

常用Bash技巧总结3

Alex / 2020-04-19 / free_learner@163.com / AlexBrain.cn

更新于2023-07-12，主要是文字排版上的更新，内容基本保持不变。

总结最近学习的一些Bash技巧，包括在Bash中调用Matlab脚本并传递变量、远程访问服务器常用的命令（断点续传、保持会话不中断、同时打开多个窗口）、如何记录运行的命令。

一、将Bash变量传递给Matlab

在Bash脚本中调用Matlab脚本，可以这样实现：

```
matlab -nodisplay -nosplash -nodesktop -r "run('myscript.m');exit;"
```

如何将Bash脚本中的变量传递给Matlab呢？比如切换到某个指定的路径：

```
MYDIR=/home/alex/data
matlab -nodisplay -nosplash -nodesktop -r
"cd('${MYDIR}');run('myscript.m');exit;"
```

也就是说直接用 `$` 索引变量即可。

二、使用rsync实现断点续传

从服务器上下载大量数据，容易出现网络中断，使用rsync进行断点续传可以节约时间：

```
rsync -rP --append-verify -e 'ssh -p port_number' $source_path $target_path
```

其中 `-r` 表示读取文件夹，`-P` 表示保留部分传递的文件，`-append-verify` 就是断点续传。

三、保持ssh会话不中断

通过ssh访问服务器处理数据时，一旦长时间不做操作，ssh会话就会中断，就需要重新登录，非常麻烦。通过自动向服务器发送信号，就可以避免这一点：

1. 新建 `~/.ssh/config` 文件。
2. 在文件中输入：

```
Host *
ServerAliveInterval 240
```

3. 修改文件权限:

```
chmod 600 ~/.ssh/config
```

其中240表示每隔240s发送一次信号。

四、screen基本用法

在远程访问服务器时，有时候需要打开多个窗口，可以使用screen实现：

```
screen ## 启动screen
Ctrl+C ## 同时按键盘的Ctrl和C键新建screen会话
Ctrl+A+Number ## 同时按键盘的Ctrl和A键以及数字键在不同会话之间切换
screen -R ## 查看所有screen会话及其PID
screen -r PID ##恢复PID对应的会话
exit ## 关闭当前窗口
```

五、记录运行的命令

通常情况下，我会写一个脚本来处理数据，脚本中难免会有错误的地方，如果能记录下真实运行的命令，就能检查是否有bug存在，也便于以后查阅，在网上查到两种实现方法：

1. 在脚本中添加 `set -x/set +x` 或使用 `-x` 选项执行脚本 `bash -x script.sh`

上述方式都可以在执行脚本的同时，输出运行的命令，并在命令前用 `+` 标记。比如：`bash -x script.sh > stdout.log 2> cmd.log` 可以把脚本的输出保存在 `stdout.log` 文件中，把运行过的命令保存在 `cmd.log` 中。

2. 另一种思路是把要运行的命令当成一个字符串或者数组，分成两步来实现输出命令和执行命令的目的。比如：

```
myArray=(some commands) ## 将要执行的命令保存成数组myArray
echo "${myArray[@]}" ## 输出要执行的命令
"${myArray[@]}" ## 执行要执行的命令
```

又比如：

```
myString='some commands' ## 将要执行的命令保存成数组myString
echo $myString ## 输出要执行的命令
eval $myString ## 执行要执行的命令
```