

# ubuntu 安装

## 目录

- \*\*\*\*\*仅供参考\*\*\*\*\*
- 一 安装ubuntu
- 二 更新驱动
- 三 连接Wi-Fi
  - 配置网络
  - 下载度管家
    - 配置VPN代理
- 四 安装docker
- 五 icode
  - 第一步：每个用户生成一对自己专属的SSH密钥
  - 第二步：在~/.ssh/config中指定各个用户的密钥地址
  - 第三步：把自己的ssh公钥（.pub文件内容）贴到icode上
- 六 zsh
  - 1：检查
  - 2、安装zsh
  - 3、安装oh-my-zsh
  - 4、安装插件
- 七 ssh
- 八 pyenv
  - 安装pyenv
  - 安装python
  - 利用virtualenv 创建虚拟python环境

\*\*\*\*\*仅供参考\*\*\*\*\*

## 一 安装ubuntu

直接使用做好的 U盘就成

## 二 更新驱动

主要就是显卡驱动

## Dell precision 7760 安装ubuntu后不识别wifi网卡?

工作站的wifi模块型号是intel的AX210，对kernel要求高于5.10，因此需要先升级内核。

### 1. 内核升级方式参考

由于网卡驱动版本需要较高版本Linux内核，先去官网下载5.10版本的内核，<https://kernel.ubuntu.com/~kernel-ppa/mainline/v5.10/>，下载下面这4个文件后，`sudo dpkg -i *.deb` 安装。

202207更新：对于Ubuntu 18.04，建议升级到v5.11，<https://kernel.ubuntu.com/~kernel-ppa/mainline/v5.11/>。v5.10版本下的蓝牙会有问题，v5.12与系统和显卡会发生冲突。

&lt;/&gt;

Python

```
1 linux-headers-5.10.0-051000_5.10.0-051000.202012132330_all.deb
2 linux-headers-5.10.0-051000-generic_5.10.0-051000.202012132330_amd64.deb
3 linux-image-unsigned-5.10.0-051000-generic_5.10.0-051000.202012132330_amd64.deb
4 linux-modules-5.10.0-051000-generic_5.10.0-051000.202012132330_amd64.deb
```

载入内核：开机时按下F2，进入boot 选项界面，不使用security boot

### 2. 安装网卡驱动(驱动可以用这个)

&lt;/&gt;

Plain Text

```
1 sudo dpkg -i linux-firmware_1.202_all.deb
```

**NOTE: kenerl更新非常重要，千万不要下错文件，下错文件直接会导致系统崩溃，踩过一个坑就是apt-get install 很多都装不上比如，vim g++**



linux-firmware\_1.202\_all.deb

189.3MB



linux-modules-5.11.0-051100-generic\_5.11.0-051100.202102142330\_amd64.deb

53.0MB



linux-image-unsigned-5.11.0-051100-generic\_5.11.0-051100.202102142330\_amd64.deb

10.6MB



linux-headers-5.11.0-051100\_5.11.0-051100.202102142330\_all.deb

11.0MB



linux-headers-5.11.0-051100-generic\_5.11.0-051100.202102142330\_amd64.deb

1.2MB

# 三 连接Wi-Fi

## 配置网络

安装好后一般是没网的，需要按如下步骤配置。

首先打开网络设置，按如下步骤设置，输入员工如流的账号密码

匿名身份：INTERNAL\百度邮箱前缀

用户名：INTERNAL\百度邮箱前缀

密码：百度邮箱密码

需要 Wi-Fi 网络认证

Wi-Fi 网络要求认证

需要密码或密钥来访问 Wi-Fi 网络 BAIDU。

Wi-Fi 安全性(S):

WPA 及 WPA2 企业

认证(T):

受保护的 EAP (PEAP)

匿名身份(M):

CA 证书:

(无)

☒ 不需要 CA 证书(R)

PEAP 版本(V):

自动

内部认证(I):

MSCHAPv2

用户名(U):

internal\邮箱前缀

密码(P):

.....

