When Bad Things Happen to Good Amanda Backup Servers

By Lois Garcia, Dmitri Joukovski and Pavel Pragin

White Paper

"Even if your backup system makes volumes that can be read by native backup utilities, without a database that identifies what's where, you have no idea what system is on what volume That means that this database has now become the most important database in your company. You need to make sure that it is backed up, and its recovery should be the easiest and most tested recovery in your entire environment."

W. Curtis Preston
"Unix Backup and Recovery"

"That's just perfectly normal paranoia. Everyone in the Universe has that."

Slartibartfast
"The Ultimate Hitchhiker's Guide to the Galaxy" by Douglas Adams

Abstract

This white paper describes methods of devising a multi-layered protection to the Amanda backup server by using Amanda backups and the standard operating systems tool rsync.

Table of Contents

Ensuring the Availability of the Amanda Backup Server 3
Protecting the Amanda Server 3
Granular Protection 4
- The Application Files 4
- The Configuration Files 4
- Methodology 5

Using Rsync to Protect the Amanda Server 5
- Prerequisites 6
- Deployment 7
- Verifying and Restoring 9

Using Amanda Tape Backups to Protect Your Amanda Server 11
- Prerequisites 11
- Deployment 11
- Verifying and Restoring 13

Best Practices 15
Conclusion 16

Please send your comments about this white paper to *feedback@zmanda.com*

Ensuring the Availability of the Amanda Backup Server

After purchasing a new tape library and evaluating several options for backup software you finally designed a backup strategy for your organization. You have spent considerable time on research and testing, you have set up your Amanda backups to reliably run according to your backup policy. The Amanda backup server has gone through a couple of dump cycles and after several tests you are confident you can reliably restore files from your backups. Your data recovery procedures are in a safe place, and you have accomplished what you had in mind ... or have you?

What happens if your backup server is compromised? Multiple surveys consistently show that a major percentage of data loss is caused by hardware or system malfunctions (40% to 60%) and by human error (20% to 30%) such as accidental file deletion. The backup server is as vulnerable to these types of outages as is any other server. To comprehensively protect against data loss, a well-executed backup plan must provide coverage for all types of situations. Your Amanda backup server itself must be easily recoverable and you should test the disaster recovery of your backup server several times under different scenarios.

Since your Amanda backup server protects all the rest of your systems, that backup server could be considered the most important server in your organization and should have more than one layer of protection. The purpose of this white paper is to describe methods of devising multi-layered protection for the Amanda backup server by using Amanda backup software and the standard operating systems tool rsync.

Protecting the Amanda Server

Whether through hardware failure or file system corruption, loss of the Amanda configuration files, file indexes, and other dynamic records that reside on the backup server would not only interrupt your backup schedule but also make it more difficult to recover files from all computers protected by Amanda. Of course, since Amanda does not use any proprietary formats, you can always recover your files even without having Amanda installed, but a working installation of Amanda makes file restores so much easier. For more information about Amanda, please review the white papers and technical presentations about Amanda available to subscribers of Zmanda Network.

Amanda depends on the configuration files to control backup media, to calculate dump cycles, to know exactly what is to be backed up, to send email to the correct address, where to find a file for restore, and for other information relevant to your environment. The Amanda **curinfo** and **index** directories contain information that is updated at the time of every backup. Depending on the degree of customization and complexity in your backups and your environment, returning to your regular backup schedule could mean a considerable loss of productivity. Loss of regular backup capability might also leave the enterprise out of compliance with increasingly stringent state and federal regulations.

To protect the Amanda server, the first recommendation is to always make the server an Amanda client and add the server to your disk list entries (DLEs). Installing an Amanda client on the backup server and backing up the server's file system ensures the server's future availability. You have the ability to restore the server to its state at varying points in its history, for example, to its state before a disk crash. Amanda can communicate with standard operating system tools such as tar and dump, so you can perform a restore from backup media without Amanda software. Restore the Amanda server first, then restore the remainder of your data with the convenience of the Amanda interface.

