How-to for installing OpenELISGlobal Database backups

# Background

Good practices for database backups include backing up often and to a location as distant from the host machine. To support these practices the OpenELIS installer creates the structure for doing backups and the cron job for the doing a daily backup. The installed backup is not suitable for all installations and this document provides guidance for having a suitable solution.

# Assumptions

1. The person setting up the backup is familiar with basic Linux commands and knows how to find information to fill in their gaps of knowledge.
2. An ftp site has been identified to receive the backup files.
3. The network and ftp site have sufficient security to transmit and host files with identifiable patient information.

# Basic backup processes

1. A cron job will run the backup script, “DatabaseBackup.pl”
2. The script will do the following:
   1. Do a backup of the database and compress it.
   2. Copy that backup to a sub-directory named “daily”. That directory will always have the latest copy of the backup
   3. Copy the backup to a sub-directory named “cumulative”. That directory will have copies of the last 30 days of the backups
   4. Copy the backup to a sub-directory named “transmissionQueue”. That directory will have copies of backups waiting to be sent off-site.
   5. Send the files in transmissionQueue to an ftp site. As soon as they are successfully sent they will be removed from tansmissionQueue.

Customizing the script

The installed script can be found in the sub-directory, named openElisBackup, of the home directory of the logged in user during installation. The installed script can be replaced with a new script of the same name. The person configuring the backup should have been given a copy of a new DatabaseBackup.pl file, which is the one to be modified.

First, copy over the existing DatabaseBackup.pl to your computer, this has a few important variables such as the database name, the postgres password, siteID, etc. Pay attention to lines: 71, 73, 85 as they have info which is possibly different for each installation. More details on variables which needs to be changed are below with a screenshot.

Use this info to customize your backup script which you will then copy over to the server. Use whatever text editor you prefer for the customization. Save your changes.

After this we need to change the name of the file on the server. Connect to the server (LINUX) and run:

*sudo mv DatabaseBackup.pl DatabaseBackup.old*

The directory openElisBackup needs to have permissions changed to copy the file over using the “itech” user. Run:

*sudo chmod 777 openElisBackup*

Then connect via scp and copy over the changed DatabaseBackup.pl to



Steps:

Line 71: The password for the database being backed up. Replace “clinlims”. If it is not known it can be found in /usr/share/tomcat6/conf/Catalina/localhost in the xml file.

Line 73: The name of the site. Replace “IPCI”. This will be used to create the backup file name and will be the prefix to the file.

Line 74: The destination for the file. Replace “192.168.1.1/EFI/backup/”. The IP address is the IP address of the ftp server and the rest is the path on the server which will receive the files. The person doing this installation should make sure that path exists. Make sure that the trailing ‘/’ is part of the path.

Line 75: The user name of the ftp account. Replace “backup”

Line 76: The password of the ftp account. Replace “backupoe”

Line 81: The database backup command. This should not have to be changed but will be reviewed here. There are three places where clinlims is used, which can be confusing.

1. –U clinlims This is the user name for the account
2. –n \”clinlims\” This is the schema name.
3. Trailing clinlims. This is the name of the database.

Line 84: The base name of the backup file. Replace “CI\_IPCIOpenElis”. This will be prefixed by the site name and will have the current date and time added to it to make up the name of the backup file.

Line 85: If a device is mounted to the machine then another backup can be done to that device. If the device has been removed then it will fail quietly. Not that if the device is not available and is then made available then the missed backups will not be done.

# Testing

The script is a perl script and by tested by running

sudo perl ./DatabaseBackup.pl

The end result will be a message from the curl application indicating that it has sent the file to the ftp server and seeing a new file on the ftp server