☐ 显示密码(W)

取消(C)

连接(O)

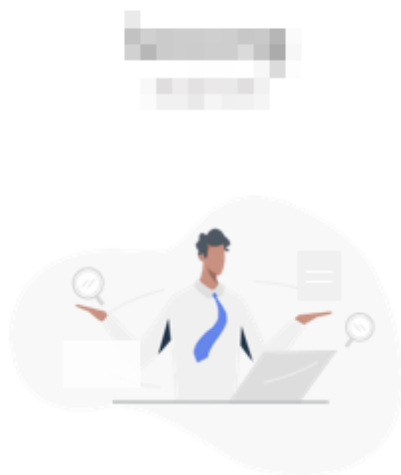
断开重连一次有线网络。



如果重连一次后终端能ping通baidu.com，但是打不开网页，应该是还需要先下载安装度管家Linux版本，看下面步骤。

## 下载度管家

<https://duguanjia.baidu.com/> 点击下载Linux客户端，拷贝到工作站中安装，登录员工如流账号。



注销

度管家

到这应该就可以上网了，但是还不能访问Google等，看下面步骤。

## 配置VPN代理

依然打开网络设置，找到VPN选项下单Networkd Proxy，打开右边齿轮设置，在Configuration URL中输入<http://pac.internal.baidu.com/bdnew.pac>，这样就可以打开谷歌了。

到这网络配置完了，后面的就是配置代码所需的环境。

## 四 安装docker

### 安装docker参考

&lt;/&gt;

Plain Text

```
1 sudo apt-get update && sudo apt-get install -y apt-transport-https ca-certificates
  curl software-properties-common && curl -fsSL
  https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add - && sudo add-apt-
  repository "deb [arch=amd64] http://mirrors.aliyun.com/docker-ce/linux/ubuntu
  $(lsb_release -cs) stable" && sudo apt-get update && sudo apt-get install -y docker-
  ce
```

为了避免每次命令都输入sudo，可以设置用户权限(将当前用户添加到docker组里面)，注意执行后须注销重新登录

&lt;/&gt;

Plain Text

```
1 sudo usermod -a -G docker $USER
2
3 ## 显示的是
4 permission denied while trying to connect to the Docker daemon socket at
  unix:///var/run/docker.sock: Get
  "http://%2Fvar%2Frun%2Fdocker.sock/v1.24/images/json": dial unix
  /var/run/docker.sock: connect: permission denied
```

安装完成Docker后，默认已经启动了docker服务，如需手动控制docker服务的启停，可执行如下命令

### 启动docker

```
sudo service docker start
```

### 停止docker

```
sudo service docker stop
```

### 重启docker

```
sudo service docker restart
```

### 登陆内部docker库

#### 1. 用docker命令登录新仓库

&lt;/&gt;

JSON

```
1 docker login -u <username> -p <passwd> iregistry.baidu-int.com
```

# 五 icode

## 第一步：每个用户生成一对自己专属的SSH密钥

用-f参数指定密钥文件的位置，把下面例子中的yangyang22换成你的名字，一路回车

&lt;/&gt;

Bash

```
1 $ ssh-keygen -t rsa -f ~/.ssh/id_rsa.chenjingyuan02
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/users/yangyang22/.ssh/id_rsa.yangyang22.
Your public key has been saved in /home/users/yangyang22/.ssh/id_rsa.yangyang22.pub.
The key fingerprint is:
23:e4:7f:df:52:92:fb:f5:49:20:d4:36:f0:2d:bc:de yangyang22@cp01-rdqa-
dev383.cp01.baidu.com
```

## 第二步：在~/.ssh/config中指定各个用户的密钥地址

【注】~/.ssh/config文件的权限必须是644，

&lt;/&gt;

Bash

```
1 chmod 644 ~/.ssh/config
```

按下面方式配置config文件，就能实现不同用户使用不同的密钥

&lt;/&gt;

Bash

```
1 Host icode.baidu.com
  User chenjingyuan02
  IdentityFile ~/.ssh/id_rsa.chenjingyuan02

Host icode.baidu.com
  User yinyan
  IdentityFile ~/.ssh/id_rsa.yinyan
```

git clone ssh://chenjingyuan02@icode.baidu.com就会用~/.ssh/id\_rsa.chenjingyuan02来认证

git clone ssh://yinyan@icode.baidu.com就会用~/.ssh/id\_rsa.yinyan来认证

## 第三步：把自己的ssh公钥（.pub文件内容）贴到icode上

&lt;/&gt;

Bash

```
1 cat ~/.ssh/id_rsa.chenjingyuan02.pub
```

在iCode右上角>个人设置>添加SSH Keys

## 六 zsh

### 1: 检查

1、首先，查看一下你系统里所有的shell

&lt;/&gt;