Granular Protection

The Amanda server architecture consists of the base application and its associated configuration files. During the course of a backup cycle, backup logs, database and index files are created that are specific to that backup cycle and that are critical for Amanda's functionality.

The Application Files

The original software packages you used to install Amanda might not always be available when you need them. Even if you are on a timely upgrade schedule, there is no guarantee that when disaster strikes you will be on the latest Amanda release. Each release includes versions of Amanda executables and libraries. Always save a copy of the build that you have currently installed, whether it is a prebuilt package or one that you have built yourself from source.

The Configuration Files

Having your Amanda configuration files available in addition to the base application greatly simplifies data recovery. If you have followed the suggested configuration steps, the **/etc/amandates** file, the **/etc/dumpdates** file, and the **/etc/amanda** and **/var/lib/amanda** directories contain everything needed to return a default Amanda server installation to your individual setup. If you have changed the locations of any critical files, remember to adapt the instructions to conform to your changes.

/etc/amandates and **/etc/dumpdates**
Contain filesystem/directory names, dump levels and dump dates. This information is used to determine the need for incremental backups. The **amandates** file tracks Amanda's use of tar to perform incremental backups, while the **dumpdates** file tracks the use of dump.

/etc/amanda
The **/etc/amanda** directory holds a subdirectory for each backup configuration you have created. In each subdirectory is the **amanda.conf** file that regulates that backup

configuration. You will also find backup log files, the **disklist** file, the **tapelist** files, tape changer configuration files, and the **curinfo** and **index** directories. Both the **curinfo** and **index** directories contain crucial data related to your backups. The contents of the **/etc/amanda** directory change upon every backup and at every backup configuration modification. Note that the dynamic nature of the **curinfo** and **index** databases, as well as the **/etc/amandates** and **/etc/dumpdates** file, and the backup **log** files, affects the ability to use Amanda to backup its own server configuration.

/var/lib/amanda
By default, **/var/lib/amanda** is the home directory of the *amandabackup* user if you install from Amanda RPMs built by Zmanda. This directory holds typical user information, such as the .**ssh** directory containing user authentication details. Another important Amanda file in **/var/lib/amanda** is the hidden file, .**amandahosts**. The .**amandahosts** file facilitates connections between the Amanda backup server and its clients which is similar in function to the .rhosts file for rsh.

Methodology

Given the importance of the Amanda server to the enterprise and the highly stressful conditions surrounding most disaster recovery procedures, a streamlined method to recover the Amanda server is a sensible component of any business continuity planning. With the Amanda server file system backed up as part of your regular backup plan, the first level of the protection scheme for your Amanda server is accomplished. The next level is to protect just the configuration files. Keep in mind that some Amanda configuration files, e.g. logs, change when you backup the Amanda server itself. To get exact copies of all of your Amanda configuration files after a backup, including backup of the server itself, is complete, you can use ***rsync***.

Using Rsync to Protect the Amanda Server

rsync is an open source UNIX/Linux utility which synchronizes files and directories from one location to another while minimizing data transfer using delta encoding when appropriate. ***rsync*** can be used to easily and reliably keep a copy of your Amanda configuration files in a safe location.

In-depth information about ***rsync*** is available from the ***rsync*** website. Here we will provide just a brief overview of ***rsync*** functionality that is relevant to our illustration of how to use it for protecting the Amanda server.

