Moving the NMIS database to a new partition

In it's default configuration NMIS likes to be at /usr/local/nmis8, with all of it's data held within this directory but NMIS is not required to be in this location. All, or just parts, can be moved to different directories which can be on different partitions or Logical Volumes.

Example:

Goal: To move the NMIS database off of the same partition as /usr/local and at the same time make a backup of the NMIS runtime. To accomplish this goal we first need a location where we will move/copy the files to, for this example I will use a target of /opt/nmis8. Once we have this we will copy all of NMIS8 to the new location, then edit the config files in the current location to use the new data directories.

1. Copy or move files to new directory (on a production system, you should stop NMIS from polling by editing the crontab and appending a "#" to the lines for NMIS polling). **Pick one of these options, C is recommended.**

    a. Copy files to a new directory

    ```
    cp -rp /usr/local/nmis8/database /opt/nmis8
    ```

    b. Move files to a new directory

    ```
    mv /usr/local/nmis8/database /opt/nmis8
    ```

    c. **If you have a large database** and copying or moving files takes a significant amount of time then we will prefer to use rsync. In this case **do not** stop NMIS from polling at this time, it will be done in a later step. We will do a "hot" copy now and follow it with a "cold" copy.

    ```
    #do a hot copy where nmis is still updating
    rsync -av --inplace /usr/local/nmis8 /opt/
    ```

2. Check permissions on the copied files. The above commands should preserve owners and permissions, but double check anyway

    ```
    ls -l /opt # nmis8 should be owned by nmis, group nmis
    ls -l /opt/nmis8/ # all should be owned by nmis, group nmis

    #if the have incorrect permissions run these commands:
    chown -Rv nmis:nmis /opt/nmis8
    chmod -Rv g+w /opt/nmis8
    ```

3. **If you have a large database,** now is the time to stop NMIS from polling. As mentioned previously, the safest way to do this is by commenting out the lines that run NMIS polling, be sure to do all of them, just to be safe. Another option (and not necessarily recommended) is to stop the cron daemon with: `service crond stop #Do Not forget to restart it later`. Now we will run the rsync copy one more time to get any updates:

    a. Stop polling. Wait for nmis to stop updating.

    ```
    crontab -e #comment out all nmis.pl lines
    # continue running the following command until you no longer see nmis running
    ps aux | grep nmis.pl
    # make sure you don't see a line like this:
    # /usr/bin/perl ./nmis.pl
    ```

    b. Now that NMIS is no longer polling we can run a "cold" copy to get the last updates that NMIS made while we were "hot" copying.

    ```
    sudo rsync -av --inplace /usr/local/nmis8 /opt/
    ```

4. Edit config files so that NMIS will write data to our new location, edit /usr/local/nmis8/conf/Config.nmis

    ```
    # this line
    line 13: '<nmis_data>' => '<nmis_base>',
    # should change to
    line 13: '<nmis_data>' => '/opt/nmis8',
    ```

5. To test, move the existing data directores to new folders and symbolically link the new database location to the old one.

```
mv /usr/local/nmis8/database /usr/local/nmis8/database_old
ln -s /opt/nmis8/database /usr/local/nmis8/database
```

   In a new installation, the symbolic link will not be necessary, but when changing the database location in a running NMIS8 installation, putting the SYM link in will help during the update.

6. Run a manual update, you should see no errors.

```
/usr/local/nmis8/bin/nmis.pl type=update debug=1
# you should not see any errors like this:
# createRRD, ERROR (localhost) unable to create /opt/nmis8/database/health/server/localhost-reach.rrd:
creating '/opt/nmis8/database/health/server/localhost-reach.rrd': No such file or directory
```

7. Open or reload the NMIS web page and everything should still look the same.
8. After confirming there are no errors you can now safely delete the old database directory.

```
rm -rf /usr/local/nmis8/database_old
```

9. Now re-enable your cron jobs. If you commented lines out, remove the comments. If you stopped crond, start it again: `service crond start`

Conclusion

In addition to moving the database folder, it may also be a good idea to move the logs folder if you are running short of space on the partition holding /usr /local.

We now have NMIS saving all databases to a different folder (which could be on a different partition).

If you have copied data, you will need to delete the files you no longer need to free up disk space.

This article describes a high level process for relocating data, the final process used by an organisation should be tested before running on a production system.