# 使用经验

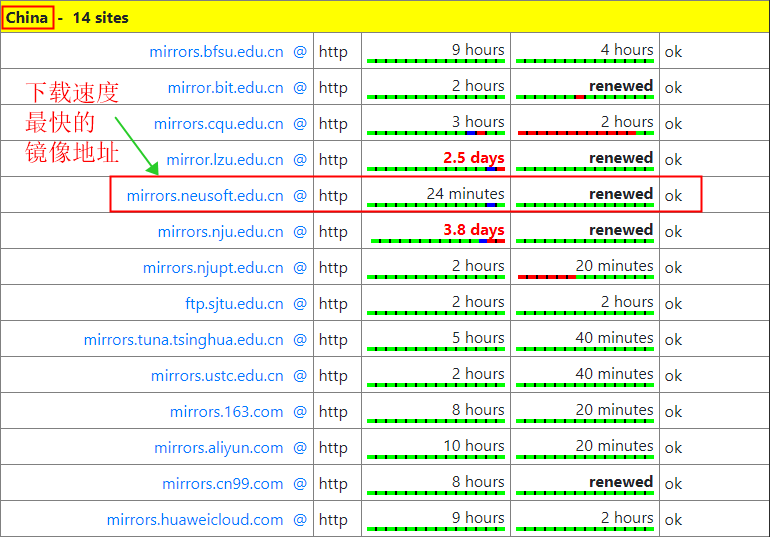
CentOS 全部版本下载地址：

<http://vault.centos.org/>

## 安装CentOS7虚拟机

### 系统下载

打开CentOS的镜像地址：<https://mirror-status.centos.org/>



也就是 <http://mirrors.neusoft.edu.cn/centos/> 这个地址。

一路点击下去到：

<http://mirrors.neusoft.edu.cn/centos/7.8.2003/isos/x86_64/>

选择[CentOS-7-x86\_64-DVD-2003.iso](http://mirrors.neusoft.edu.cn/centos/7.8.2003/isos/x86_64/CentOS-7-x86_64-DVD-2003.iso)进行下载。

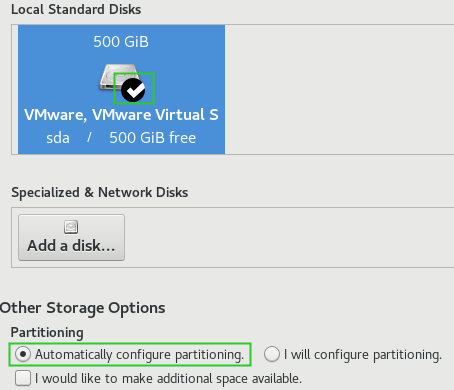
最新版本的CentOS可通过 <https://developer.aliyun.com/mirror/> 进行下载。

### 硬件配置

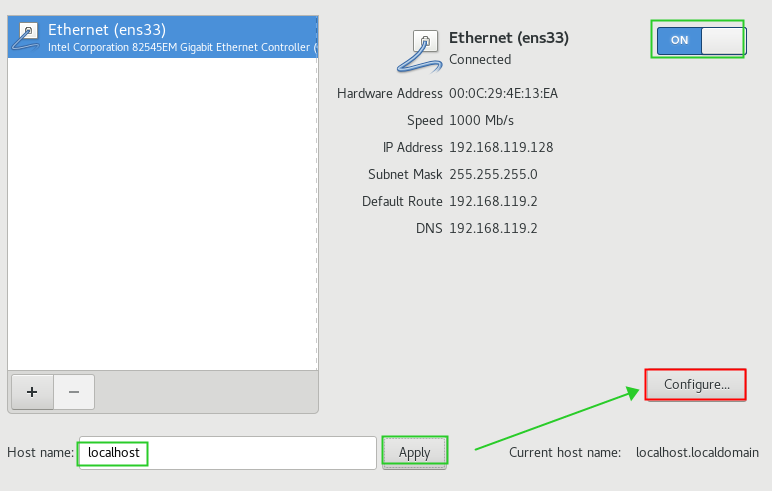
内存配置：8G，因为我们要安装MySQL8.0数据库。

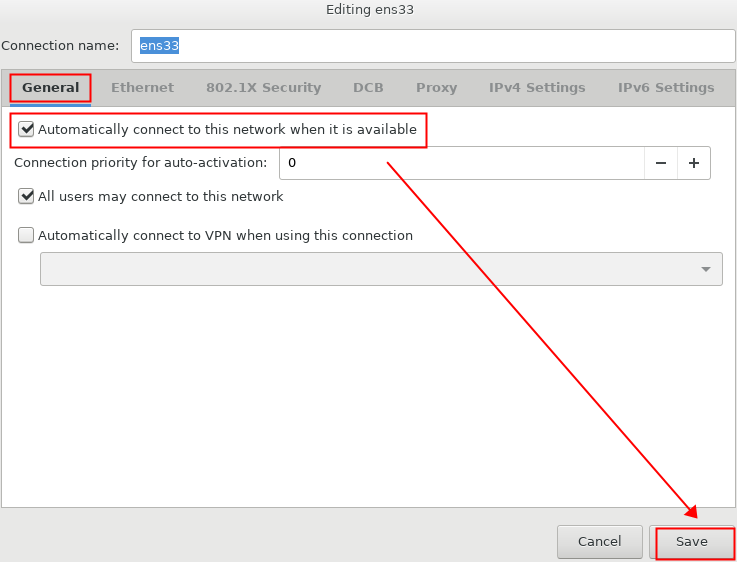
### 安装选项配置

* 选择 Server with GUI 图形界面。
* 选中磁盘(即，磁盘显示 √ 即为选中)，同时，Device Selection 选中 Automatic(默认)，如下图所示：



* 网络配置





* 账号配置：

|  |  |
| --- | --- |
| root | a5s7sh4u |
| shanhai | a5s7sh4u |

剩下的就是自动安装的过程了。

## 安装VMware Tools

## 配置root登录密码

|  |
| --- |
| sudo passwd root |

然后输入新的密码即可。

## 修改系统休眠时间

Applications → System Tools → Settings → Power → Power Saving：

Blank screen 设置为Never即可。

## 配置阿里镜像源地址

把镜像地址修改为为阿里镜像：<https://developer.aliyun.com/mirror/>

过程如下：

|  |
| --- |
| mv /etc/yum.repos.d/CentOS-Base.repo /etc/yum.repos.d/CentOS-Base.repo.backup  wget -O /etc/yum.repos.d/CentOS-Base.repo http://mirrors.aliyun.com/repo/Centos-7.repo  yum clean all  yum makecache |

参考安装地址：<https://yq.aliyun.com/articles/525282>

### CentOS 7.5 更新 yum 源

1、打开 CentOS 的 yum 目录

|  |
| --- |
| cd /etc/yum.repos.d/ |

2、用 wget 下载 repo 文件

|  |
| --- |
| wget http://mirrors.aliyun.com/repo/Centos-7.repo |

注意:

a、如果 wget 命令不生效，说明还没有安装 wget 工具，输入 yum -y install wget 回车进行安装。

3、备份系统原来的 repo 文件，其实就是重命名

|  |
| --- |
| mv CentOS-Base.repo CentOS-Base.repo.bak |

4、替换系统原来的 repo 文件

|  |
| --- |
| mv Centos-7.repo CentOS-Base.repo |

5、执行 yum 源更新命令

|  |
| --- |
| yum clean all  yum makecache //把 yum 源缓存到本地 加快软件的搜索和安装速度  yum -y update |

6、完成 yum 源更新操作。

附带阿里云和 163 的yum 源:

http://mirrors.aliyun.com/repo/

http://mirrors.aliyun.com/repo/Centos-7.repo

http://mirrors.aliyun.com/repo/Centos-6.repo

http://mirrors.aliyun.com/repo/Centos-5.repo

http://mirrors.163.com/.help/centos.html

http://mirrors.163.com/.help/CentOS7-Base-163.repo

http://mirrors.163.com/.help/CentOS6-Base-163.repo

http://mirrors.163.com/.help/CentOS5-Base-163.repo

CentOS 7 安装 Tengine

https://www.jianshu.com/p/0a2892028350

CentOS 7.4 Tengine 安装配置详解

https://blog.csdn.net/With\_\_Sunshine/article/details/84581451

安装 CentOS 7.5 以后的事

https://blog.csdn.net/qq\_34562959/article/details/82905865

CentOS 7.2 编译安装 Tengine

https://www.cnblogs.com/doseoer/p/6257939.html

yum install gcc gcc-c++ autoconf automake

## 系统更新命令

|  |
| --- |
| yum -y update |
| yum -y upgrade |

更新完成之后，重启系统。

## 查看系统版本

|  |
| --- |
| [root@local ~]# cat /etc/redhat-release  CentOS Linux release 7.8.2003 (Core)  [root@local ~]# uname -a  Linux local 3.10.0-1127.8.2.el7.x86\_64 #1 SMP Tue May 12 16:57:42 UTC 2020 x86\_64 x86\_64 x86\_64 GNU/Linux  [root@local ~]# getconf LONG\_BIT  64  [root@local ~]# |