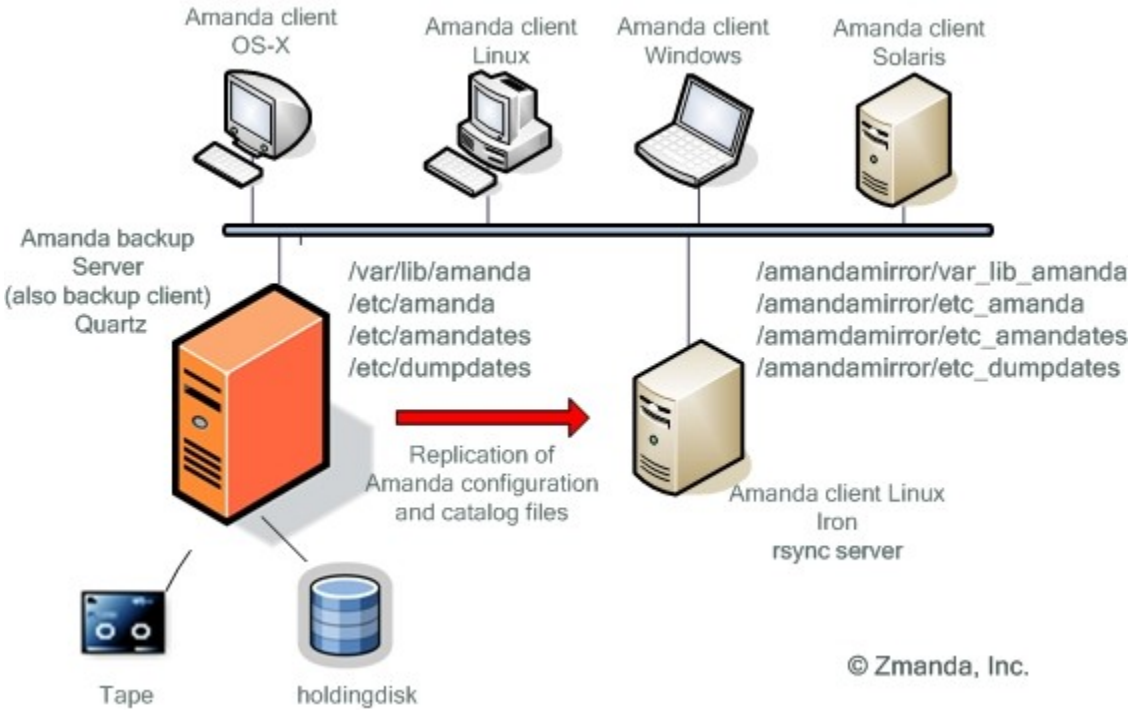
rsync works incrementally in that it copies only the changes to files, minimizing the size and length of transfers. Even though each data transfer is incremental, ***rsync*** keeps only one, up-to-date version of a file. ***rsync*** runs over rsh by default, but can be run over any shell, including ssh, and in daemon mode runs directly over TCP/IP. ***rsync*** has a long list of options that have been developed over time so that you can continue its use as your

needs become more complex.

Some additional features of ***rsync*** are:

- support for copying links, devices, owners, groups, and permissions
- exclude options
- can use any transparent remote shell, including ssh or rsh
- does not require super-user privileges
- support for anonymous or authenticated ***rsync*** daemons

To create and maintain a repository of Amanda configuration files, we will run ***rsync*** from an ***rsync*** server, separate from the Amanda server. We will put ***rsync*** into a script so that we can run the script from a crontab automatically or run the script manually as needed.



Prerequisites

Start with the Amanda server whose configuration you want to backup. We used a server based on Amanda 2.5.1 RPMs built by Zmanda and freely available from the Zmanda downloads page.

In addition to the Amanda server installation, you will need:

- An ***rsync*** server on which to replicate your Amanda configuration files
- ***rsync*** and ***OpenSSH*** installed on the Amanda server and on the ***rsync*** server
- *root* access to the Amanda server and the ***rsync*** server
- Amanda client software installed on the ***rsync*** server. This is to ensure an identical *amandabackup* Amanda user on both machines.

Deployment

The ***rsync*** server will host an ***rsync*** script and will also be the repository of the duplicate Amanda server files. This server can reside anywhere, as long as you can reach it with ssh over a TCP/IP connection.

