If you like the nice curses based „Finder" from the unix world called „**Midnight commander**" and would like to use the supplied **mc-wrapper.sh-Script**, that changes the current working directory to mc's last selected one, you're a bit lost on the Mac platform, because it doesn't work. (I've installed mc using the wonderful **Homebrew**)

Why? Interesting you ask. Let's look into the mc source code. Specifically, the file *lib/vfs/interface.c*. In line 916, the function „mc_tmpdir" is defined:

```
mc_tmpdir (void)
{
    static char buffer[64];
    static const char *tmpdir = NULL;
    const char *sys_tmp;
    struct passwd *pwd;
    struct stat st;
    const char *error = NULL;
```

You might have noticed the variable „buffer" with a size of 64. That's the character buffer the temporary directory will be squeezed into. Now, let's look at the Linux platform. Where is a temporary directory located on a Linux operating system? Yes, *\/tmp*. Exactly. And mc designs it temporary directory by using that temporary directory, the prefix „mc-" and then the currently logged in user.

Mac OS does that differently. Mac OS (since 10.5 that is) holds the temporary files in its /var/folders-directory tree. James Reynolds did a very nice investigation about that directory tree. Basically, it's just a very long folder name with crypting (random) characters in it. The emphasis here is **very long folder name**.

Because when you got a long username (like dennis.ploeger for example), 64 bits tend to get really short:

```
$ echo "${TMPDIR}mc-dennis.ploeger" | wc -c
67
```

So in my case, mc's temporary folder ends in „mc-dennis.plo". This is something, that isn't taken into account when it comes to mc-wrapper.sh. So, to use that script, add a line after the second line, so it looks like this:

```
MC_USER=`id | sed 's/[^(]*(//;s/).*//'`
MC_TMP="${TMPDIR-/tmp}/mc-$MC_USER"
MC_PWD_FILE="${MC_TMP:0:63}/mc.pwd.$$"
```

Now, you can just source it and use it as described in the link above.