## 系统重启命令

|  |  |  |
| --- | --- | --- |
| reboot | 普通重启 |  |
| shutdown -r now | 立刻重启 | root用户使用 |
| shutdown -r 10 | 过10分钟自动重启 | root用户使用 |
| shutdown -r 20:35 | 在时间为20:35时自动重启 | root用户使用 |

如果我们要取消shutdown设置的重启命令怎么办？输入如下命令即可：

|  |
| --- |
| shutdown -c |

## 系统关机命令

|  |  |  |
| --- | --- | --- |
| halt | 立刻关机 |  |
| poweroff | 立刻关机 |  |
| shutdown -h now | 立刻关机 | root用户使用 |
| shutdown -h 10 | 10分钟后自动关机 |  |

如果我们要取消shutdown设置的关机命令怎么办？输入如下命令即可：

|  |
| --- |
| shutdown -c |

## 查看可登陆系统的用户

|  |
| --- |
| cat /etc/passwd | grep -v /sbin/nologin | cut -d : -f 1 |

## 批量创建目录命令

脚本代码如下：

|  |
| --- |
| echo "开始时间为：`date +%Y-%m-%d,%H:%M:%S`"  for i in {1..10000};  do  mkdir one\_$i  cd one\_$i  for j in {1..10000};  do  mkdir two\_$j  done  cd ..  done  echo "结束时间为：`date +%Y-%m-%d,%H:%M:%S`" |

## 设置显示中文界面

执行如下命令：

|  |
| --- |
| gedit /etc/locale.conf |

然后把：

|  |
| --- |
| LANG="en\_US.UTF-8" |

修改为：

|  |
| --- |
| LANG="zh\_CN.UTF-8" |

并重启系统。这样就可以设置为中文界面了，反之可以设置为英文界面。

## 创建软件桌面快捷方式

我们这里以PyCharm为例说明：

解压缩 PyCharm 压缩包：

|  |
| --- |
| tar -zxvf pycharm-professional-2019.2.4.tar.gz |

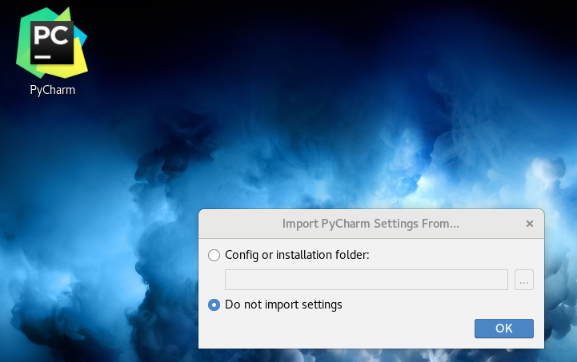
进入当前用户的 /Home/Desktop 目录，并创建如下文件：

|  |
| --- |
| gedit PyCharm.desktop |

在创建的文件中，编写如下代码：

|  |
| --- |
| [Desktop Entry]  Encoding=UTF-8  Type=Application  Name=PyCharm  Exec=/root/Downloads/PyCharm/bin/pycharm.sh  GenericName=PyCharm  Comment=Python Development Tools  Icon=/root/Downloads/PyCharm/bin/pycharm.png  Categories=Application  Terminal=false |

然后把该文件复制到桌面粘贴即可，然后 ==双击 → trust and launch== 即可。如下图所示：



# 文件列表

## 列出详细的文件和目录

|  |
| --- |
| ll -a |
| [git@local ~]$ ll -a  total 16  drwx------. 5 git git 128 May 26 13:49 .  drwxr-xr-x. 4 root root 32 May 26 11:38 ..  -rw-------. 1 git git 53 May 26 13:46 .bash\_history  -rw-r--r--. 1 git git 18 Apr 1 10:17 .bash\_logout  -rw-r--r--. 1 git git 193 Apr 1 10:17 .bash\_profile  -rw-r--r--. 1 git git 231 Apr 1 10:17 .bashrc  drwxrwxr-x. 3 git git 18 May 26 11:39 .cache  drwxrwxr-x. 3 git git 18 May 26 11:39 .config  drwxr-xr-x. 4 git git 39 May 19 16:18 .mozilla |

## 无法解压 Windows 下用 7zip 压缩的 zip文件

一般都是文件没有复制完全，注意查看文件大小。

不要直接上传，用 FTP 上传才不会出错。

需要在 CentOS 上也安装 7zip ，然后用 7zip 命令来解压缩：

打开 https://www.7-zip.org/ → Download → https://www.7-zip.org/download.html → Download p7zip for Linux → Download p7zip → http://sourceforge.net/projects/p7zip/files/ → https://sourceforge.net/projects/p7zip/files/latest/download 即可下载。

安装过程：

|  |
| --- |
| wget https://nchc.dl.sourceforge.net/project/p7zip/p7zip/16.02/p7zip\_16.02\_src\_all.tar.bz2  tar -jxvf p7zip\_16.02\_src\_all.tar.bz2  cd p7zip\_16.02/  make && make install |

使用方式：

|  |
| --- |
| 7za x list.zip |

很简单。

# 端口操作

## 查看端口开放状态

比如，查看 8888 这个端口是否开放：

|  |
| --- |
| firewall-cmd --query-port=8888/tcp |

结果为yes，说明8888端口正常开放。

## 修改 SSH 的 22 端口

在命令行输入：

|  |
| --- |
| gedit /etc/ssh/sshd\_config |

把如下代码：

|  |
| --- |
| #Port 22 |

删除，同时增加如下代码：

|  |
| --- |
| Port 20110 |

然后执行如下代码：

|  |
| --- |
| semanage port -a -t ssh\_port\_t -p tcp 20110  firewall-cmd --zone=public --add-port=20110/tcp --permanent  如果没有 semanage 则执行 yum -y install policycoreutils-python |

注意：如果没有 semanage 命令，则执行如下两个命令进行安装：

参考地址:

<https://blog.csdn.net/sean908/article/details/89020888>

<https://blog.csdn.net/runsnail2018/article/details/81185653>

|  |
| --- |
| yum provides semanage  yum install -y policycoreutils-python |

然后执行删除 22 端口的防火墙规则：

|  |
| --- |
| firewall-cmd --zone=public --remove-port=22/tcp --permanent |

然后执行重启防火墙操作：

|  |
| --- |
| firewall-cmd --reload  systemctl restart sshd  systemctl restart firewalld.service |

以下是一些查询：

|  |
| --- |
| firewall-cmd --permanent --query-port=20110/tcp # 查询防火墙是否开启某个端口号  firewall-cmd --zone=public --list-ports # 查询防火墙开放的端口号  netstat -ntlp|grep 20110 # 查看端口是否开启  semanage port -l|grep ssh # 查看 SSH 可用的端口号 |

到这里就完成了 SSH 端口号修改的操作了。

# 软件安装

## 宝塔面板

宝塔面板官方网站：<https://www.bt.cn/>

### 宝塔面板安装

CentOS7安装宝塔面板，安装参考地址：

<https://www.bt.cn/bbs/thread-19376-1-1.html>

安装命令：

|  |
| --- |
| yum install -y wget && wget -O install.sh http://download.bt.cn/install/install\_6.0.sh && sh install.sh |

然后开始自动安装，安装成功后，显示如下信息：

|  |
| --- |
| ==================================================================  Congratulations! Installed successfully!  ==================================================================  Bt-Panel: http://117.25.180.91:8888/67677a27  username: lqahvonw  password: 73dcd792  If you cannot access the panel,  release the following panel port [8888] in the security group  若无法访问面板，请检查防火墙/安全组是否有放行面板[8888]端口  ==================================================================  Time consumed: 2 Minute! |