1. For ***rsync*** to run over a non-interactive secure shell (SSH) connection, we have to generate a public and private key pair for the *amandabackup* user. ***rsync*** connects non-interactively but securely across the network using ssh. If you have already generated ssh keys while under the *amandabackup* user account, you can skip the following ssh-related steps.

   ```
   amandabackup@iron:~> ssh-keygen -t rsa
   ```

   The generated files are saved in the directory /var/lib/amanda/.ssh

   ```
   amandabackup@iron:~> cd ~amandabackup/.ssh; ls -l
   total 16
   -rw------- 1 amandabackup disk  399 2006-12-13 21:10 authorized_k
   eys
   -rw------- 1 amandabackup disk 1671 2006-12-13 21:13 id_rsa
   -rw-r--r-- 1 amandabackup disk  401 2006-12-13 21:13 id_rsa.pub
   -rw-r--r-- 1 amandabackup disk  454 2006-12-13 21:23 known_hosts
   ```

2. Create an authorized_keys file in the .ssh directory in the home directory of the *amandabackup* user on the Amanda backup server machine (Quartz), and append the contents of the public key file, id_rsa.pub, created on Iron. If you have already begun a keyring, add the new key.

   Note that you cannot just transfer and rename the file. The contents of the id_rsa.pub file must be appended to the authorized_keys file. Make sure the permissions on the authorized_keys file are read and write for *amandabackup*, with no permissions for anyone else. Also compare file sizes; both files should be exactly the same.

3. Test the new key pair. The first time you login, you will be prompted to add Iron to the list of known hosts on Quartz.

   ```
   amandabackup@iron:~> ssh quartz.zmanda.com
   The authenticity of host 'quartz (192.168.10.168)' can't be estab
   lished.
   RSA key fingerprint is 07:bf:c0:b7:7a:69:47:6c:a4:4d:31:f8:63:38:
   00:5e.
   Are you sure you want to continue connecting (yes/no)? yes
   Warning: Permanently added 'quartz,192.168.10.168' (RSA) to the l
   ist of known hosts.
   ```

```
amandabackup@quartz:~>
```

4. Return to Iron and create the ***rsync*** directories where the Amanda server configuration files will be mirrored.

```
amandabackup@quartz:~> exit
logout
Connection to quartz closed.
amandabackup@iron:~> mkdir /amandamirror/etc_amanda
amandabackup@iron:~> mkdir /amandamirror/etc_amandates
amandabackup@iron:~> mkdir /amandamirror/etc_dumpdates
amandabackup@iron:~> mkdir /amandamirror/var_lib_amanda
amandabackup@iron:~> ls -l /amanda*
/amandamirror:
total 0
drwxr-xr-x 3 amandabackup disk 72 2006-12-13 21:23 etc_amanda
drwxr-xr-x 2 amandabackup disk 80 2006-12-13 21:26 etc_amandates
drwxr-xr-x 2 amandabackup disk 80 2006-12-13 21:27 etc_dumpdates
drwxr-xr-x 3 amandabackup disk 72 2006-12-13 21:28 var_lib_amanda
```

5. Add the ***rsync*** commands to a script that can be run manually or placed in a crontab.

```
amandabackup@iron:~> pwd
/var/lib/amanda

amandabackup@iron:~> vi sync

#!/bin/bash
rsync -av amandabackup@quartz.zmanda.com:/etc/amanda /amandamirro
r/etc_amanda >> /var/lib/amandabackup/sync.log
rsync -av amandabackup@quartz.zmanda.com:/etc/amandates /amandami
rror/etc_amandates >> /var/lib/amandabackup/sync.log
rsync -av amandabackup@quartz.zmanda.com:/etc/amandates /amandami
rror/etc_dumpdates >> /var/lib/amandabackup/sync.log
rsync -av amandabackup@quartz.zmanda.com:/var/lib/amanda /amandam
irror/var_lib_amanda >> /var/lib/amandabackup/sync.log

amandabackup@iron:~> chmod 700 sync

amandabackup@iron:~> ls -l sync
-rwx------ 1 amandabackup disk 257 2006-12-13 23:08 sync

amandabackup@iron:~>
```

6. The *sync* script should run after, but never during, a backup, because Amanda updates several files during every backup. If you run ***rsync*** before Amanda has completed the backup, your results will be corrupt.

