

Backing up MySQL Databases

Posted November 27th, 2012 by Simon Davies (/) & filed under Bash (<https://simon-davies.name/categories/bash>), MySQL (<https://simon-davies.name/categories/mysql>).

As a web developer you learn that there is nothing more important than keeping regular backups of your databases. I've created a little bash script to help with Backing up MySQL databases for this I'll use mysqldump. Mysqldump is a nifty little utility that lets you dump the contents of a database to a file, so I use this to create a little bash script that will backup the database at regular intervals, and I'll even chuck in deleting backups older than 30 days so you don't over run with files. First lets state the obvious security warning here, this script will contain details of your database so make sure you set the correct permissions, I use 700 but this may change depending on your hosting provider. Also don't place the backup folder in your public_html folder otherwise people will be able to see the contents of your database and again lets make sure the .sql have the correct permissions otherwise over users on your server will be able to access them. I create a folder in my home folder called _backup and inside here I will have a directory for my backups mysql and a bash script called backup.sh. First I'll create a few variables to contain the database credentials.

```
#!/bin/bash

# Database credentials
user=""
password=""
host=""
db_name=""
```

The line at the top is the shebang and is used to select the interpreter, in my case this is bash. Next I'll declare a few other variables, 1 for which directory I want to use for the backup files. The other is for the current date as I also like to include the date in the file name.

```
# Other options
backup_path="/path/to/your/home/_backup/mysql"
date=$(date +"%d-%b-%Y")
```

Now we need to create the actual backup files, before doing so I run umask to set the default permissions for the backup file. This will give me 600 which will allow only my user to read and write to the file. Then we run the mysqldump utility with are variables from above. I create the file name using the database name and current date, so it's easy to see from the file name which backup is which.

```
# Set default file permissions
umask 177

# Dump database into SQL file
mysqldump --user=$user --password=$password --host=$host $db_name > $backup_path/$db_name-$date.sql
```

The last step is to delete files older than 30 days, for this we will use the find utility. First we pass the path to the directory that contains the .sql files and use a wildcard to select all. Then we need to set 2 options, the first is mtime which we use to select files that are older than a given amount of days. The next option exec allows us pass the files that are found into another utility, which I will use rm to delete the files. The last few characters are to end the line.

```
# Delete files older than 30 days
find $backup_path/* -mtime +30 -exec rm {} \;
```

And that's it, all you need to do now is setup a cron job to run whenever you want to backup your database. I run this everyday at midnight but you can run it hourly, weekly or however else you want. The whole backup.sh is below.

```
#!/bin/bash

# Database credentials
user=""
password=""
host=""
db_name=""

# Other options
backup_path="/path/to/your/home/_backup/mysql"
date=$(date +"%d-%b-%Y")

# Set default file permissions
umask 177

# Dump database into SQL file
mysqldump --user=$user --password=$password --host=$host $db_name > $backup_path/$db_name-$date.sql

# Delete files older than 30 days
find $backup_path/* -mtime +30 -exec rm {} \;
```

This is creating a local backup of your database files, and to be extra safe you should be backing up these files on another server in a different location as well. If you want help setting this up let me know if comments below.

ALSO ON SIMON DAVIES

Setting a different ul bullet colour in CSS

9 years ago • 1 comment

```
ul { color: red; } li { color:
black; } ul { list-style-type:
none; position: relative; ...
```

WP-Admin broken after updating WordPress

9 years ago • 2 comments

```
/Applications/MAMP/conf/...
zlib.output_compression =
Off ...
```

Syncing Dropbox

10 years ago

```
cd ~/Dropt
/Applicatio
MAMP_DE
```

38 Comments

Simon Davies



Login ▾

Favorite

Tweet

Share

Sort by Best ▾



Join the discussion...

LOG IN WITH

OR SIGN UP WITH DISQUS

Name

**Brunobliss** • 7 years ago

simple and awesome! I've added an array of databases and looped through each to preform a backup

```
db=(database1 database2)
```

```
for i in "${db[@]}"
```

```
do
```

```
mysqldump --user=$user --password=$password --host=$host $i >
$backup_path/$i_name-$date.sql
```

```
done
```

3 ^ | ▾ • Reply • Share ›

**Paul DiCecco** • 8 years ago

The rm command is not working for me. I changed it to +5 and noticed it not working so I then tried +05 and still not deleting anything.

```
# Delete files older than 30 days
```

```
find $backup_path/* -mtime +05 -exec rm {} \;
```

2 ^ | ▾ • Reply • Share ›

**Paul DiCecco** ➔ Paul DiCecco • 8 years ago

The response from my hosting support:

Unfortunately, the issue you are having would be considered coding-related and it is outside the scope of support. However, you

should be able to use the -delete flag for find instead of

should be able to use the `delete` flag for `find` instead of having to `-exec`, and I noticed that you are using a asterisk without escaping. You may need to contact someone familiar with bash scripting to assist you if you are still having issues with your script.