虚拟机本机访问的地址是：

|  |
| --- |
| http://127.0.0.1:8888/67677a27/ |

Windows客户端访问虚拟机的地址是：

|  |
| --- |
| http://192.168.119.128:8888/67677a27/ |
| 其中 192.168.119.128 为虚拟机的IP地址。 |

如果面板登陆信息丢失了，无法登陆怎么办？

在CentOS服务器上打开终端，然后输入如下命令：

|  |
| --- |
| /etc/init.d/bt default |

即可查看相关登陆地址、账号和密码。

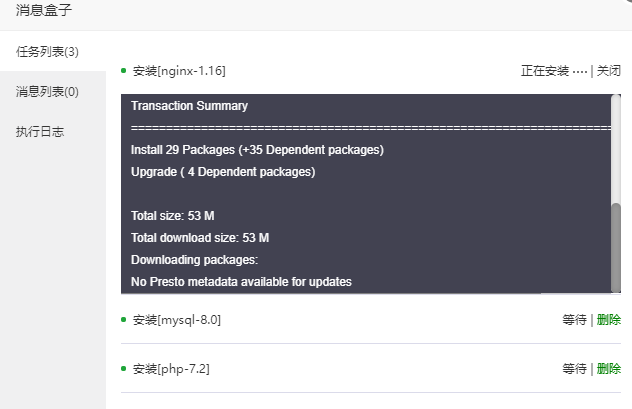
目前，把登陆地址、账号、密码修改成如下，统一管理：

|  |  |  |
| --- | --- | --- |
| URL | USERNAME | PASSWORD |
| orange | orange | a5s7sh4u |

### 选择和安装网站环境

在宝塔面板中，我们要安装MySQL8.0数据库，要求内存至少8G。







### 宝塔面板设置

为了提高安全，在面板设置界面中可对以下几项做修改：

* + 面板端口
  + 安全入口
  + 面板用户
  + 面板密码

### 宝塔面板升级

|  |
| --- |
| wget -O update.sh http://download.bt.cn/install/update\_pro.sh && bash update.sh pro |

## 搭建Git仓库

### 安装Git服务

|  |
| --- |
| yum -y install git |
| [root@local ~]# git --version  git version 1.8.3.1  [root@local ~]# |

### 创建Git用户

|  |  |  |
| --- | --- | --- |
| 1、 | 创建git用户组和用户 | adduser git |
| 2、 | 设置git用户密码 | passwd git  Changing password for user git.  New password: a5s7sh4u  Retype new password: a5s7sh4u  passwd: all authentication tokens updated successfully. |
| **adduser git**命令说明：  建立一个CentOS的用户，用户名为git，同时会自动生成一个对应的用户组git。  用户列表文件地址：/etc/passwd  用户组列表文件地址：/etc/group | | |

### 创建安全证书

创建保存安全证书的目录并创建安全证书，我们需要先切换到刚刚创建的git用户，操作步骤如下：

|  |
| --- |
| [root@local ~]# su git  [git@local root]$ cd ~  [git@local ~]$ mkdir .ssh  [git@local ~]$ ssh-keygen  Generating public/private rsa key pair.  Enter file in which to save the key (/home/git/.ssh/id\_rsa):  Enter passphrase (empty for no passphrase):  Enter same passphrase again:  Your identification has been saved in /home/git/.ssh/id\_rsa.  Your public key has been saved in /home/git/.ssh/id\_rsa.pub.  The key fingerprint is:  SHA256:C6AFmcUvQh3ae6sq+qOMCBZSsPi85BL5c8BeUU+5bMg git@local  The key's randomart image is:  +---[RSA 2048]----+  |. .\*o.. .. |  |.o++o. o. |  |o.o =o o.. |  | \*.o.+E + |  |+.B.o.o.S |  |.\*.+ . o . |  |o.\* . . . |  |B..o . |  |\*=oo. |  +----[SHA256]-----+  [git@local ~]$ |

把证书导入到authorized\_keys文件

|  |
| --- |
| [git@local ~]$ cd .ssh/  [git@local .ssh]$ cat ./id\_rsa.pub >> ./authorized\_keys |
| **或者**通过复制文件并重命名的方式来达到导入证书的目的 |
| [git@local .ssh]$ cp -r id\_rsa.pub authorized\_keys |

### 创建存放仓库的目录

在这里，我们把所有仓库都放在git用户目录下的GitRepo目录下。

|  |
| --- |
| [root@local ~]# su git  [git@local root]$ cd ~  [git@local ~]$ mkdir GitRepo  [git@local ~]$ |

### 在仓库目录创建空仓库

需要进入仓库目录，才能创建空仓库，过程如下所示：

|  |
| --- |
| [root@local ~]# su git  [git@local root]$ cd ~/GitRepo/  [git@local GitRepo]$ **git init --bare dedecms.git**  Initialized empty Git repository in /home/git/GitRepo/dedecms.git/  [git@local GitRepo]$ |

到这里，我们就已经创建了一个git空的仓库了。

当我们需要新增仓库的时候，执行该步骤进行创建就可以了。

如果是要把已存在的目录创建为git仓库，如何操作呢？

首先进入该目录；

然后执行git init 命令即可。

这样就会在目录中生成一个.git文件夹，这样目录就变成了一个git仓库。

比如：我执行如下代码创建一个目录，如下所示：

|  |
| --- |
| [git@localhost GitRepo]$ composer create-project --prefer-dist laravel/laravel ./mycms  Creating a "laravel/laravel" project at "./mycms"  Installing laravel/laravel (v7.12.0)  - Installing laravel/laravel (v7.12.0): Downloading (100%)  Created project in /home/git/GitRepo/./mycms  > @php -r "file\_exists('.env') || copy('.env.example', '.env');"  Loading composer repositories with package information  Updating dependencies (including require-dev)  …… |

然后再把mycms初始化为git仓库，然后再到Windows系统中拉取该仓库代码。

### 测试仓库服务

这里，我的客户端是Windows10系统，虚拟机的IP地址为：192.168.119.128。

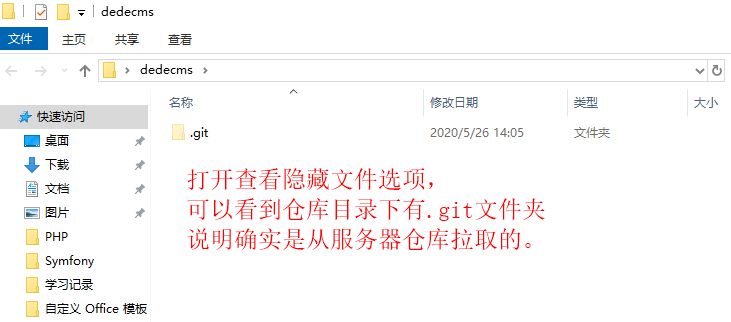
所以测试的命令为：

|  |
| --- |
| git clone git@192.168.119.128:/home/git/GitRepo/dedecms.git |

我们在客户端打开Cmder软件，然后输入以上的命令，完整过程如下：

|  |
| --- |
| C:\Users\Administrator\Desktop  λ git clone git@192.168.119.128:/home/git/GitRepo/dedecms.git  Cloning into 'dedecms'...  git@192.168.119.128's password:  warning: You appear to have cloned an empty repository. |

结果显示如下：

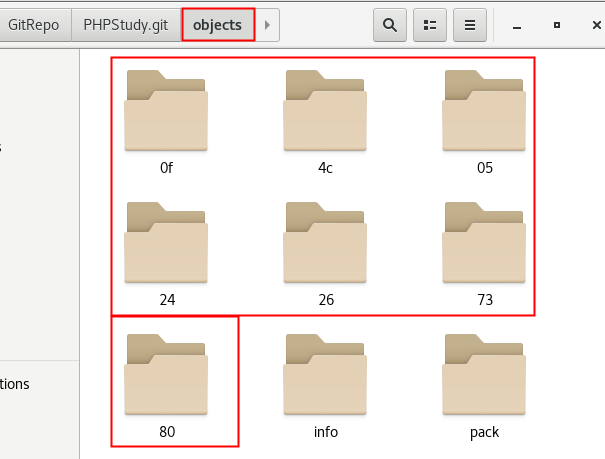


如果出现如下错误信息：

|  |
| --- |
| D:\WorkSpace  λ git clone git@192.168.119.128:/home/git/GitRepo/dedecms.git  Cloning into 'dedecms'...  @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  @ WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED! @  @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!  Someone could be eavesdropping on you right now (man-in-the-middle attack)!  It is also possible that a host key has just been changed.  The fingerprint for the ECDSA key sent by the remote host is  SHA256:B2nsMcZhyvFxU1Umu0kHkl2cEI78cA5wNozfhAFPsLM.  Please contact your system administrator.  Add correct host key in /c/Users/Administrator/.ssh/known\_hosts to get rid of this message.  Offending ECDSA key in /c/Users/Administrator/.ssh/known\_hosts:1  ECDSA host key for 192.168.119.128 has changed and you have requested strict checking.  Host key verification failed.  fatal: Could not read from remote repository.  Please make sure you have the correct access rights  and the repository exists. |