Plain Text

```
1 cat /etc/shells
```



### 2、安装zsh

&lt;/&gt;

Plain Text

```
1 apt install zsh #安装zsh
2
3 chsh -s /bin/zsh #将zsh设置成默认shell（不设置的话启动zsh只有直接zsh命令即可）
```

### 3、安装oh-my-zsh

事实上如果我们要自己配置zsh的配置的话会比较麻烦，好在github上有大佬制作了一个配置文件，“oh-my-zsh”，这是目前zsh中最流行的一个配置了。

（如果没装git请： apt install git）

一般网上的一键式安装方法是：



&lt;/&gt;

Plain Text

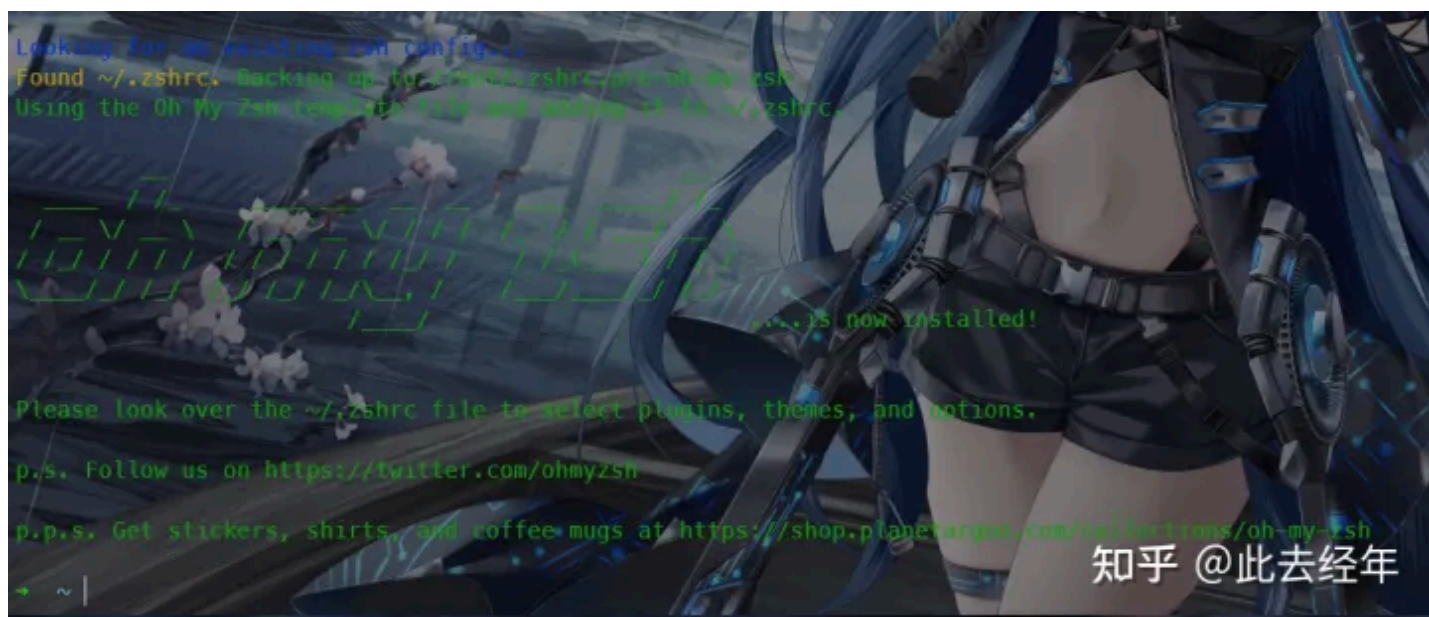
- 1 wget 下载脚本并执行（网站好像是证书过期了curl -fsSL 的下载方式不行）
- 2 wget --no-check-certificate https://github.com/robbyrussell/oh-my-zsh/raw/master/tools/install.sh -O - | sh

但是不知道为什么，我在我的centos的云服务器中安装成功了，Ubuntu子系统却是报404错误，怀疑是hosts的问题，但是问题不大，我也懒得改hosts，就找了一个国内的镜像源下载了oh-my-zsh的install.sh

&lt;/&gt;

