

mv &amp; mmv

# Cool Moves

mv (“move”) is a standard command that has been around on UNIX systems right from the outset. It allows you to move or rename files and directories. Unfortunately, things become slightly more complex when you need to move several things around in a single step. This is the point where mmv (“multiple move”) comes into play. **BY HEIKE JURZIK**



The *mv* (“move”) command, moves and renames files and directories. Unfortunately, if you need to address multiple files and, for example, add a new file extension to them, you are not going to get far with Bash **wildcards**. *for* loops and *basename* allow you to perform some pretty neat tricks, but *mmv* can save you a lot of headaches.

## Easy Moves

*mv* allows you to rename files. The simple syntax in the current working directory is as follows:

```
h@hej~ mv file.html file1.html
```

This also works across directories if you supply the absolute or relative pathname [1]. As the following command:

```
h@hej~ mv /tmp/1.pdf file.pdf
```

moves the file *1.pdf* from the */tmp* directory to your current directory and renames it to *file.pdf* in the process. You can also do this with whole directories: *mv /tmp/mp3s/ /home/huhn/* moves the *mp3s* directory to your home directory. If you require a more secure approach, you can set the *-i* flag (or use the longer *--interactive* parameter). This will tell the program to prompt you to confirm if a file of the same name already exists in the target directory:

```
h@hej~ mv -i /tmp/1.pdf file.pdf
mv: overwrite `file.pdf'?
```

If the directory contains multiple files with the *.htm* extension, and if you want to rename all of them to *.html*, you might find the going slightly tougher with *mv*:

```
h@hej~ mv *.htm *.html
mv: when moving multiple files ↵
last argument must be a ↵
directory. Try `mv --help' for ↵
more information.
```

Bash parses the wildcards and attempts to move the individual files (*file1.htm*, *file2.htm*, ...) to a directory since *mv* expects a directory as the last argument if you are moving multiple files. This

## Command Line

Although GUIs such as KDE or GNOME are useful for various tasks, if you intend to get the most out of your Linux machine, you will need to revert to the good old command line from time to time. Apart from that, you will probably be confronted with various scenarios where some working knowledge will be extremely useful in finding your way through the command line jungle.

command would only work if you had a directory that matched \*.html; but it would move the .htm files to that directory, and that is not exactly what you had in mind. A for loop combined with basename should do the trick:

```
h@hej~ for i in *.htm; do mv ?
$i `basename $i .htm`.html; done
```

Each filename with the .htm extension is assigned to the variable i, and mv is then called for the variable with two parameters. The first parameter is the source file (\$i), the second the output file, which is created using only the "basename" by removing the original extension and adding the new extension .html.

### Elegant Mover!

The mmv tool is not installed by default on most distributions, so you will have to install it before you continue. If you cannot locate the tool on your installations CDs, try a search engine like Rpmseek [2] or Rpmfind.net [3]. The sources are available from [http://ftp.debian.org/debian/pool/main/m/mmv/mmv\\_1.01b.orig.tar.gz](http://ftp.debian.org/debian/pool/main/m/mmv/mmv_1.01b.orig.tar.gz). Debian users can

use `apt-get install mmv` to install the package.

mmv allows you to re-write the for loop from the previous example:

```
h@hej~ mmv "*.htm" "#1.html"
```

The asterisk in the source file refers to any files with the .htm extension in the current directory. #1 for the target file refers to the first wildcard in the source. It makes sense to put both parameters in quotes to stop the shell interpreting the \* and # metacharacters. Let's assume you have three files called hid.txt, had.txt and hod.txt, and want to change the extension to .html for all three of them. mmv is quite happy to do the following:

```
h@hej~ mmv "h?d.txt" "h#1d.html"
```

Uppercase letters in filenames can be an issue, particularly when exchanging files between Windows and Linux machines. mmv provides an elegant solution: just type an l after the hash sign # (for lowercase) or a u (for uppercase):

```
h@hej~ ls
```

```
ENGELBERT.HTM Huhn.HTM
h@hej~ mmv "*.HTM" "#1.html"
h@hej~ ls
engelbert.htm huhn.htm
```

mmv will even accept multiple parameters and help you exchange them. Assuming you want to rename multiple files, e.g. file1\_alt.txt and file2\_alt.txt, to alt\_file1.txt and alt\_file2.txt, the mmv syntax is as follows: mmv "\*\_\*" "#2\_#1" refers to any files with an underscore in the middle, and tells mmv to swap the name parts immediately preceding and following the underscore.

The mmv package also includes the mcp ("multiple copy"), mad ("multiple append") and mln ("multiple link") commands; and all of these tools have some extremely useful parameters, such as -v, which outputs a status message on screen (file1\_alt.txt -> alt.txt\_file1 : done). Specifying the -s flag will not move the file but instead create a symbolic link to the old file with the new name. ■

### GLOSSARY

**Wildcards:** Wildcards act as replacements for single characters or groups of characters within strings. A question mark "?" in a string replaces a single character (e.g. h?llo => hallo, hGlllo, h7llo,...) at the position marked by the

question mark, whereas an asterisk "\*" can represent any number of characters (e.g. g\*t => get, goat). In the Bash you can address any files that with the ".html" extension by typing \*.html.

### INFO

- [1] Heike Jurzik: "The Right Path – Command Line: pwd, cd, pushd, popd and dirs", Linux Magazine, Issue 34, September 2003, p78-79 [http://www.linux-magazine.com/issue/34/pwd\\_cd.pdf](http://www.linux-magazine.com/issue/34/pwd_cd.pdf)
- [2] <http://rpmseek.com/>
- [3] <http://www.rpmfind.net/>

# SELLING OUT FAST!

For more information see:  
[www.linux-magazine.com/Backissues](http://www.linux-magazine.com/Backissues)