   The following cron example runs *sync* twice a day at 7:30 am and 7:30 pm.

```
amandabackup@iron:~> crontab -e

30 7,19 * * * /var/lib/amanda/sync

crontab: installing new crontab
```

Verifying and Restoring

Let's test our work so far.

1. As the *amandabackup* user, run *sync*.

   ```
   amandabackup@iron:~> ./sync
   ```

2. Look for a **sync.log** file in your home directory as *amandabackup*.

   ```
   amandabackup@iron:~> ls -l
   total 8
   -rwx------ 1 amandabackup disk 344 2006-12-15 00:00 sync
   -rw-r--r-- 1 amandabackup disk 356 2006-12-15 00:00 sync.log
   amandabackup@iron:~> cat sync.log
   receiving file list ... done

   sent 20 bytes  received 950 bytes  1940.00 bytes/sec
   total size is 95931  speedup is 98.90
   receiving file list ... done

   sent 20 bytes  received 90 bytes  220.00 bytes/sec
   total size is 0  speedup is 0.00
   receiving file list ... done

   sent 20 bytes  received 107 bytes  220.00 bytes/sec
   total size is 0  speedup is 0.00
   receiving file list ... done

   sent 20 bytes  received 367 bytes  774.00 bytes/sec
   total size is 41354  speedup is 106.86
   ```

3. The directories created under **/amandamirror** on Iron should now reflect the contents of the directories named in the *sync* script. Backup configurations and indexes should now all be safely mirrored.

   ```
   amandabackup@iron:~> cd /amandamirror

   amandabackup@iron:/amandamirror> ls
   etc_amanda  etc_amandates  etc_dumpdates  var_lib_amanda

   amandabackup@iron:/amandamirror> tree
   .
   |-- etc_amanda
   |   `-- amanda
   |       `-- test
   |           |-- amanda.conf
   |           |-- amdump.1
   |           |-- amdump.2
   |           |-- amdump.3
   |           |-- amdump.4
   |           |-- amdump.5
   |           |-- amdump.6
   |           |-- amdump.7
   |           |-- changer.conf-access
   ```

```
|           |-- changer.conf-clean
|           |-- changer.conf-slot
|           |-- curinfo
|           |   `-- copper.zmanda.com
|           |       |-- _amandadownloads
|           |       |   `-- info
|           |       `-- _boot
|           |           `-- info
|           |-- disklist
|           |-- index
|           |   `-- copper.zmanda.com
|           |       |-- _amandadownloads
|           |       |   |-- 20060714_0.gz
|           |       |   |-- 20060714_1.gz
|           |       |   `-- 20060719_1.gz
|           |       `-- _boot
|           |           |-- 20060719_0.gz
|           |           `-- 20060719_1.gz
|           |-- log.20060714.0
|           |-- log.20060714.1
|           |-- log.20060714.2
|           |-- log.20060719.0
|           |-- log.20060719.1
|           |-- log.20060719.2
|           |-- log.20060719.3
|           |-- oldlog
|           |-- tapelist
|           |-- tapelist.amlabel
|           `-- tapelist.yesterday
|-- etc_dumpdates
|   `-- dumpdates
|-- etc_amandates
|   `-- amandates
`-- var_lib_amanda
    `-- amanda
        |-- example
        |   `-- amanda.conf
        `-- gnutar-lists

17 directories, 31 files
```

4. Replication was successful! To complete the scenario, the true test (of any backup) is a successful restore. The ***rsync*** restores must be run as the *root* user on the Amanda backup server.

