

[x Dismiss](#)

## Join the Stack Overflow Community

Stack Overflow is a community of 7.4 million programmers, just like you, helping each other.

Join them; it only takes a minute:

[Sign up](#)

## Variables as commands in bash scripts



I am writing a very simple bash script that tars a given directory, encrypts the output of that, and then splits the resultant file into multiple smaller files since the backup media don't support huge files.

I don't have a lot of experience with bash scripting. I believe I'm having issues with quoting my variables properly to allow spaces in the parameters. The script follows:

```
#!/bin/bash

# This script tars the given directory, encrypts it, and transfers
# it to the given directory (likely a USB key).

if [ $# -ne 2 ]
then
    echo "Usage: `basename $0` DIRECTORY BACKUP_DIRECTORY"
    exit 1
fi

DIRECTORY=$1
BACKUP_DIRECTORY=$2
BACKUP_FILE="$BACKUP_DIRECTORY/`date +%Y-%m-%dT%H-%M-%S`.backup`"

TAR_CMD="tar cv $DIRECTORY"
SPLIT_CMD="split -b 1024m - \"$BACKUP_FILE\""

ENCRYPT_CMD='openssl des3 -salt'

echo "$TAR_CMD | $ENCRYPT_CMD | $SPLIT_CMD"

$TAR_CMD | $ENCRYPT_CMD | $SPLIT_CMD

say "Done backing up"
```

Running this command fails with:

```
split: "foo/2009-04-27T14-32-04.backup"aa: No such file or directory
```

I can fix it by removing the quotes around `$BACKUP_FILE` where I set `$SPLIT_CMD`. But, if I have a space in the name of my backup directory it doesn't work. Also, if I copy and paste the output from the "echo" command directly into the terminal it works fine. Clearly there's something I don't understand about how Bash is escaping things.

[bash](#) [unix](#) [shell](#)

edited Mar 12 '16 at 12:48



Alan W. Smith

13.8k 3 43 70

asked Apr 27 '09 at 18:36



wxs

2,647 2 21 39

Why would you embed `$BACKUP_FILE` in `SPLIT_CMD` when you could just put it after `$SPLIT_CMD` in the pipeline? – [Nietzsche-jou](#) Apr 27 '09 at 18:50

Well I could do that, but at that point there's not really much point in having variables to contain my commands and I may as well just expand it all out as in [Juliano's](#) answer below – [wxs](#) Apr 27 '09 at