

One of the most important things about having a website is making sure you have a recent backup of your entire website in case of any misfortunes, whether there be a server problem/issue, malicious intent from a hacker or a move to a new host.

Remembering to backup your webpages is very important, but also making a recent backup of your bulletin board/forums database is just as important.

This tutorial is designed to guide you through the process of backing up a MySQL database using a shell script, telnet and crontabs.

If you are not familiar with either Telnet or Crontabs, here's are some tutorials for you:

- [SuperScripts Telnet Tutorial](#)
- [SuperScripts Crontab Tutorial](#)

### **First Things First – The Script**

Visit [this thread](#) over at the [vBulletin.com](#) forums.

Highlight all the code in the first part of the post and save it using a text editor with **no extension** (eg mysqlbackup). The reason the file is to be saved without an extension is because it's a shell script to be run on the server using either Telnet or Crontabs and not a browser.

This script was written by Chris Schreiber from [vBulletin.Com](#) (a bulletin board which uses PHP and MySQL)

#### **Shell Script Information:**

##### **Line 7**

```
databases="database_name"
```

This is the name of your database. If you are unsure of this you should contact your host.

##### **Line 10**

```
backupdir=~/.backup
```

This line tells the shell script where to save the backup file. Using the tilda sign (~) before the "/backup" folder tells the shell script to create a folder (if not already created) called backup on your hosting spaces root folder. Alternatively, you can specify where you want to have this folder created by entering the full path to the folder itself, eg:  
/usr/home/htdocs/backup

##### **Line 13**

```
mysqldumpcmd=/usr/local/mysql/bin/mysqldump
```

This line tells the shell script where to find the mysqldump command. If you are unsure or do not know where this command is located on your host, don't panic. Simply login to your host using Telnet and type :

```
which mysqldump
```

This should command should come back with a result something like :

/usr/local/mysql/bin/mysqldump

This tells you where this particular command is located on *your* host.

#### **Line 16**

```
userpassword="--user=root --password=password"
```

This line tells the shell script what mysql login info to use.

This info is the same info you normally use to login mysql using Telnet. If you are unsure of this info please contact your host.

#### **Line 19**

```
dumppoptions="--quick --add-drop-table --add-locks --extended-insert --lock-tables"
```

This line tells the shell script what kind of backup to make (no changes need to be made to this line). The details of this line are not that important to know. If you would like to know more about them, you can read about them [here](#).

#### **Lines 22–24**

```
gzip=/bin/gzip  
uuencode=/usr/bin/uuencode  
mail=/bin/mail
```

These 3 lines tell the shell script where to find the necessary utilities to zip, uuencode and mail the backup files created. Again these locations can be found by using Telnet and typing :

```
which gzip  
which uuencode  
which mail
```

#### **Lines 28–30**

```
sendbackup="y"  
subject="Your MySQL Backup"  
mailto="you@yourdomain.com"
```

These 3 lines of the shell script tell the script whether you would like an email sent to the address stated with the backup as an attachment.

The rest of the shell script does not require any changes.

From here we have 2 choices as to how to run the backup.

- 1. Manually
- 2. Using crontab.

### **Manually running the backup & using crontab**

#### ***Manually running the backup:***

You can manually run the backup by logging into your host with Telnet and typing the following at the command prompt:

```
./script_name (eg mysqlbackup)
```

#### **Using Crontab:**

You can use Crontab to run the shell script automatically as certain intervals (ie Daily, Weekly, Monthly).

To setup crontab, log into your host using telnet and at the command prompt type:

Pico cronfile

When the file is opening for editing you can use the tutorial about crontabs mentioned earlier in this tutorial to tell the cronfile when to run the shell script (eg \* \* \* \* mon ~/mysqlbackup)

Once done save the cronfile. This will bring you back to the telnet command prompt. Now you need to set the crontab. To do this you simply type:

```
crontab cronfile
```

at the command prompt.

And that's it.

If you have setup your crontab and have turned on the "email the backup" option in the shell script itself, you should receive an email (at the set interval) with the backup as an attachment.