则进入C:\Users\Administrator\.ssh目录，把known\_hosts文件删除，再重新拉取即可。

如果我们添加并提交新的文件，则这些提交的文件会保存到objects目录下，如下图所示：



### 使用hooks自动部署

通过hooks，我们可以把我们提交的文件同步部署到对应的网站根目录中，如何实现这个过程呢？

|  |
| --- |
| su git  cd GitRepo/PHPStudy.git/hooks/  vim post-receive  添加 git --work-tree=/www/wwwroot/phpstudy checkout -f  chmod +x post-receive |

这样就可以把PHPStudy.git里面的所有我们提交的文件都同步到/www/wwwroot/phpstudy目录中，以下截图是同步到/www/wwwroot/phpstudy目录后的文件列表：



我们在截图中可以看到，同步过来的文件的所有者仍然是git，而不是www，但是不影响我们对网站的正常访问。

### 禁止Git账号登陆Shell

通常情况下，为了安全性着想，我们在上面使用完git账号之后，需要禁止git登陆shell的权限，操作过程如下所示：

|  |
| --- |
| gedit /etc/passwd |

然后把

|  |
| --- |
| git:x:1001:1001::/home/git:/bin/bash |

修改成如下：（这里的1001也可能是别的数字）

|  |
| --- |
| git:x:1001:1001::/home/git:/usr/bin/git-shell |

由于我们是在本地的CentOS虚拟机上操作的，可能需要创建多个仓库，所以这里我们就不禁止了。

## 安装rpm包

可以使用yum方式安装rpm包，会自动解决依赖问题，比如：

|  |
| --- |
| rpm -ivh google-chrome-stable\_current\_x86\_64.rpm |

出现如下问题：

|  |
| --- |
| warning: google-chrome-stable\_current\_x86\_64.rpm: Header V4 DSA/SHA1 Signature, key ID 7fac5991: NOKEY  error: Failed dependencies:  libXss.so.1()(64bit) is needed by google-chrome-stable-83.0.4103.97-1.x86\_64  libappindicator3.so.1()(64bit) is needed by google-chrome-stable-83.0.4103.97-1.x86\_64  liberation-fonts is needed by google-chrome-stable-83.0.4103.97-1.x86\_64  libvulkan.so.1()(64bit) is needed by google-chrome-stable-83.0.4103.97-1.x86\_64 |

修改成yum方式安装：

|  |
| --- |
| yum -y install google-chrome-stable\_current\_x86\_64.rpm |

安装结果如下：

|  |
| --- |
| Installed:  google-chrome-stable.x86\_64 0:83.0.4103.97-1  Dependency Installed:  libXScrnSaver.x86\_64 0:1.2.2-6.1.el7 libappindicator-gtk3.x86\_64 0:12.10.0-13.el7  libdbusmenu.x86\_64 0:16.04.0-4.el7 libdbusmenu-gtk3.x86\_64 0:16.04.0-4.el7  liberation-fonts.noarch 1:1.07.2-16.el7 liberation-narrow-fonts.noarch 1:1.07.2-16.el7  libindicator-gtk3.x86\_64 0:12.10.1-6.el7 vulkan.x86\_64 0:1.1.97.0-1.el7  vulkan-filesystem.noarch 0:1.1.97.0-1.el7  Complete! |

可以看出yum方式安装解决了依赖问题。

## deb转化为rpm

第一步：安装alien

|  |
| --- |
| yum install epel-release  rpm --import http://li.nux.ro/download/nux/RPM-GPG-KEY-nux.ro  rpm -Uvh http://li.nux.ro/download/nux/dextop/el7/x86\_64/nux-dextop-release-0-5.el7.nux.noarch.rpm  yum update && yum install alien |

到这一步就安装完了alien。

第二步：把.deb转化为.rpm软件包

|  |
| --- |
| alien --to-rpm --scripts xxxxxx\_amd64.deb |

参考地址：

<https://www.tecmint.com/convert-from-rpm-to-deb-and-deb-to-rpm-package-using-alien/>

## ifconfig

|  |
| --- |
|  |

## 安装 GNOME 图形桌面 和 远程桌面连接

1、安装 GNOME 图形桌面

2、安装远程桌面连接工具

在安装之前先更新系统：

|  |
| --- |
| yum -y update |

安装 GNOME 图形桌面：

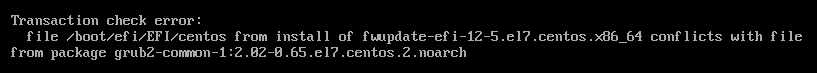
开启虚拟机 → root root → 登录系统 → 打开终端，输入如下命令：

|  |
| --- |
| yum groupinstall "GNOME Desktop" "Graphical Administration Tools"  ln -sf /lib/systemd/system/runlevel5.target /etc/systemd/system/default.target # 如果该语句不行，则使用下面的方式 |

如果出现如下错误：

|  |
| --- |
| centos 7.5 cannot find a valid baseurl for repo: base/7/x86\_64  打开 vi /etc/sysconfig/network-scripts/ifcfg-ens33 每个机子都可能不一样，但格式会是"ifcfg-eth数字"  把 ONBOOT=no 改为 ONBOOT=yes  保存后重启网络服务  service network restart  然后  ping www.baidu.com  看看能不能 ping 通 |

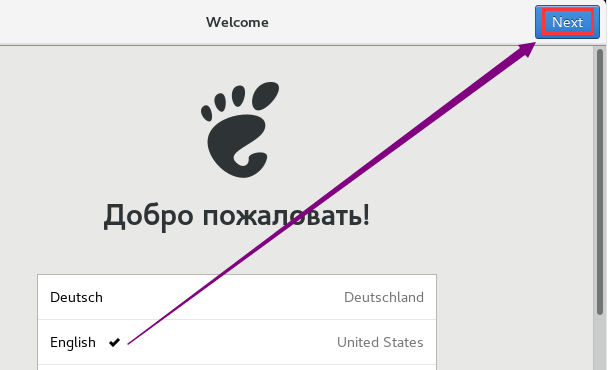
出现如下错误：

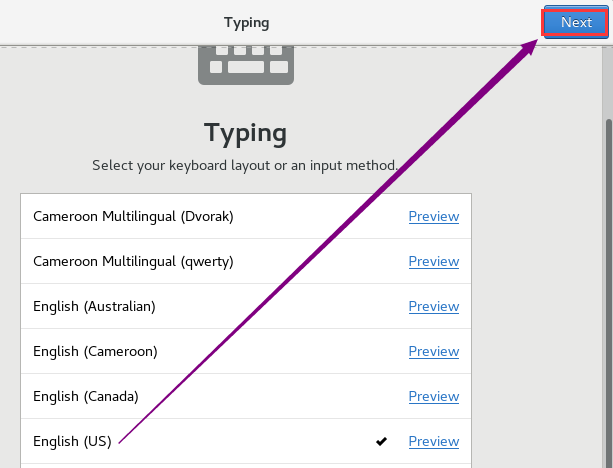


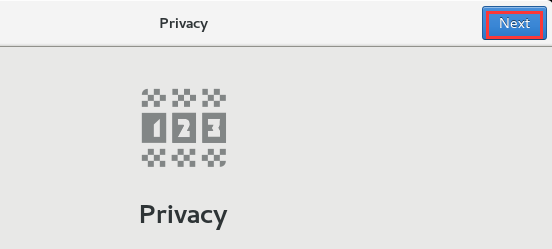
则执行：

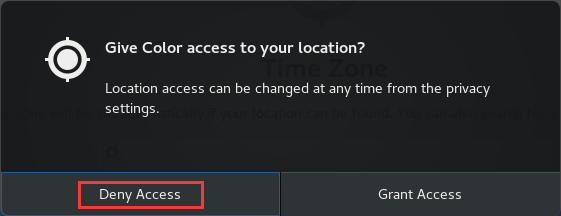
|  |
| --- |
| yum update grub2-common  yum install fwupdate-efi |