Plain Text

- 1 sh -c "\$(curl -fsSL https://gitee.com/shmhlsh/oh-my-zsh-install.sh/raw/master/install.sh)"



出现上图那么恭喜你，安装成功。

## 4、安装插件

下载了几个官方推荐的插件比如：

&lt;/&gt;

Plain Text

- 1 #zsh-autosuggestions 命令行命令键入时的历史命令建议
- 2 git clone https://github.com/zsh-users/zsh-autosuggestions \${ZSH\_CUSTOM:-~/.oh-my-zsh/custom}/plugins/zsh-autosuggestions
- 3 #zsh-syntax-highlighting 命令行语法高亮插件
- 4 git clone https://gitee.com/Annihilater/zsh-syntax-highlighting.git \${ZSH\_CUSTOM:-~/.oh-my-zsh/custom}/plugins/zsh-syntax-highlighting

其余好用的插件等我慢慢摸索在来更新

## 七 ssh



&lt;/&gt;

Plain Text

```
1 sudo apt update
2 sudo apt install openssh-server
```

## 八 pyenv

### 安装pyenv

&lt;/&gt;

Bash

```
1 git clone https://github.com/yyuu/pyenv.git ~/.pyenv
2
3 ## 下三行 加入 .zshrc
4 export PATH=~/.pyenv/bin:$PATH
5 export PYENV_ROOT=~/.pyenv
6 eval "$(pyenv init -)"
7
8 source .zshrc
```

### 安装python

&lt;/&gt;

Bash

```
1 安装python依赖包
  sudo apt-get install make build-essential libssl-dev zlib1g-dev
  sudo apt-get install libbz2-dev libreadline-dev libsqlite3-dev wget curl
  sudo apt-get install llvm libncurses5-dev libncursesw5-dev
  sudo apt-get update
```

首先我们可以查看一下有哪些版本的python 可以安装。会列出能所有安装的python版本

&lt;/&gt;

Bash

```
1 pyenv install --list
```

### 安装python版本

&lt;/&gt;

Bash

```
1 pyenv install 3.7.0
```

## 更新数据库

&lt;/&gt;

Bash

```
1 pyenv rehash
```

## 查看安装的python版本

&lt;/&gt;

Bash

```
1 pyenv versions
```

## 切换python版本

&lt;/&gt;

Bash

```
1 xxx表示要切换的版本
2 有三种切换方式 glocal local shell
    1. glocal 全局环境,在未再次使用 glocal切换环境之前,一直使用此环境。
    2. local 本次登录环境。重启后,则环境失效,并返回当前glocal的环境。
    3. shell 局部(临时)环境。关闭命令行窗口,则环境失效,并返回当前glocal的环境。
pyenv glocal xxx
```

## 卸载某个版本

&lt;/&gt;

Bash

```
1 pyenv uninstall xxx
```

# 利用virtualenv 创建虚拟python环境

pyenv已经帮我们以plugin的形式安装好了。如果未安装,则需要我们手动安装一下

&lt;/&gt;

Bash

```
1 git clone https://github.com/pyenv/pyenv-virtualenv.git $(pyenv
root)/plugins/pyenv-virtualenv
```

但此时还不能使用。我们需要将以下命令添加到 ~/.bashrc文件结尾,这样就可以在命令行使用virtualenv了

&lt;/&gt;

Bash

```
1 echo 'eval "$(pyenv virtualenv-init -)"' >> ~/.bashrc
2 source ~/.bashrc
```

## 通过一下命令创建虚拟环境

这条命令在本机上创建了一个名为env370的python虚拟环境,这个环境的真实目录位于: ~/.pyenv/versions/

&lt;/&gt;

Bash

```
1 pyenv virtualenv 3.7.0 env370
```

注：注意，命令中的 '3.7.0' 必须是一个安装前面步骤已经安装好的python版本，否则会出错  
然后我们可以继续通过 'pyenv versions' 命令来查看当前的虚拟环境，结果如下：

&lt;/&gt;

Bash

```
1 * system (set by /home/zhai/.pyenv/version)
2 3.7.0
3 env370
```

激活虚拟环境

```
pyenv activate env370
```

1

如果要切换回系统环境，运行这个命令即可

```
pyenv deactivate
```

1

删除这个虚拟环境,只要直接删除它所在的目录就好：

```
rm -rf ~/.pyenv/versions/env370
```