Embedded NAS RAID Data Recovery Procedure

06 AUG 2010 Aaron Harper

Alpha Dog Technical Services, LLC.

Summary

This procedure is to recover the data on a Linux RAID array. This format is used in small inexpensive consumer / SOHO RAID cabinets which use a linux based microcontroller as the CPU. Such systems will use USB, Firewire, or Ethernet to connect to host systems or networks and make the data stored upon them available. Rather than use a true hardware RAID controller, they use the microcontroller and the Linux operating system to do the work.

The trouble with these drives is that the quality control of the unit (which may cost less than $100.00) is generally less than either the host PC or the drives installed. The controller itself, and often the power supply are the weak links. This is exactly opposite of the typical Linux file server using RAID, where the hard disks are the expected failure point. The point is that while many tools and techniques are available for the reconstruction of a single drive which has failed in a RAID array, the tools are just not there for the reconstruction of an entire array in a foreign machine.

Worse, the average user of this system is not using Linux, but rather Windows or Mac OSX. These operating systems usually can't even read the partition types native to the embedded Linux RAID controller, much less extract the data. While software and services are available to fix this, they are expensive, and in the case of a recovery service, several times the value of the drive. Unless you have mission critical data on the failed controller, it is generally not worth the $2,500.00-10,000.00 a service will charge to recover it.

Fear not. Not all is lost when disaster strikes and your $400.00 Lacie NAS controller goes up in smoke. A spare PC with modest system requirements, a broadband connection to download software, an Ubuntu install CD, and a bit of time is all we need to save the day.

Required Hardware

Other than the drives from the NAS RAID, the only hardware you will need is a PC with enough free SATA ports to install all of the drives from the RAID array plus the drive and CD for the operating system.

Tip: Don't connect the SATA drives from the RAID array in the system until after Ubuntu is installed. It would be easy to make a mistake and install Ubuntu on the wrong hard disk which would overwrite any data on the drive. You would then permanently loose the ability to recover any data.

Required Software

Install Ubuntu 10.04LTS, and allow to update. While the distribution is not that critical, this procedure was written for Ubuntu with the Linux Newbie in mind. It must be a complete and stable install. While a LiveCD may work, this has not been tested. There are plenty of on line how-to guides on installing Linux, though the process is quick and painless.

Open the Applications menu at the top menu bar and select Ubuntu Software Center at the bottom. From within this application we can install thousands of applications to suit Ubuntu to our tasks.

```
Application > Ubuntu Software Center
```

In the search area type the name of the application we are looking for, in this case mdadm. This package is a comprehensive command line RAID toolkit. Use it carefully and read the manual. Like most tools it can destroy easier than it can repair. Click on the Install button. You will need to enter you password to proceed.

```
Search {mdadm} > Install
```

Next, search for and install gparted using the Ubuntu software center. This package is a graphical partition manager. It is overkill for our purposes, but it is the easiest way to visualize the partition layout in an graphical and intuitive way.

```
Search {gparted} > Install
```

Install xxdiff using the Ubuntu software center. This is a old, bare metal file and directory compare routine. The invocation is command line, but the output is legacy X, it is a graphical application, just barely.

```
Search {xxdiff} > Install
```

Preparing the Tools

At this point, it is assumed that you have mounted the drives from the enclosure in a PC with Ubuntu 10.04LTS installed as indicated in the previous section. Open the System menu at the top menu bar. Move your mouse to Administration, and select Gparted.

```
System > Administration > Gparted
```

You will need to enter an admin password to run Gparted. Once Gparted has opened, click on Gparted in the top menu and move your mouse to Devices. Here find the device name of the drive(s) you wish to recover. You are looking for the largest partition on each drive. Make a note of the device IDs (ie. /dev/sdb2) for each drive, or simply leave Gparted open with your information available, you will need it shortly.

```
GParted > Devices > ...
```

Now open a command prompt. To do this, go to the Applications menu at the top menu bar. Move your mouse to Accessories, then to Terminal, near the bottom.

```
Accessories > Terminal
```

The next series of commands will need to be run as root. To do so in Ubuntu, type the following command in the terminal window.

```
> sudo su
```

Enter an admin password when prompted. The command prompt will have changed from a $ to a #, indicating root access.