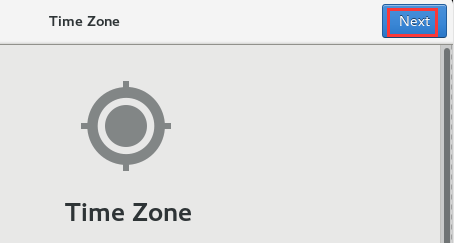
安装完成之后，重启，就可以进入图形界面了，如图：

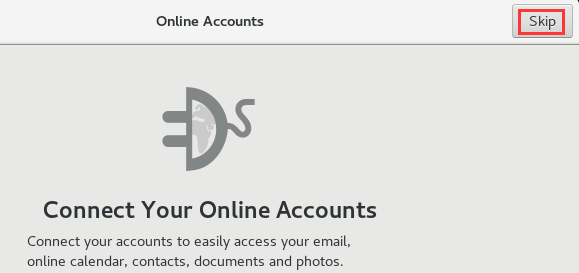


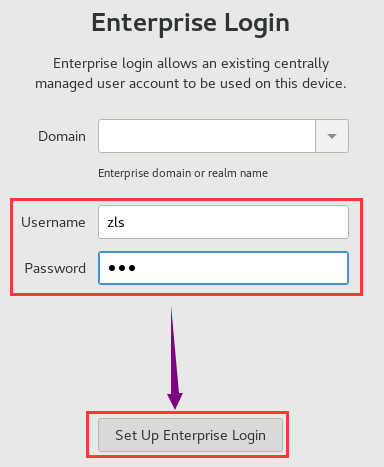


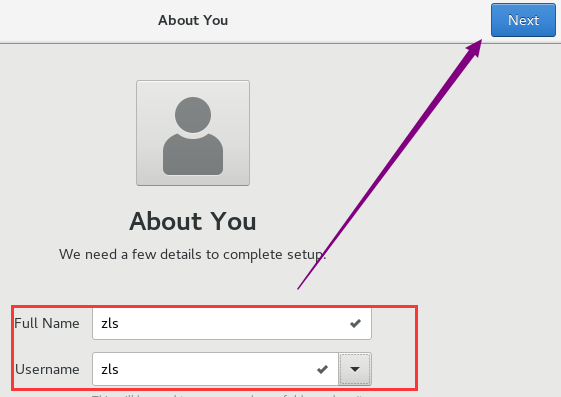


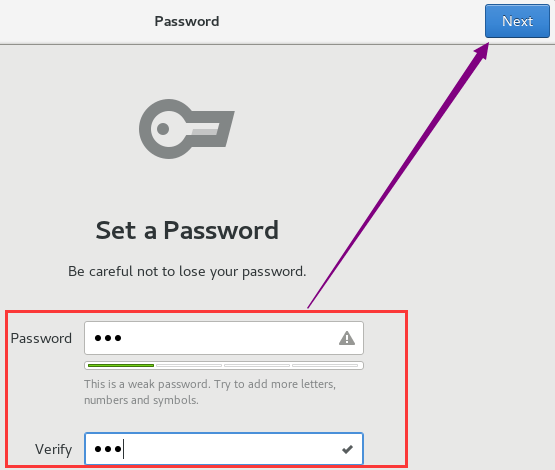












重启，用 root 登录。

图形界面与字符界面切换：

|  |
| --- |
| 图形界面 → 字符界面:  台式电脑 → Ctrl+Alt+F6  笔记本 → Ctrl+Alt+shift+Fn  字符界面 → 图形界面:  台式电脑 → Ctrl+Alt+F1  笔记本 → Alt+F7 |

取消自动锁屏：

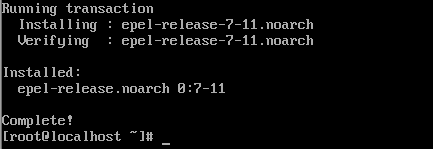
|  |
| --- |
| Application → System Tools → Setting → Power → Blank screen → Never |

**安装远程连接软件**：

|  |
| --- |
| yum -y install perl make gcc kernel-headers kernel-devel  yum install epel-release |

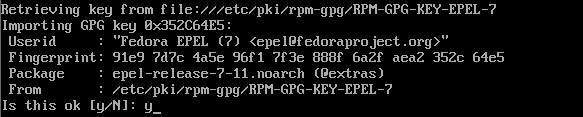


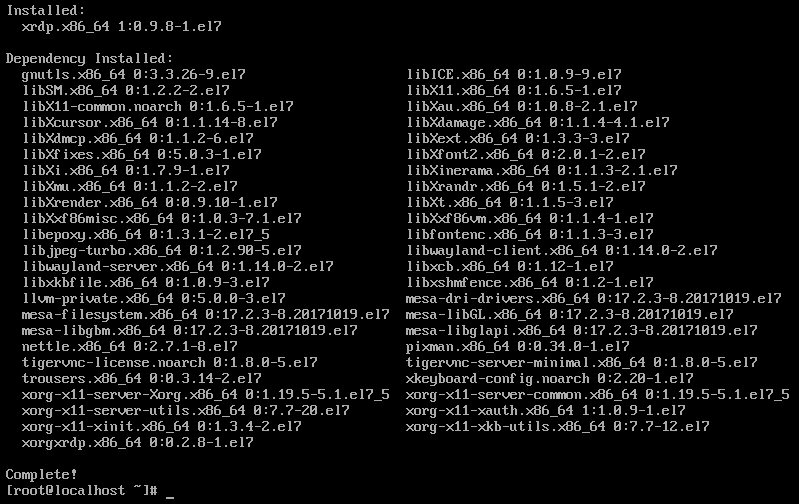




|  |
| --- |
| https://kojipkgs.fedoraproject.org/packages/xorgxrdp/  方式一:  yum install https://kojipkgs.fedoraproject.org//packages/xorgxrdp/0.2.6/2.el7/x86\_64/xorgxrdp-0.2.6-2.el7.x86\_64.rpm  yum install xrdp  方式二:  yum install https://kojipkgs.fedoraproject.org/packages/xorgxrdp/0.2.8/1.el7/x86\_64/xorgxrdp-0.2.8-1.el7.x86\_64.rpm  yum install xrdp  以上两种方式都可以正常安装 xrdp 库文件，过程如下： |







安装 tigervnc-server 库文件：

|  |
| --- |
| yum -y install tigervnc-server  yum install tigervnc # 如果需要客户端,可安装,在服务器上不需要安装 |

设置 tigervnc-server 密码：

|  |
| --- |
| 切换到 root 用户，输入  vncpasswd a5s7sh4u  Would you like to enter a view-only password (y/n)? n |

配置 xrdp.ini 文件：

|  |
| --- |
| gedit /etc/xrdp/xrdp.ini |

把 max\_bpp 的值 32 修改为 24 然后保存，并且把远程桌面连接端口号进行修改，如下：

|  |
| --- |
| [Globals]  ; xrdp.ini file version number  ini\_version=1  ; fork a new process for each incoming connection  fork=true  ; tcp port to listen  port=3389  把 3389 修改为 20183 |

配置 selinux 服务：

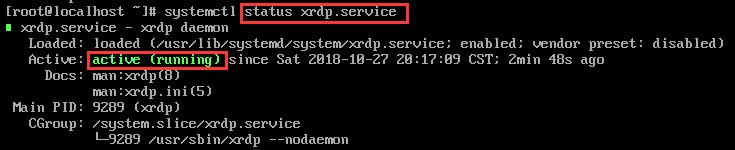
|  |
| --- |
| chcon -t bin\_t /usr/sbin/xrdp  chcon -t bin\_t /usr/sbin/xrdp-sesman |

然后，我们把 xrdp 设置为系统服务，并开启该服务：

|  |
| --- |
| systemctl start xrdp #设置为系统服务  systemctl enable xrdp #设置为开机启动  systemctl restart xrdp #重启 xrdp 服务 |

查看xrdp服务是否正常启动用如下两个命令中的任意一个：

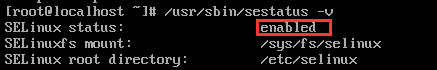
|  |
| --- |
| systemctl status xrdp.service  ss -antup|grep xrdp |



说明 xrdp 处于运行状态中。因为要进行桌面远程连接，所以 3389 端口需要打开，当然在后面我们可以把 3389 端口修改成别的端口。

查看防火墙是否打开：

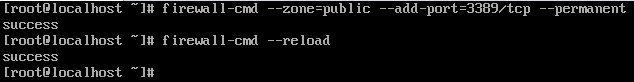
|  |
| --- |
| /usr/sbin/sestatus -v #查看 SELinux 状态 |