^ | v • Reply • Share ›



simonda86 Mod → Paul DiCecco • 8 years ago

Hi Paul,

Changing the number of days to "+5" is correct, so there must be something else wrong here. Try running `find ./ -mtime +5` from the command line in the backup directory. This should show you a list of all files older than 5 days.

^ | v • Reply • Share ›



Paul DiCecco → simonda86 • 8 years ago

Yes, that worked correctly but the command

```
find $backup_path/*.sql -mtime +5 -exec rm -f {} \;
```

in my script doesn't delete the one file that is over five days old.

^ | v • Reply • Share ›



Paul DiCecco → Paul DiCecco • 8 years ago

ok it looks like it may be working but setting it to +5 actually keeps 6 days..

^ | v • Reply • Share ›



seanrall • 8 years ago

How do you run this in a Cron Job? Im a noob so please be nice.

2 ^ | v • Reply • Share ›



simonda86 Mod → seanrall • 8 years ago

Hi Sean,

Enter `"crontab -e"` in your command line, to edit your crontab. Then enter a new line in this file for the job. If you want to run this every day at midnight for example you would enter add the following.

```
00 00 * * * /path/to/your/file
```

I would put this script in the bin folder of my home directory so my path would be `~/bin/backup.sh`

I hope this helps.

Simon

1 ^ | v • Reply • Share ›

**seanrall** → simonda86 • 8 years ago

That is what I needed. Thanks!

^ | v • Reply • Share ›

**Rajat Sethi** • 9 years ago

how to backup each database in a separate file in a folder named as the present date?

2 ^ | v • Reply • Share ›

**simonda86** Mod → Rajat Sethi • 9 years ago

Hi Rajat,

You can do something like this, this should backup all databases that the user has access to.

<https://gist.github.com/simonda86/1111111>

Simon

^ | v • Reply • Share ›

**Vamsi k** → simonda86 • 4 years ago

Hi Simon i want to delete dbbackups between date-month-year Hours-minutes-seconds of specified dates how can I do that

^ | v • Reply • Share ›

**Travis Hershberger** • 8 years ago

Great little script. Even us old-hats need a refresher, we only set this stuff up once each time we setup a database/database server.

I added 4 lines in mine to compress the backup file.

```
#Compress the backup file
7z a $backup_path/$db_name-$date.7z $backup_path/*.sql
#Remove the original sql file
rm -I $backup_path/*.sql
```

Handy if you expect the database to be quite large, or just save a little drive space. I used 7z, but you could use any compression program you like (zip, gzip, etc)

2 ^ | v 1 • Reply • Share ›

**Ben Yanke** → Travis Hershberger • 6 years ago

Old comment, I know....but You should probably include some error checking. Otherwise if the compression fails, you have no backup.

Something like (pseudocode):

```
if backup successful: rm
else: log an error somewhere (and notably, do not remove
the sql file)
```

^ | v • Reply • Share ›

**Peter Furesz** • 8 years ago



Just be aware to set root / directory as backup path, because the last find command will delete every file from your entire hard drive :)

I suggest to use like this:

```
find $backup_path/* -name *.sql -mtime +30 -exec rm {} \;
```

2 ^ | v 1 • Reply • Share ›



John → Peter Furesz • 8 years ago

I wish I saw this comment 5 hours ago. ugh!

1 ^ | v • Reply • Share ›



Haider Ali • 5 years ago

Variables should be in single quotes, other wise it can make escape issue like dollar sign in password will produce access denied.

Script appreciated.

1 ^ | v • Reply • Share ›



nipon • 5 years ago

How to backup to a server with ssh access securely ?

1 ^ | v • Reply • Share ›



simonda86 Mod → nipon • 5 years ago

Hi Nipon, I would recommend running the backup script on the database server and then use scp to copy the SQL files over to a remote server.

1 ^ | v • Reply • Share ›



nipon → simonda86 • 5 years ago

do you have example script ?

