

Howto: Run PAUP on the server

Yes I know, this Howto is basically a modified copy of the Howto: Run STRUCTURE on the server (command line version). Doesn't make it less useful:

1. Modify *.bashrc* on *evo-lutra*

In order to run the command line version of PAUP on *evo-lutra*, you'll first need to modify one certain file on *evo-lutra* (*.bashrc*). The reason for this is that whenever you type a shell command (e.g. *paup*), the system needs to know which software to start. It searches for an executable file called "paup" in a list of folders (this list is called *\$PATH*). However, the folder which contains the PAUP executable is not yet in this *\$PATH*, so we'll have to include it first.

Connect to *evo-lutra* using Secure Shell. I explained this in detail in the Howto: Run software on the server (general). Here's the short version: Use the Terminal and type:

```
ssh yourusername@evo-lutra.zoo.unibas.ch
```

and enter your password. Then type:

```
ls -a
```

You know the *ls* command already. Using the *-a* option, hidden files (those starting with a period) are also shown. The file *.bashrc* is included in the list:

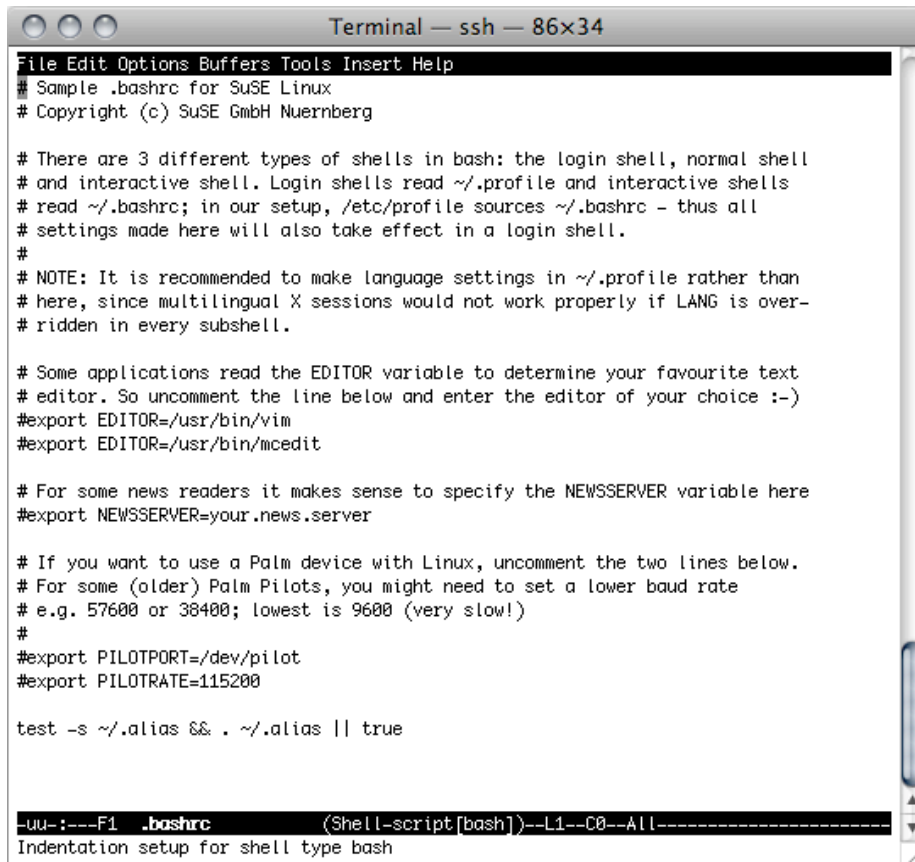


```
Terminal — ssh — 77x13
micham@evo-lutra:~> ls -a
.          Documents  .gnu-emacs  .profile   .xcoralrc
..         .dvipsrc   .inputrc   public_html .xemacs
AppleVolumes .emacs    .kernrc    .ssh       .xim.template
.bash_history .emacs.d  .lesshst   .structure .xinitrc.template
.bashrc     .exrc     .mozilla   .urview    .xtalkrc
.bashrc~    .fontconfig .muttrc    .viminfo
bin         .fonts    .mysql_history .vimrc
.bup_exclude .forward  PRIV_SHARE .xauthority
micham@evo-lutra:~>
```

Use the command line text editor *emacs* to modify *.bashrc*. To do so, type:

```
emacs .bashrc
```

You'll see the following:



```
Terminal — ssh — 86x34
File Edit Options Buffers Tools Insert Help
# Sample .bashrc for SuSE Linux
# Copyright (c) SuSE GmbH Nuernberg

# There are 3 different types of shells in bash: the login shell, normal shell
# and interactive shell. Login shells read ~/.profile and interactive shells
# read ~/.bashrc; in our setup, /etc/profile sources ~/.bashrc - thus all
# settings made here will also take effect in a login shell.
#
# NOTE: It is recommended to make language settings in ~/.profile rather than
# here, since multilingual X sessions would not work properly if LANG is over-
# ridden in every subshell.

# Some applications read the EDITOR variable to determine your favourite text
# editor. So uncomment the line below and enter the editor of your choice :-)
#export EDITOR=/usr/bin/vim
#export EDITOR=/usr/bin/mcedit

# For some news readers it makes sense to specify the NEWSERVER variable here
#export NEWSERVER=your.news.server

# If you want to use a Palm device with Linux, uncomment the two lines below.
# For some (older) Palm Pilots, you might need to set a lower baud rate
# e.g. 57600 or 38400; lowest is 9600 (very slow!)
#
#export PILOTPORT=/dev/pilot
#export PILOTRATE=115200

test -s ~/.alias && . ~/.alias || true

uu-:---F1 .bashrc (Shell-script[bash])--L1--C0--All-----
Indentation setup for shell type bash
```

Move the cursor to the end of the file, below “test -s ~/.alias...”, and type:

```
export PATH=/usr/local/paup:$PATH
```

Close emacs by typing the key combination

Ctrl-X, Ctrl-C

before closing, emacs will ask you whether you want to save the file (in the window's bottom line).

Confirm with “y”.

Close the Terminal window and open a new Terminal session. Log back in to evo-lutra. See whether the system knows now where paup is by typing

which paup

You should receive the answer

```
/usr/local/paup/paup
```

2. Start PAUP

Simply type:

```
paup
```