5. To understand the directories and files that need to be restored, look in the **/amandamirror** directory on your ***rsync*** server, Iron. The replication directories were named to give a visual clue as to the original location of the directories and file.

```
iron:~ # cd /amandamirror

iron:/amandamirror # ls
etc_amanda  etc_amandates  etc_dumpdates  var_lib_amanda

iron:/amandamirror #
```

6. Then, as the *root* user on your Amanda server, Quartz, issue the corresponding ***rsync*** restore commands.

```
quartz:~ # rsync -a root@iron:/amandamirror/etc_amanda/ /etc/
quartz:~ # rsync -a root@iron:/amandamirror/etc_amandates/ /etc/
quartz:~ # rsync -a root@iron:/amandamirror/etc_dumpdates/ /etc/
quartz:~ # rsync -a root@iron:/amandamirror/var_lib_amanda/ /var/
lib/amanda
```

Using Amanda Tape Backups to Protect Your Amanda Server

In addition to adding the Amanda server filesystem to your regular backup schedule, you can add targeted backups of the Amanda configuration and database files. Targeting these files will ease your recovery efforts, simplifying their accessibility from tape and so shortening the time required to return your Amanda backup server to production. We will isolate these critical Amanda files, similarly to how we targeted the files in the ***rsync*** method, except that in this case we will backup the files to tape.

It's important to note that because Amanda writes to server files during a backup, certain log files cannot be captured by an Amanda backup except in their most recent state prior to the backup. Amanda is still running as it backs up its own log files and database information, so the configuration backup will always be one backup behind. When disaster recovery is performed, the Amanda database will not be as current as the last backup. It will be as current as the backup before the last.

Prerequisites

You need only an Amanda backup server installation with *root* and the Amanda user account access, in our case, *amandabackup*.

Deployment

1. To make sure that every Amanda backup includes a backup of the most recent set of server configuration files, as the *amandabackup* user, add two new dumptypes called *amanda-config-backup* and *include* to each **amanda.conf** file in **/etc/amanda/$CONFIG**. Let's assume just one configuration, called *DailySet1*.

```
define dumptype amanda-config-backup {
   root-tar
   comment "force Level 0 backups for Amanda config files"
   strategy noinc
}

define dumptype include {
   amanda-config-backup
   comment "force Level 0 backups and only backup files specified
```

```
in the include list"
   priority medium
   include list "/var/lib/amanda/srv_configs"
}
```

2. Create the file list called by the new dumptype, *include*.

```
amandabackup@quartz:/etc/amanda> cd /var/lib/amanda/DailySet1
amandabackup@quartz:/var/lib/amanda/DailySet1> vi srv_configs
./amandates
./dumpdates
```

3. Append the following lines to the existing **/etc/amanda/DailySet1/disklist.**

```
quartz.zmanda.com    /etc/amanda      amanda-config-backup
quartz.zmanda.com    /var/lib/amanda  amanda-config-backup
quartz.zmanda.com    /etc             include
```

4. Still as the *amandabackup* user, run *amcheck* to verify that a backup will run successfully with the latest changes.

```
quartz:/var/lib/amanda/DailySet1> amcheck DailySet1
Amanda Tape Server Host Check
-----------------------------
Holding disk /data/amanda/hold: 157646136 KB disk space available,
 that's plenty
slot 1: read label `TapeSet1-1', date `X'
NOTE: skipping tape-writable test
Tape TapeSet1-1 label ok
NOTE: host info dir /etc/amanda/DailySet1/curinfo/quartz.zmanda.c
om does not exist
NOTE: it will be created on the next run.
NOTE: index dir /etc/amanda/DailySet1/index/quartz.zmanda.com doe
s not exist
NOTE: it will be created on the next run.
Server check took 6.239 seconds
Amanda Backup Client Hosts Check
--------------------------------
Client check: 1 host checked in 0.318 seconds, 0 problems found
```

20. You can either wait until the next scheduled backup, or you can run *amdump* manually. The DailySet1 backup will run, including the new backup of the Amanda server configurations added to the **disklist**.