^ | v • Reply • Share ›



simonda86 Mod → nipon • 5 years ago

Sorry I don't have an example but it should be relatively simple to write.

^ | v • Reply • Share ›



Vural Acar → nipon • 5 years ago • edited

@nipon you can simply use (linux only)

```
>ssh {address-to-server} -l {username}
```

```
>@Password: {enter your password}
```

```
>crontab -e
```

vim editor will be opened and press "i" to be able to insert.

enter following line to run the [backup.sh](#) every 12 hours.

(you can change, how you like)

```
0 */12 * * * /path/to/your/site/backup.sh >/dev/null 2>&1
```

this crontab line will run every 12hours..

if you don't know how to check the current path of your directory

```
> pwd
```

good luck :)

^ | v • Reply • Share ›



Luiz Barros • 6 years ago

Not work for me

octal number out of range 11 :umask 117

/home/admin/[backup.sh](#) line 13:

/home/admin/web/default.domain/public_html/bk_my

No such file or directory

find '/home/admin/web/default.domain/public_html/bk_mysql\r*' :

No such file or directory

Help - ME

1 ^ | v • Reply • Share ›



Guest • 8 years ago

Hi,

So I am completely new at this, so not even sure if this applies to what I am trying to do...

But can I use this to run a cron job in cPanel? Currently I am backing up my databases using the following command:

```
nice -n 19 mysqldump -u myusername --password=mypassword
```

```
mydatabase -c | nice -n 19 gzip -9 >
```

```
~/backups/mydatabasename/myfilename-$(date
```

```
'+\\%Y\\%m\\%d').sql.gz
```

can I add the find bit on to this to delete older backup files?

sorry if this is a stupid question.

1 ^ | v • Reply • Share ›



simonda86 Mod ➔ Guest • 8 years ago

Hi Hanna,

No question is a stupid one. :-)

Yes you can use the find command to delete the older backup files. Something along the lines of "find
~/backups/mydatabasename/* -mtime +30 -exec rm {} \;"
would delete files that are older than 30 days.

I hope that helps!

^ | v • Reply • Share ›



Mau España • 8 months ago

help please, I already did that, but only do this: -- Server version
5.5.5-10.4.17-MariaDB-cll-lve

```
/*!40101 SET
```

```
@OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLI
```

```
*/;
```

```
/*!40101 SET
```

```
@OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET R
```

```

@OLD_CHARACTER_SET_NAME=@@CHARACTER_SET_NAME
*/;
/*!40101 SET
@OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION
*/;
/*!50503 SET NAMES utf8mb4 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET
@OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS,
UNIQUE_CHECKS=0 */;
/*!40014 SET
@OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS
FOREIGN_KEY_CHECKS=0 */;
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='NO_AUTO_VALUE_ON_ZERO' */;
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES,
SQL_NOTES=0 */;

```

I don't know what to do? heeeeeelp

^ | v • Reply • Share ›



gsc 89 • 5 years ago

Its better to use password="

The single quotes wont cause issues when using passwords which contain characters that would need escaping I spent literally 2 hours trying to figure this out.

^ | v • Reply • Share ›



Nick Vlădiceanu • 5 years ago

awesome, thanks !!

^ | v • Reply • Share ›



huzanspenta • 5 years ago

Hi,

Thanks but it shows permission denied. Please support.

^ | v • Reply • Share ›



simonda86 Mod ➔ huzanspenta • 5 years ago

Hi,

This is a MySQL issue, please check your connection

CONTACT

Want to hear more about Simon Davies Digital and what we can do for your business or simply want to say hello, then get in touch.

+44 7846 400933

hello@simon-davies.name (<mailto:hello@simon-davies.name>).

in

[. \(\[!\\[\\]\\(e3f8612927870f2e0f9f5989e6dd3064_img.jpg\\)\]\(https://uk.linkedin.com/in/simon-</p></div><div data-bbox=\)](https://uk.linkedin.com/in/simon-)

[davies-](https://twitter.com/simondavies)



[. \(<https://github.com/simondavies>](https://github.com/simondavies)

© 2022 Full-Stack Web Developer | Built with [Craft CMS](https://craftcms.com/) (<https://craftcms.com/>).

Simon Davies Digital Ltd. is a company registered in England and Wales with company number 09952124

[Terms & Conditions](/terms-conditions) (</terms-conditions>) [Privacy Policy](/privacy-policy) (</privacy-policy>).