把远程连接端口加入防火墙规则：

|  |
| --- |
| firewall-cmd --zone=public --add-port=20183/tcp --permanent #增加  firewall-cmd --zone=public --remove-port=20183/tcp --permanent #删除  firewall-cmd --reload |

重启系统，然后就可以使用 Windows 的远程桌面连接 Linux 了。



查看所有打开的端口：

|  |
| --- |
| firewall-cmd --zone=public --list-ports |

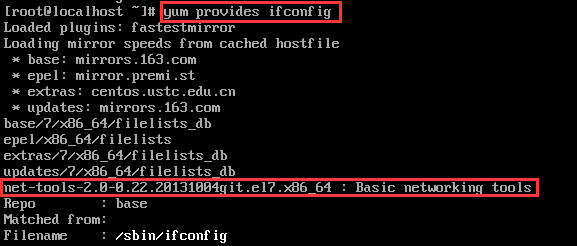
在远程连接虚拟机之前，需要查看虚拟机的端口，输入：

|  |
| --- |
| ifconfig |



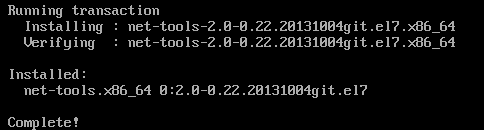
启动 ifconfig 经过如下步骤：

|  |
| --- |
| yum provides ifconfig |

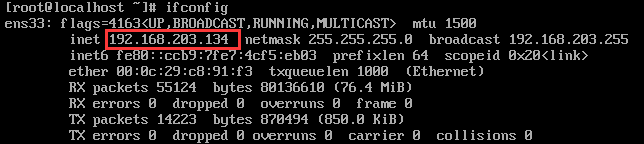


|  |
| --- |
| yum install net-tools |





然后我们就可以正常使用 ifconfig 功能了。



在设置远程桌面连接的操作中，把端口加入防火墙规则时，发现 firewalld 无效怎么办？这时候需要安装 firewalld 相关软件：

|  |
| --- |
| yum install firewalld firewall-config  systemctl start firewalld  systemctl status firewalld |

安装完成之后，重启系统，看是否成功，如果发现启动后，查看状态失败，则进行如下操作：

|  |
| --- |
| gedit /etc/dbus-1/system.d/com.foxbryant.demo.conf |

填入如下内容：

|  |
| --- |
| <!DOCTYPE busconfig PUBLIC "-//freedesktop//DTD D-BUS Bus Configuration 1.0//EN" "http://www.freedesktop.org/standards/dbus/1.0/busconfig.dtd">  <busconfig>  <policy user="root">  <allow own="com.foxbryant.demo"/>  <allow send\_destination="com.foxbryant.demo"/>  <allow send\_interface="com.foxbryant.demo"/>  </policy>  <policy user="foxbryant">  <allow own="com.foxbryant.demo"/>  <allow send\_destination="com.foxbryant.demo"/>  <allow send\_interface="com.foxbryant.demo"/>  </policy>  </busconfig> |

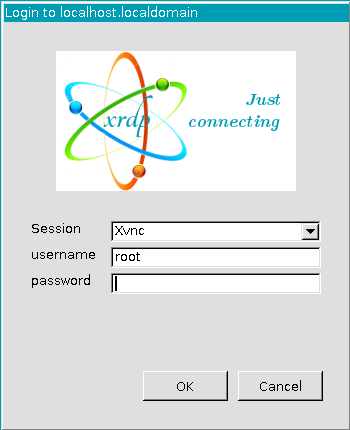
然后再次重启系统，不重启的话就算配置对了也启动失败。

firewall-cmd 命令详解：

<https://blog.csdn.net/GMingZhou/article/details/78090963>

<https://blog.csdn.net/s_p_j/article/details/80979450>

## 远程桌面登录，出现 Xrdp 登录界面怎么处理？



输入 SSH 的 root 账号和密码即可，如果是输入用户名和密码即可登录，那么很有可能是我们修改了密码，在远程登录这边没有把密码修改过来。

## 安装 Apache 2.4 软件

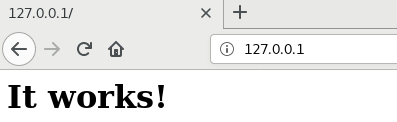
在安装软件之前，先安装开发工具环境：

|  |
| --- |
| yum groupinstall "Development Tools" |

开始安装 Apache 相关软件：

|  |
| --- |
| 官网 http://httpd.apache.org/download.cgi  wget http://archive.apache.org/dist/httpd/httpd-2.4.37.tar.gz  香港 wget http://ftp.cuhk.edu.hk/pub/packages/apache.org/httpd/httpd-2.4.37.tar.gz  wget https://mirrors.cnnic.cn/apache/apr/apr-1.6.5.tar.gz  wget https://mirrors.cnnic.cn/apache/apr/apr-util-1.6.1.tar.gz  wget https://ftp.pcre.org/pub/pcre/pcre-8.42.tar.gz  tar -zxvf httpd-2.4.37.tar.gz  tar -zxvf apr-1.6.5.tar.gz  tar -zxvf apr-util-1.6.1.tar.gz  tar -zxvf pcre-8.42.tar.gz  cp -r apr-1.6.5 /root/Downloads/httpd-2.4.37/srclib/apr  cp -r apr-util-1.6.1 /root/Downloads/httpd-2.4.37/srclib/apr-util  cp -r pcre-8.42 /root/Downloads/httpd-2.4.37/srclib/pcre  yum install -y expat-devel openssl-devel libevent-devel  cd apr-1.6.5  ./configure --prefix=/zls/server/apr  make && make install  cd apr-util-1.6.1  ./configure --prefix=/zls/server/apr-util --with-apr=/zls/server/apr  make && make install  cd pcre-8.42  ./configure --prefix=/zls/server/pcre  make && make install  cd httpd-2.4.37  ./configure --prefix=/zls/server/apache --enable-so --enable-ssl --enable-cgi --enable-rewrite --enable-zlib --with-pcre --enable-mods-shared=most --enable-module=most --enable-mpms-shared=all --with-mpm=prefork --with-included-apr --with-zlib --enable-deflate --enable-cgid --with-apr=/zls/server/apr --with-apr-util=/zls/server/apr-util --with-pcre=/zls/server/pcre  make && make install  gedit /zls/server/apache/conf/httpd.conf  在 #ServerName www.example.com:80 下面添加一行如下  ServerName localhost:80  /zls/server/apache/bin/apachectl start  firewall-cmd --zone=public --add-port=80/tcp --permanent  firewall-cmd --reload  # 如果没有防火墙则 https://www.jianshu.com/p/a2eebf4cd6a9  配置 Apache 环境变量  gedit /etc/profile  export PATH=$PATH:/zls/server/apache/bin  source /etc/profile  echo $PATH  开机启动方式一:  gedit /etc/rc.d/rc.local  /zls/server/apache/bin/apachectl start  开机启动方式二:  cp /zls/server/apache/bin/apachectl /etc/rc.d/init.d/httpd  chmod +x /etc/rc.d/init.d/httpd  chkconfig --add httpd  chkconfig httpd on  service httpd does not support chkconfig  解决过程如下：  gedit /etc/rc.d/init.d/httpd  在 #!/bin/bash 语句下添加如下语句 包括 # 部分  #chkconfig:345 61 61 //此行的345参数表示,在哪些运行级别启动,启动序号(S61);关闭序号(K61)  #description:Apache httpd //此行必写,描述服务.  service httpd start  service httpd stop  service httpd restart |

