

Dealing with spaces in filenames

BASH for loop works nicely under UNIX / Linux / Windows and OS X while working on set of files. However, if you try to process a for loop on file name with spaces in them you are going to have some problem. For loop uses \$IFS variable to determine what the field separators are. By default \$IFS is set to the space character. There are multiple solutions to this problem.

Set \$IFS variable

Try it as follows:

```
#!/bin/bash
SAVEIFS=$IFS
IFS=$(echo -en "\n\b")
for f in *
do
    echo "$f"
done
IFS=$SAVEIFS
```

OR

```
#!/bin/bash
SAVEIFS=$IFS
IFS=$(echo -en "\n\b")
# set me
FILES=/data/*
for f in $FILES
do
    echo "$f"
done
# restore $IFS
IFS=$SAVEIFS
```

More examples using \$IFS and while loop

Now you know that if the field delimiters are not whitespace, you can set IFS. For example, while loop can be used to get all fields from /etc/passwd file:

```
....
while IFS=: read userName passWord userID groupID geCos homeDir userShell
do
    echo "$userName -> $homeDir"
done < /etc/passwd
```

source: <http://www.cyberciti.biz/tips/handling-filenames-with-spaces-in-bash.html>

Revision #3

Created 2026-04-01 17:13:29 CEST by Philip

Updated 2026-04-13 19:24:55 CEST by Philip