Re-creating the RAID array

The next step is to to test access to the RAID drives. For this part you will need the device IDs (ie. /dev/sdb2) from Gparted in the step earlier. Replace "/dev/sdb2" with whatever Gparted indicated the correct partitions are.

```
>mdadm --examine /dev/sdb2
>mdadm --examine /dev/sdc2
...
```

Repeat this process until all drives have been checked. The next step constructs the RAID drive from the components. Again, replace "/dev/sdb2" with whatever Gparted indicated the correct partition is.

```
>mdadm --assemble --run /dev/md0 /dev/sdb2
>mdadm --assemble --run /dev/md1 /dev/sdc2
...
```

This is all that is required in a RAID 1 (mirrored) or RAID 0 (striped) array. You must add all the original drives to recover the data. It is possible to do so with one less than the total number of original drives in and all but RAID 0 (striped), but the process is more complex and outside the scope of this paper. To add additional drives, add drives to the constructed array. The number ascribed to the multiple drive device (/dev/md0) is sequential, and replace "/dev/sdc2" with whatever Gparted indicated the next correct partition is. Continue this process for each drive in the RAID array.

Mounting the RAID Array

When complete, create an empty directory for Ubuntu to mount the RAID array to. This directory may be anywhere, but I recommend placing the directory in either the active user's home directory or preferably in the /mnt directory which is where mount points are placed by convention.

```
>mkdir /mnt/Lacie
```

Now we simply make the connection between the RAID array and the mount point. This is called mounting the drive. To do so we issue the mount command followed by the first multiple drive device, usually /dev/md0, and finally the mount point we created in the previous step.

```
>mount /dev/md0 /mnt/Lacie
```

The RAID array will now appear on the desktop as an icon and all the files will be visible by double clicking on the icon.

Handling and Moving the Data

Since we are in root in Linux, anything we create will have root as the owner. This means that the files may have no permissions for anyone else to do anything. Further, since Windows and Unix files use permissions differently, this could result in unpredictable behavior. To counteract this, we simply change the permissions to where anyone can do anything with the files. Replace "/mnt/Lacie" with whatever mount point was used in the previous two steps.

```
>chmod -R 777 /mnt/Lacie/*
```

Next, plug in portable drive or connect to network resource which will be used to get the data off the drive. Once this drive shows up on the desktop as an icon, open both it and the RAID array. Move the contents of RAID array to portable drive or network resource with drag and drop.

This part may take quite a while depending upon the size and speed of the drives, but you will need to check on it occasionally. If there is an error, the transfer will halt temporarily and a dialog box will pop up asking if you wish to abort, cancel, or skip. Make a note of the file that did not transfer and the reason, then press skip.

Not every transfer error will be file damage, in fact those will be quite rare in a RAID array. One reason an undamaged file may not transfer is size. The largest single file size for a FAT32 partition is 4GB. If your portable drive uses FAT32, you may not be able to transfer huge files like a DVD image or system backup. If this is the case, you may wish to store these large files on a different media.

Checking the Data

Once this is complete, you may wish to check the transferred directory for completeness or errors. This process is optional, but highly recommended. In typing this command replace "/media/Elements/RAID" with the directory where the data was copied to on the portable drive or network resource, and "/mnt/Lacie " with the RAID array mount point we created a few steps back.

```
>xxdiff /media/Elements/RAID /mnt/Lacie
```

This will take a while, during which the window will "grey out" as though the process were locked up. This is not the case, but rather xxdiff is hard at work checking the two directories for differences.

When it is complete, compare difference for transfer failures and fix or ignore them. You may move forward and back in the list by hitting the "n" key for the next difference, or the "p" key for the previous one. The rest of the menus are pretty straightforward. Just don't loose track of which window is which. They are labeled at the top of the column frame to help keep it all straight.

Closing the Process

If the icons are present on the desktop, right click on the portable drive or network resource and unmount it. Then to unmount the RAID array, issue the following command, replacing "/mnt/Lacie " with the RAID array mount point we created a few steps back.

```
>umount /mnt/Lacie
```

Finally, disassemble the RAID array by stopping its component drives. Begin at the last created, usually the highest number, and work your way to the first created, usually /dev/md0.

```
> ...

>mdadm --stop /dev/md1

>mdadm --stop /dev/md0
```

Now we need to exit the root session. Simply type "exit" in the terminal window.

```
>exit
```

You will notice that the prompt in the terminal window will have changed from a # to a $, indicating normal user access rather than root access. To close out of the terminal session, type "exit" in the terminal window again.

```
>exit
```

At this point, we can shut down the system. Either click on the power icon in the far right of the top bar, or tap the PC's power button. In either case, select Shut Down, safely turning off the PC.

```
Power > Shut Down >Shut Down
```

Once the PC shuts down, you may remove the RAID hard disks for use in other arrays or devices. Your data is stored on the portable drive or connect to network resource.