a. Executing the backup manually:

```
quartz:/var/lib/amanda/DailySet1> amdump DailySet1
```

b. Generated report for the backup:

```
DUMP SUMMARY:
                                        DUMPER STATS
TAPER STATS
HOSTNAME     DISK        L ORIG-KB  OUT-KB  COMP%  MMM:SS   KB/s
MMM:SS   KB/s
-------------------------- ------------------------------------
-------------
quartz.zmand /etc        0      10      64    --     0:00  242.6
0:05  108.5
```

```
quartz.zmand /etc/amanda 0     150     192     --      0:00 2221.4
0:02  403.2
quartz.zmand -lib/amanda 0     120     160     --      0:00 1822.7
0:02  337.6
```

Verifying and Restoring

Become the *root* user to perform restores.

1. To restore the Amanda server configuration and database files, insert the most recent backup tape into the tape drive.

2. Create a temporary directory in which to extract the files from tape.

```
quartz:~ # cd /tmp; mkdir config
quartz:/tmp # cd /config
quartz:/tmp/config #
```

3. We can now pull the files from tape with *amrestore*. From the new **/tmp/config** directory, extract the files backed up from **/etc** and **/etc/amanda**.

```
quartz:/tmp/config # mt -f /dev/nst0 rewind
quartz:/tmp/config # amrestore /dev/nst0 quartz.zmanda.com /etc
amrestore: 1: restoring quartz.zmanda.com._etc.20060726122359.0
amrestore: 2: skipping quartz.zmanda.com._var_lib_amanda.20060726
122359.0
amrestore: 3: restoring quartz.zmanda.com._etc_amanda.20060726122
359.0
```

4. Extract the files backed up from **/var/lib/amanda**.

```
quartz:/tmp/config # mt -f /dev/nst0 rewind
quartz:/tmp/config # amrestore /dev/nst0 quartz.zmanda.com /var/l
ib/amanda
amrestore: 1: skipping quartz.zmanda.com._etc.20060726122359.0
amrestore: 2: restoring quartz.zmanda.com._var_lib_amanda.2006072
6122359.0
amrestore: 3: reached end of tape: date 20060726122359
```

5. List the extracted tar archives.

```
quartz:/tmp/config # ls
quartz.zmanda.com._etc.20060726122359.0         quartz.zmanda.com.
_var_lib_amanda.20060726122359.0
quartz.zmanda.com._etc_amanda.20060726122359.0
```

6. Now, restore the files to their original locations.

   a. Enter the existing **/etc** directory and restore the amandates and dumpdates files.

```
quartz:/tmp/config # cd /etc
quartz:/etc # tar -vxf /tmp/config/quartz.zmanda.com._etc.2006072
6122359.0
./amandates
```

   b. Enter the existing **/etc/amanda** directory and restore the **/etc/amanda** files.

```
quartz:/etc # cd /etc/amanda
```

```
quartz:/etc/amanda # tar -vxf /tmp/config/quartz.zmanda.com._etc_
amanda.20060726122359.0
./
./DailySet1/
./DailySet1/curinfo/
./DailySet1/curinfo/quartz.zmanda.com/
./DailySet1/curinfo/quartz.zmanda.com/_etc/
./DailySet1/curinfo/quartz.zmanda.com/_etc_amanda/
./DailySet1/curinfo/quartz.zmanda.com/_var_lib_amanda/
./DailySet1/index/
./DailySet1/index/quartz.zmanda.com/
./DailySet1/index/quartz.zmanda.com/_etc/
./DailySet1/index/quartz.zmanda.com/_etc_amanda/
./DailySet1/index/quartz.zmanda.com/_var_lib_amanda/
./DailySet1/oldlog/
.
.
.
./DailySet1/log.20060726115258.0
./DailySet1/log.20060726122019.0
./DailySet1/tapelist
./DailySet1/tapelist.amlabel
./DailySet1/tapelist.yesterday
./DailySet1/curinfo/quartz.zmanda.com/_etc/info
./DailySet1/curinfo/quartz.zmanda.com/_etc_amanda/info
./DailySet1/curinfo/quartz.zmanda.com/_var_lib_amanda/info
./DailySet1/index/quartz.zmanda.com/_etc/20060726115258_0.gz
./DailySet1/index/quartz.zmanda.com/_etc/20060726122019_0.gz
./DailySet1/index/quartz.zmanda.com/_etc/20060726122359_0.gz
./DailySet1/index/quartz.zmanda.com/_etc_amanda/20060726115258_0.
gz.tmp
./DailySet1/index/quartz.zmanda.com/_etc_amanda/20060726122019_0.
gz
./DailySet1/index/quartz.zmanda.com/_etc_amanda/20060726122359_0.
gz.tmp
./DailySet1/index/quartz.zmanda.com/_var_lib_amanda/2006072612201
9_0.gz
./DailySet1/index/quartz.zmanda.com/_var_lib_amanda/2006072612235
9_0.gz
```