在浏览器输入 127.0.0.1，结果如下：



则表示 Apache 安装成功。

查看 Apache 是否运行：

|  |
| --- |
| ps -ef | grep httpd |

结果如下图：



说明 Apache 运行正常。

## 安装 MySQL 5.7 软件

下载 MySQL 数据库和安装：

|  |
| --- |
| https://dev.mysql.com/downloads/mysql/ → Looking for previous GA versions? → :  Select Version: 5.7.24  Select Operating System: Linux - Generic  Select OS Version: Linux - Generic (glibc 2.12) (x86, 64-bit)  Compressed TAR Archive: mysql-5.7.24-linux-glibc2.12-x86\_64.tar.gz  → Download → No thanks, just start my download. → :  wget https://dev.mysql.com/get/Downloads/MySQL-5.7/mysql-5.7.24-linux-glibc2.12-x86\_64.tar.gz  wget http://mirrors.sohu.com/mysql/MySQL-5.7/mysql-5.7.24-linux-glibc2.12-x86\_64.tar.gz  国内镜像下载:  http://mirrors.sohu.com/mysql/  http://mirrors.ustc.edu.cn/mysql-ftp/Downloads/  tar -zxvf mysql-5.7.24-linux-glibc2.12-x86\_64.tar.gz  mkdir -p /zls/server/mysql/  mv mysql-5.7.24-linux-glibc2.12-x86\_64/\* /zls/server/mysql/  mkdir -p /zls/server/mysql/data/  mkdir -p /zls/server/mysql/log/  groupadd mysql  useradd -g mysql -s /bin/false -M mysql  chmod -R 777 /zls/server/mysql/log/  chown -R mysql:mysql /zls/server/mysql/  chown -R mysql:mysql /zls/server/mysql/data/  chown -R mysql:mysql /zls/server/mysql/log/ |

编辑 my.cnf 配置文件：

|  |
| --- |
| gedit /etc/my.cnf  [client]  socket=/zls/server/mysql/data/mysql.sock  [mysqld]  port=3306  user=mysql  character\_set\_server=utf8mb4  init\_connect='SET NAMES utf8mb4'  lower\_case\_table\_names = 1  basedir=/zls/server/mysql/  datadir=/zls/server/mysql/data/  socket=/zls/server/mysql/data/mysql.sock  tmpdir=/tmp/  log-error=/zls/server/mysql/log/mysqld.log  pid-file=/zls/server/mysql/data/mysqld.pid  server-id=1  sql\_mode=NO\_ENGINE\_SUBSTITUTION,STRICT\_TRANS\_TABLES,NO\_AUTO\_CREATE\_USER,NO\_AUTO\_VALUE\_ON\_ZERO,NO\_ZERO\_IN\_DATE,NO\_ZERO\_DATE,ERROR\_FOR\_DIVISION\_BY\_ZERO,PIPES\_AS\_CONCAT,ANSI\_QUOTES  [mysql]  socket=/zls/server/mysql/data/mysql.sock  port=3306 |

初始化 MySQL 数据库：

|  |
| --- |
| cd /zls/server/mysql/bin/  ./mysqld --initialize --user=mysql --basedir=/zls/server/mysql/ --datadir=/zls/server/mysql/data/ --pid-file=/zls/server/mysql/data/mysqld.pid --tmpdir=/tmp/  如果出现 ./mysqld: error while loading shared libraries: libaio.so.1: cannot open shared object file: No such file or directory 错误，则参考https://www.cnblogs.com/hufulinblog/p/10124001.html  cat /zls/server/mysql/log/mysqld.log #文件可以找到生成的随机密码 G4Utcerc1n=y  [root@localhost mysql]# cd data/  [root@localhost data]# ls  auto.cnf ibdata1 ib\_logfile1 performance\_schema  ib\_buffer\_pool ib\_logfile0 mysql sys  cd /zls/server/mysql/  cp support-files/mysql.server /etc/init.d/mysqld  gedit /etc/init.d/mysqld  basedir=/zls/server/mysql/  datadir=/zls/server/mysql/data/  service mysqld start → 启动之后才会生成 mysqld.pid 和 mysql.sock 文件  [root@localhost mysql]# service mysqld start  Starting MySQL.. SUCCESS!  [root@localhost mysql]# cd data/  [root@localhost data]# ls  auto.cnf ib\_logfile0 mysql mysql.sock.lock  ib\_buffer\_pool ib\_logfile1 mysqld.pid performance\_schema  ibdata1 ibtmp1 mysql.sock sys  chkconfig --add mysqld  chkconfig mysqld on  ln -s /zls/server/mysql/data/mysql.sock /tmp/mysql.sock  ln -s /zls/server/mysql/bin/mysql /usr/bin/mysql  mysql -uroot -p  alter user 'root'@'localhost' identified by 'a5s7sh4u'; //修改密码  或者  ./mysqladmin -u root -p password "root" //-u 需要修改密码的用户名,新密码为 abc123  flush privileges;  exit; |

设置环境变量：

|  |
| --- |
| echo 'PATH=/zls/server/mysql/bin:/zls/server/mysql/lib:$PATH' >> /etc/profile  echo 'export PATH' >> /etc/profile  source /etc/profile  echo $PATH |

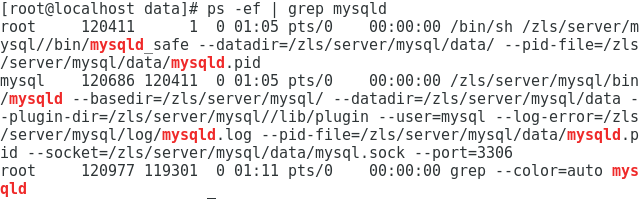
授权远程登陆数据库：

|  |
| --- |
| mysql -uroot -p  #第一个 \* 代表所有数据库,第二 \* 代表所有表,赋予 root 权限 % 代表所有服务器终端,可设为IP地址,密码为 abc123 .  grant all privileges on \*.\* to zlsroot@'%' identified by 'zlS#wJPZz$96%ER';  flush privileges;  exit; |

查看是否运行：

|  |
| --- |
| ps -ef | grep mysqld |

如下图所示：



说明 MySQL 运行正常。把 3306 加入防火墙规则：

|  |
| --- |
| firewall-cmd --zone=public --add-port=3306/tcp --permanent  firewall-cmd --reload |

在客户端用 Navicat 远程连接数据库，看是否能连接上，能连上则说明 zlsroot 账号正常使用。

## 安装 PHP 7.1 软件

下载 PHP 地址：<http://php.net/downloads.php>

下载 PHP 7.1 软件：

|  |
| --- |
| wget http://cn2.php.net/get/php-7.1.23.tar.gz/from/this/mirror  tar -zxvf mirror |

安装 PHP 7.1 依赖库：

|  |
| --- |
| yum -y install gcc gcc-c++ autoconf bison bison-devel zlib-devel libmcrypt libmcrypt-devel mcrypt mhash-devel openssl openssl-devel libxml2 libxml2-devel libcurl libcurl-devel bzip2 bzip2-devel readline readline-devel libedit-devel libjpeg libjpeg-devel libpng libpng-devel freetype freetype-devel gmp gmp-devel libxslt libxslt-devel systemd-devel openjpeg-devel php-mysql php-gd php-imap php-ldap php-odbc php-pear php-xml php-xmlrpc |

创建 www 用户：

|  |
| --- |
| groupadd www  useradd -g www -s /sbin/false -M www  这里创建的用户名和用户组与编译时的  --with-fpm-user 和 --with-fpm-group 的值保持一致 |

安装 PHP 7.1 软件：

|  |
| --- |
| cd php-7.1.23  ./configure --prefix=/zls/server/php/ --with-config-file-path=/zls/server/php/etc/ --with-config-file-scan-dir=/zls/server/php/etc/ --with-apxs2=/zls/server/apache/bin/apxs --enable-inline-optimization --disable-debug --disable-rpath --enable-shared --enable-opcache --enable-fpm --with-fpm-user=www --with-fpm-group=www --with-mysql=mysqlnd --with-mysqli=mysqlnd --with-pdo-mysql=mysqlnd --with-gettext --enable-mbstring --with-iconv --with-mhash --with-mcrypt --with-openssl --enable-bcmath --enable-soap --with-libxml-dir --enable-pcntl --enable-shmop --enable-sysvmsg --enable-sysvsem --enable-sysvshm --enable-sockets --with-curl --with-zlib --enable-zip --with-bz2 --with-readline --without-sqlite3 --without-pdo-sqlite --with-pear --with-iconv-dir --with-freetype-dir --with-png-dir --with-jpeg-dir --enable-gd-native-ttf --with-gd --with-curlwrappers --with-xmlrpc --enable-xml --enable-mbregex --with-sqlite3 --with-pdo-sqlite --enable-session  如果编译时出现 error: mcrypt.h not found. Please reinstall libmcrypt. 错误，则执行如下两个步骤安装  yum install -y epel-release  yum install -y libmcrypt-devel  然后再执行上面的编译  make && make install |

添加环境变量：

|  |
| --- |
| gedit /etc/profile  export PATH=$PATH:/zls/server/php/bin  source /etc/profile  echo $PATH |

