that replicates the active ARX's global-config by a user-defined schedule to the target ARX cluster. The source ARX will use the cluster name to resolve the target ARXes from the configured target cluster. The best RON route will be used for reaching the switches in the target cluster.

Replication begins when it is enabled, and occurs according to the frequency and times defined in the associated schedule. During the process, a copy of the global configuration is replicated to both members of the target cluster, into the configs directory on the system disks of the target switches. The filename of the global configuration on the target switch will be whatever the user specified in the replication task configuration. If the target filename exists already on the target switch, the replication will ensure transactionally that it is not overwritten until the latest configuration replication is complete.

Executing the config-replication command creates the config-replication object and enters gbl-cfg-repl mode. The command syntax is:

config-replication *name* cluster *clustername*

Where *name* is the replication task you are creating and *clustername* is the ARX cluster from which the global-config originates.

For example, the following command defines a replication task named "prov2newport" which replicates the global configuration from the source cluster Providence to the target cluster Newport:

```
provA(gbl)# config-replication prov2newport cluster providence
```

Subsequently, the CLI is in gbl-cfg-repl mode, enabling you to specify the details of the configuration replication task, such as the target cluster, the target filename, the schedule used to control the frequency at which the replication task runs, and the administrative user account and password used to authenticate to the target cluster.

For example, the following command sequence specifies an ARX administrative user named "admin" with a password of "mypwd"; the target filename is "provSvcs.rcfg", and it is replicated according to the schedule named "oncePerDay" to the target cluster "newport". Additional commands name the prefix of the replication report and define a textual description of the replication object. The enable command puts the replication task into effect.

```
provA(gbl-cfg-repl[prov2newport])# target-cluster newport
provA(gbl-cfg-repl[prov2newport])# user admin
Password: mypwd
Validate password: mypwd
provA(gbl-cfg-repl[prov2newport])# target-file provSvcs.rcfg
provA(gbl-cfg-repl[prov2newport])# schedule oncePerDay
provA(gbl-cfg-repl[prov2newport])# report gblRepl2newport
provA(gbl-cfg-repl[prov2newport])# description "send service config to Newport site"
provA(gbl-cfg-repl[prov2newport])# enable
provA(gbl-cfg-repl[prov2newport])# exit
provA(gbl)#
```

Use the show config-replication command to see the current status of one or more config-replication rules.

Use the show replicated-configs command to list the global configurations that have been replicated to the current ARX.

Refer to *Appendix 12, Creating a Policy Schedule* in the *ARX® CLI Storage-Management Guide* for complete information about defining and using schedules.

Manual Failover Tasks

You must execute the following tasks manually when failing over from the previously active cluster to the backup cluster:

1. Disable the replication of the ARX global configuration by executing the no enable command in gbl-cfg-repl mode.
2. Halt the active ARX cluster's front-end services if the active cluster is still viable. Do this by executing no enable for each CIFS/NFS service and for the global server.
3. Halt file server replication.

4. Fail over the file server infrastructure from the active cluster to the backup cluster. The backup site file server volumes/file systems now are read-write and available.
5. Load the latest replicated global-config from the previously active cluster onto the backup ARX cluster. Refer to *Loading a Configuration on the Backup Cluster* for details.
6. Optionally, perform an Active Directory forest re-discovery to facilitate discovery of the backup cluster's local domain controllers.
7. Activate the configuration that you have just loaded on the backup ARX cluster. Refer to *Activating a Configuration on the Backup Cluster* for details.
8. Your data now will be imported to the backup cluster using seamless import.

Loading a Configuration on the Backup Cluster

The privileged-exec mode CLI command load replicated-configs enables you to load all or part of a replicated global configuration. All objects are loaded in a non-enabled state, regardless of whether they were enabled in the source.

If there are conflicting objects in the global configuration, the objects that exist already in the switch will take precedence. Conflict resolution is described in detail in *Conflict Resolution For Replicated Configurations*, on page 14-18.

The load replicated-configs command always generates a report with all CLI output. The report name is generated automatically and includes a timestamp and the name of the replicated-config filename. The report type name is "Load".

The command syntax is:

load replicated-configs *filename* [global-server *gsname*] [tentative]

Where *filename* is the name of the replicated configuration file and *gsname* is the name of the global server on which the replicated configuration will be loaded.

For example,

```
provA# load replicated-configs provSvcs.rcfg global-server ac1.medarch.org
```

This command loads the global configuration named "provSvcs.rcfg" on the global server named "ac1.medarch.org".

