

# 冯柏淋一生中遇到过的各种问题的解决办法

冯柏淋

November 12, 2020

## Contents

|  |          |
|--|----------|
| <b>第一章 ssh</b>                           | <b>2</b> |
| 1.1 通过跳板机连接服务器 . . . . .                 | 2        |
| 1.1.1 password . . . . .                 | 2        |
| 1.1.2 public key . . . . .               | 2        |
| 1.2 本地 pycharm 使用远程 python 解释器 . . . . . | 2        |
| <b>第二章 latex</b>                         | <b>4</b> |
| 2.1 简单的代码 listings . . . . .             | 4        |
| 2.2 添加超链接 . . . . .                      | 4        |
| <b>第三章 matplotlib</b>                    | <b>6</b> |
| <b>第四章 macOS 相关</b>                      | <b>7</b> |
| 4.1 homebrew 换源 . . . . .                | 7        |

# 第一章 ssh

## 1.1 通过跳板机连接服务器

现有本地主机 A,跳板机器 B(公网 ip:113.114.115.116)和远程服务器 C(仅内网 ip:192.168.1.1),需要从主机 A 连接到服务器 C 上。两种登陆方式的 ssh 配置方式如下。

### 1.1.1 password

在 A 的.ssh/config 文件中做如下配置。

---

```
host B
    hostname 113.114.115.116
    user jump
    port 6666
host C
    hostname 192.168.1.1
    proxycommand ssh -p 6666 B -W %h:%p
```

---

然后可以直接在主机 A 的终端上 ssh 连接 C。需要依次输入 B 和 C 的登陆密码。

### 1.1.2 public key

首先在 A 上通过ssh-keygen生成Aid\_rsa和Aid\_rsa.pub两个密钥文件。

然后通过如下命令将 pubkey 复制到跳板机中。

---

```
A> ssh-copy-id -i ~/.ssh/Aid_rsa.pub username@B_ip_addr
```

---

同样在 B 上通过 ssh-keygen 生成 Bid\_rsa 和 Bid\_rsa.pub 两个密钥文件,将其 pubkey 复制到远程服务器上。

---

```
B> ssh-copy-id -i ~/.ssh/Aid_rsa.pub username@C_ip_addr
```

---

最后在 A 的.ssh/config 文件中做如下配置:

---

```
host B
    hostname 113.114.115.116
    user jump
    port 6666
    identityfile ~/.ssh/fbl_id_rsa
host C
    hostname 192.168.1.1
    user fengbailin
    proxycommand ssh -o 'ForwardAgent yes' B 'ssh-add /home/B/.ssh/Bid_rsa && nc %h %p'
```

---

即可在 A 主机上通过ssh C命令直接连接到服务器 C,而无需密码。

## 1.2 本地 pycharm 使用远程 python 解释器

---

```
ssh -L [arbitrary port on localhost]:[remote server IP]:[remote port] [jumper_usrname]@[jumper_IP] -p [jumper port]
```

---

e.g.:

---

```
ssh -L 6231:192.168.1.151:22 jump
```

---

然后在 pycharm 中完成相应的 ssh 解释器的配置即可。

## 第二章 latex

### 2.1 简单的代码 listings

使用 listings 包, 或者使用 fancyvrb 包, 参考[这里](#)。

```
\usepackage{listings}
\usepackage{color}

\definecolor{dkgreen}{rgb}{0,0.6,0}
\definecolor{gray}{rgb}{0.5,0.5,0.5}
\definecolor{mauve}{rgb}{0.58,0,0.82}

\lstset{frame=tb,
  language=Python,
  aboveskip=3mm,
  belowskip=3mm,
  showstringspaces=false,
  columns=flexible,
  basicstyle={\small\ttfamily},
  numbers=none,
  numberstyle=\tiny\color{gray},
  keywordstyle=\color{blue},
  commentstyle=\color{dkgreen},
  stringstyle=\color{mauve},
  breaklines=true,
  breakatwhitespace=true,
  tabsize=3
}

\begin{lstlisting}{language=java}
  code here
\end{lstlisting}
```

或者使用\verbatim环境:

```
\begin{verbatim*}
  import os
  import matplotlib.pyplot as plt

  print('hello, world')
  plt.show()
\end{verbatim*}
```

行内则使用:

使用\verb|ssh dm| 则可以连接到 dm 服务器

### 2.2 添加超链接

```

\usepackage{fancyvrb} % for fancy code lstlisting
begin{Verbatim}[formatcom=\color{dkgreen}]
  \usepackage{hyperref}
  \hypersetup{
    colorlinks=true,
    linkcolor=blue,
    filecolor=magenta,
    urlcolor=cyan,
  }

  参考\href{https://mirror.tuna.tsinghua.edu.cn/help/homebrew/}{这里}。

end{Verbatim}

```

### 第三章 matplotlib

## 第四章 macOS 相关

### 4.1 homebrew 换源

参考[这里](#)。

---

```
# brew 程序本身, Homebrew/Linuxbrew 相同
git -C "$(brew --repo)" remote set-url origin https://mirrors.tuna.tsinghua.edu.cn/git/homebrew/brew.git

# 以下针对 mac OS 系统上的 Homebrew
git -C "$(brew --repo homebrew/core)" remote set-url origin
    https://mirrors.tuna.tsinghua.edu.cn/git/homebrew/homebrew-core.git
git -C "$(brew --repo homebrew/cask)" remote set-url origin
    https://mirrors.tuna.tsinghua.edu.cn/git/homebrew/homebrew-cask.git
git -C "$(brew --repo homebrew/cask-fonts)" remote set-url origin
    https://mirrors.tuna.tsinghua.edu.cn/git/homebrew/homebrew-cask-fonts.git
git -C "$(brew --repo homebrew/cask-drivers)" remote set-url origin
    https://mirrors.tuna.tsinghua.edu.cn/git/homebrew/homebrew-cask-drivers.git

# 以下针对 Linux 系统上的 Linuxbrew
git -C "$(brew --repo homebrew/core)" remote set-url origin
    https://mirrors.tuna.tsinghua.edu.cn/git/homebrew/linuxbrew-core.git

# 更换后测试工作是否正常
brew update
```

---