c. Enter the existing **/var/lib/amanda** directory and restore the files.

```
quartz:/etc/amanda # cd ~amandabackup
quartz:/var/lib/amanda # tar -vxf /tmp/config/quartz.zmanda.com._
var_lib_amanda.20060726122359.0
./
./.gnupg/
./.ssh/
./gnutar-lists/
./.amandahosts
.
.
.
./gnutar-lists/quartz.zmanda.com_etc_0
./gnutar-lists/quartz.zmanda.com_etc_amanda_0
```

```
./gnutar-lists/quartz.zmanda.com_var_lib_amanda_0
./gnutar-lists/quartz.zmanda.com_var_lib_amanda_0.newfiles
```

7. As the *amandabackup* user, run *amcleanup*. We need to run *amcleanup* due to the inconsistencies created by having Amanda backup its configuration files while a backup is running.

```
amandabackup@quartz:~> amcleanup DailySet1
amcleanup: processing outstanding log file.
```

10. Run *amcheck* on the restored server configuration, again making sure that a successful backup run is possible.

```
amandabackup@quartz:~> amcheck DailySet1
Amanda Tape Server Host Check
-----------------------------
Holding disk /data/amanda/hold: 157646136 KB disk space available,
 that's plenty
slot 3: read label `TapeSet1-3', date `20060726122359'
cannot overwrite active tape TapeSet1-3
slot 4: read label `TapeSet1-4', date `X'
NOTE: skipping tape-writable test
Tape TapeSet1-4 label ok
Server check took 192.672 seconds
Amanda Backup Client Hosts Check
--------------------------------
Client check: 1 host checked in 0.234 seconds, 0 problems found
```

11. Run *amcheckdb* to verify the integrity of the Amanda database.

```
amandabackup@quartz:~> amcheckdb DailySet1
Ready.
```

Best Practices

- Your Amanda server should be a part of your regular backup disk list entries (DLEs). Since Amanda directories are relatively small, a full backup of your Amanda directories on every backup run is strongly recommended.
- Physically label tapes with date and contents. This will make your life much easier in any recovery situation, but will especially help in a crisis situation when untrained personnel may be called on to perform tasks outside of their normal operations.
- Save a copy of the RPM that you use to install Amanda on CD, DVD or a network drive, but not on the Amanda server itself.
- Keep copies of the original operating system and patches available, but in a safe location.
- Don't set up ***rsync*** server on the same machine as your Amanda server.
- Document the following (and keep the documentation available in case of emergency!):

- File system configurations for every server
- List of fully qualified domain names, IP addresses, and hostnames
- Hard drive configuration information
- Media device names
- Configuration information for all of the hardware you have

Conclusion

The key to disaster recovery planning is to always be ready. Taking extra steps as described here to protect the Amanda backup server will be worth your while if (when) disaster strikes. And most importantly, always test your recovery procedures including disaster recovery of your Amanda server.