

## 前言

这是一条漫长的折腾的道路，为了追求使用linux作为开发环境。可能需要付出很多的精力、时间成本，可是收获和经历是觉得值得的。

在这个过程中，会涉及到很多东西，我们可能邂逅，可能错过，并不一定深入原理。如果觉得某个东西对之后很有帮助的话，时间精力允许，请深入研究学习。

这是一篇系列文章的目录，主要是分享个人经历。或许不是一篇很详细、很深入的教程，希望还帮助到一部分，也是个人的总结。

## 为什么要使用linux

windows对于普通用户（上网冲浪、聊天看视频、玩游戏）可能是一个很容易上手、很方便的系统。但是要我承认其优秀还是有点勉强。基于windows做开发，总是会遇到一些奇奇怪怪的小问题，而且还不确定是不是开发者操作的问题；不过从xp到win7、win8经过win10，win10很多地方都要不断进步，也希望windows能越来越好。

linux对于开发者才是最好的系统，当然这里也仅限于开发者，或者编程爱好者，如果是普通用户，会面临CLI操作的操作门槛，软件生态的匮乏，学习成本高等一系列劝退问题。

这是一个取舍问题，windows好么？适合日用吧，作为开发环境的系统，有太多的小问题困扰；linux好么？适合开发吧，毕竟很多服务器系统都是用linux。作为日用，做不了太多的东西。

当然更好的选择是[OS x](#)，[苹果设备](#)，既有windows的GUI，又有linux的CLI，也是类unix系统。虽然苹果设备价格昂贵，但是省心，不用折腾太多的东西。

windows的[WSL](#)也未尝不可，[相对于vm](#)，wsl作为集成的程序，肯定是更轻量的以及更低成本的配置上手时间的，[只不过wsl还是有一些限制](#)，而且还不能直接跑docker。

说了那么多，都是个人选择，又想要win的gui，又想要linux的开发环境，那么只能windows安装vm ware在虚拟机中运行linux。

1. vm ware与linux：安装软件、系统（一）
2. vm ware与linux：配置vm ware
3. vm ware与linux：安装开发环境
4. vm ware与linux：项目文件管理

- 双系统都使用git 双账号
- 代码放宿主机，使用虚拟机运行

名称	类型	外部连接	主机连接	DHCP	子网地址
VMnet1	仅主机...	-	已连接	已启用	192.168.159.0
VMnet8	NAT 模式	NAT 模式	已连接	已启用	192.168.176.0

添加网络(E)...

移除网络(O)

重命名网络(W)...

## VMnet 信息

☐ 桥接模式(将虚拟机直接连接到外部网络)(B)

已桥接至(G):

自动设置(U)...

☒ NAT 模式(与虚拟机共享主机的 IP 地址)(N)

NAT 设置(S)...

☐ 仅主机模式(在专用网络内连接虚拟机)(H)☒ 将主机虚拟适配器连接到此网络(V)

主机虚拟适配器名称: VMware 网络适配器 VMnet8

☒ 使用本地 DHCP 服务将 IP 地址分配给虚拟机(D)

DHCP 设置(P)...

子网 IP (I): 192.168.176.0

子网掩码(M): 255.255.255.0

⚠ 需要具备管理员特权才能修改网络配置。

更改设置(C)

还原默认设置(R)

导入(T)...

导出(X)...

确定

取消

应用(A)

帮助

```

qw@ubuntu:~$ ifconfig -a
docker0: flags=4098<BROADCAST,MULTICAST> mtu 1500
    ether 02:42:91:26:03:80 txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.176.128 netmask 255.255.255.0 broadcast 192.168.176.255
    inet6 fe80::f04e:aa8c:2c63:1e71 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:63:7e:4d txqueuelen 1000 (Ethernet)
    RX packets 73 bytes 14893 (14.8 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 82 bytes 9119 (9.1 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 2736 bytes 195067 (195.0 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 2736 bytes 195067 (195.0 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

qw@ubuntu:~$ _

```

## 重启网卡

[How to Restart a Network in Ubuntu \[Beginner' s Tip\]](#)

```
#!/bin/bash
```

```
#restart network system
```

```
systemctl restart NetworkManager.service
```

```
service network-manger restart
```

```
nmcli networking off
```

```
nmcli networking on
```

```
ifdown -a && sudo ifup -a
```

```
echo "done!"
```

```
chmod +x ./network.sh
```

```
sudo ./network.sh
```

## 安装ssh

```
sudo apt-get install openssh-server
```

## 阿里Ubuntu 镜像

<https://developer.aliyun.com/mirror/ubuntu>

## Ubuntu 镜像

[清华大学镜像](#)

[apt镜像更换](#)

**如何在 Ubuntu 20.04 上安装 GCC(build-essential)GCC(build-essential)**

[如何在 Ubuntu 20.04 上安装 GCC\(build-essential\)](#)

[How to install GCC on Ubuntu](#)

[nvm使用taobao镜像](#)

[下载 mirror-config-china 设置镜像](#)

`npm i -g mirror-config-china --registry=https://registry.npm.taobao.org`

`npm i -g nrm`

下载yrm

`npm install -g yrm`