Note

The load *command may be used to load a subset of a replicated configuration based on global-server. This is an advanced option; consult F5 technical support before attempting to use it.*

Testing the Load Configs Operation

After the global-config has been copied successfully to the backup cluster, you should test the use of the load configs tentative CLI command at that cluster from time to time. This performs a test load of the active cluster's storage services onto the backup cluster.

Activating a Configuration on the Backup Cluster

The privileged-exec mode CLI command activate enables you to activate all or part of a loaded replicated global configuration. By default, the activation process enables all global servers, all of their associated CIFS and NFS services, all of their referenced volumes, and all of their referenced volume shares. By default, it does not enable volume policies; they must be enabled explicitly. The activation process activates the loaded configuration to maintain the original enable state of the loaded replicated global configuration.

The activate command provides options for enabling different portions of the loaded configuration. This enables administrators to test each portion of a loaded replicated configuration separately. In addition, the take ownership argument must be appended to the command in order to take ownership of volumes that are marked as being used by other ARXes.

The activate command always creates a report. The report includes output similar to that for the run script command. The report type name is "Act".

The command will prompt for verification, asking the user to confirm that the configuration should be activated.

The command syntax is:

activate replicated-configs *filename* [global-server *gsname*] [shares] [volumes] [global-servers] [take ownership] [policies] [tentative]

Where *filename* is the name of the replicated configuration file and *gsname* is the name of the global server on which the replicated configuration will be activated.

For example,

```
provA# activate replicated-configs provSvcs.rcfg global-server ac1.medarch.org
```

This command activates the replicated configuration named "provSvcs.rcfg" on the global server named "ac1.medarch.org".

◆ Note

Executing this command by specifying a filename without any additional arguments will enable all shares, volumes, and global-servers by default. In order to activate policies, the policies *option must be specified explicitly.*

The best practice is to activate the configuration in stages:

1. Activate shares only, with take ownership.
2. Activate volumes only. This will begin the online import process.

3. Test connectivity to ARX exports and volumes.
4. Activate policies.

Executing ARX Cluster Failback

When failover takes place, the backup ARX cluster becomes the active ARX cluster. The newly active cluster can run for an indefinite period of time, and can have configuration changes made to it. When you want to initiate a failback to the previously active cluster, execute the same actions described in these earlier sections:

- *Manual Failover Tasks*, on page 14-14
- *Loading a Configuration on the Backup Cluster*, on page 14-15
- *Activating a Configuration on the Backup Cluster*, on page 14-16

◆ Note

The configuration of the target switch must be updated to accommodate any global-level configuration changes that were made after the initial failover.

Conflict Resolution For Replicated Configurations

When loading a configuration, a conflict may occur with the existing configuration on the current ARX. If you have followed the instructions provided in this chapter, you should not encounter any configuration conflicts. However, if you have worked with F5 customer support to define an alternative configuration that is not covered by the preceding instructions, and you've encountered configuration conflicts subsequently, this section will assist you in addressing those conflicts.

The general principles of conflict resolution for replicated global configs follow:

- Conflicts are detected based on the name of the object or the name of the command.
- For most conflicts, the object or command that exists on the current switch already takes precedence over the one in the replicated configuration. The exception to this is in the handling of namespaces; this is described in detail in *Resolving Conflicts in Namespaces*, on page 14-19.
- If there is a conflict, the existing object or command on the switch will not be altered in any way.

In general, most potential conflicts occur with objects at the top level of the global configuration (schedules, filesets, etc.).

Resolving Top-Level Conflicts

These conflict resolution principles enable administrators to have preset cluster-specific top-level configuration objects. Cluster-specific top-level objects are pre-configured (e.g., NTLM authentication servers and Active Directory forests). When the configuration is loaded, the conflict resolution allows the local predefined objects to remain.

The top-level objects for which conflicts could occur are:

- active-directory
- active-directory-forest
- external-filer
- file-history
- group
- kerberos
- namespace
- nfs-access-list
- nis
- ntlm-auth-db
- ntlm-auth-server
- policy-age-fileset

- policy-filename-fileset
- policy-filesize-fileset
- policy-intersection-fileset
- policy-union-fileset
- proxy-user
- radius-server
- schedule
- user
- volume group
- windows-mgmt-auth

Resolving Conflicts in Namespaces

The exception in which global configuration replication conflicts are handled differently is that of namespaces. This is because a failover may involve a set of volumes inside a namespace.

Namespaces are handled as follows:

- Conflicting namespace names are allowed.
- If there is a conflict in the namespace name, no namespace-specific parameters in the active configuration will be altered (such as the protocol).
- Conflicts involving volumes are handled as described in the previous section.
- An NFS presentation volume can be configured to attach to a managed volume in another namespace. This case is not addressed in the loading of the configuration.

Index

F

G

I

L

M

N

P

R

S

T

U

V

W