相关配置：

|  |
| --- |
| cd /root/Downloads/php-7.1.23  cp php.ini-production /zls/server/php/etc/php.ini  cp /zls/server/php/etc/php-fpm.conf.default /zls/server/php/etc/php-fpm.conf  cp /zls/server/php/etc/php-fpm.d/www.conf.default /zls/server/php/etc/php-fpm.d/www.conf  cd /root/Downloads/php-7.1.23/sapi/fpm  cp init.d.php-fpm /etc/init.d/php-fpm  chmod +x /etc/init.d/php-fpm  chkconfig --add php-fpm  chkconfig php-fpm on  service php-fpm start  Starting php-fpm done  ps -aux|grep php  service php-fpm restart |

配置 php-fpm.conf 文件：

|  |
| --- |
| gedit /zls/server/php/etc/php-fpm.conf  '配置错误日志  error\_log = /zls/server/php/var/log/php-fpm.log  '配置pid文件  pid = /zls/server/php/var/run/php-fpm.pid |

配置 php.ini 文件：

|  |
| --- |
| gedit /zls/server/php/etc/php.ini #暂无 |

配置 www.conf 文件：

|  |
| --- |
| gedit /zls/server/php/etc/php-fpm.d/www.conf #暂无 |

Apache 关联 PHP：

|  |
| --- |
| gedit /zls/server/apache/conf/httpd.conf  添加以下内容  LoadModule php7\_module modules/libphp7.so #这行可能在编译安装 php 的过程中已经自动添加了  #LoadModule rewrite\_module modules/mod\_rewrite.so # 去掉 # 号  在文件最后添加如下三行代码:  <FilesMatch \.php$>  SetHandler application/x-httpd-php  </FilesMatch>  添加 index.php  <IfModule dir\_module>  DirectoryIndex index.php index.html  </IfModule> |

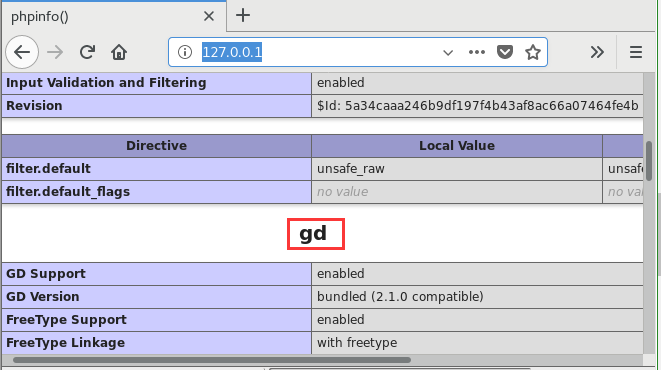
测试 Apache 是否与 PHP 关联成功：

|  |
| --- |
| gedit /zls/server/apache/htdocs/phpzls.php  #添加以下内容  <?php  phpinfo();  ?> |

重启 Apache PHP 服务：

|  |
| --- |
| service httpd restart  service php-fpm restart |

在浏览器输入 <http://127.0.0.1/> 显示如下图所示：

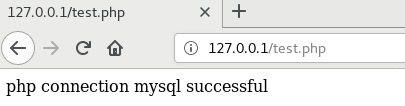


说明 Apache + PHP 整合成功，同时，GD 库也是安装成功的。

测试 PHP 与 MySQL 是否关联成功：

|  |
| --- |
| gedit /zls/server/apache/htdocs/mysqlzls.php  <?php  $mysqli = new mysqli("127.0.0.1", "root", "root");  if (!$mysqli) {  echo "database connection error";  } else {  echo "php connection mysql successful";  }  $mysqli->close();  ?> |

结果如下图所示：



说明 PHP + MySQL 整合成功。

如果测试失败，可以尝试 httpd、php-fpm 多重启几次。

## 建设 DEDECMS 站点

这里以DEDECMS程序作为建站测试。

|  |
| --- |
| cd /zls/server/apache/htdocs/  mkdir www.zlsdede.com  cp -r uploads/\* /zls/server/apache/htdocs/www.zlsdede.com/  gedit /etc/hosts  127.0.0.1 www.zlsdede.com  gedit /zls/server/apache/conf/httpd.conf  #Include conf/extra/httpd-vhosts.conf 去掉 #  #LoadModule rewrite\_module modules/mod\_rewrite.so 去掉 #  gedit /zls/server/apache/conf/extra/httpd-vhosts.conf  <VirtualHost \*:80>  DocumentRoot "/zls/server/apache/htdocs/www.zlsdede.com"  ServerName www.zlsdede.com  <Directory "/zls/server/apache/htdocs/www.zlsdede.com">  Options Indexes FollowSymLinks  AllowOverride all  Order allow,deny  Allow from all  </Directory>  </VirtualHost>  chmod -R 777 /zls/  service httpd restart |

然后在浏览器输入域名即可进行建站。

## 缺少 GD 图形处理库怎么办？

进入 PHP 安装包的 GD 目录：

|  |
| --- |
| cd /root/Downloads/php-7.1.23/ext/gd |

执行安装包下的 /bin/phpize：

|  |
| --- |
| /zls/server/php/bin/phpize  ./configure --with-php-config=/zls/server/php/bin/php-config |

然后进行编译和安装：

|  |
| --- |
| make  make install  systemctl restart php-fpm |

后续一些查看和操作：

|  |
| --- |
| find / -name gd.so  /zls/server/php/lib/php/extensions/no-debug-non-zts-20160303/gd.so  gedit /zls/server/php/etc/php.ini  [gd]  extension=gd.so  systemctl restart php-fpm  service httpd restart |

差不多就这些。

## CentOS 7.5 中文界面修改为英文界面

这是安装了图形桌面和远程连接桌面设置后进行的操作。

|  |
| --- |
| gedit /etc/locale.conf |

把

|  |
| --- |
| LANG="zh\_CN.UTF-8" |

修改成

|  |
| --- |
| LANG="en\_US.UTF-8" |

或直接注释掉。

|  |
| --- |
| source /etc/locale.conf |

然后重启操作系统。

## 使用 phpize 安装 php 模块

使用 phpize 安装 php 模块

我们在安装 php 的时候，配置了一些安装模块，但是随着工作的需要，我们还要再安装一些 php 模块，但是又不想重新编译 php，这个时候，我们就可以使用 phpize 来安装 php 模块了。

参考地址:

<https://www.cnblogs.com/saneri/p/5191096.html>

## 安装 Python 3.6.6 版本

直接贴上操作流程：

|  |
| --- |
| wget https://www.python.org/ftp/python/3.7.2/Python-3.7.2.tgz  tar -zxvf Python-3.7.2.tgz  mkdir -p /zls/python372  chmod -R 777 /zls/  cd Python-3.7.2/  ./configure --prefix=/zls/python372/  yum groupinstall 'Development Tools'  yum install zlib-devel bzip2-devel openssl-devel ncurses-devel  yum install zlib\*  gedit Modules/Setup.dist  #zlib zlibmodule.c -I$(prefix)/include -L$(exec\_prefix)/lib –lz  把该行的注释 # 去掉，并保存。  #SSL=/usr/local/ssl  #\_ssl \_ssl.c \  # -DUSE\_SSL -I$(SSL)/include -I$(SSL)/include/openssl \  # -L$(SSL)/lib -lssl -lcrypto  把这四行的注释符 # 去掉  make && make install  #如果出现 ModuleNotFoundError: No module named '\_ctypes' 则，执行 yum install libffi-devel -y 然后再重新执行编译和安装  ln -s /zls/python372/bin/python3 /usr/bin/python3  ln -s /zls/python372/bin/pip3 /usr/bin/pip3  pip3 install --upgrade pip  安装虚拟环境:  pip3 install -i https://pypi.doubanio.com/simple/ virtualenv  pip3 install -i https://pypi.doubanio.com/simple/ virtualenvwrapper  gedit ~/.bashrc  export VIRTUALENVWRAPPER\_PYTHON=/zls/python372/bin/python3  export WORKON\_HOME=/zls/python372env  source /zls/python372/bin/virtualenvwrapper.sh  source ~/.bashrc  #######################################################  # 出现这种错误，是因为 virtualenv 这个基础依赖包被安装在默认 Python 目录下，就需要做一个软连接 ln -s。  # ERROR: virtualenvwrapper could not find virtualenv in your path  # 首先还是用 find / -name "virtualenv" 去找到位置  # find / -name "virtualenv"  # /usr/local/python3/bin/virtualenv  # 最后进行软连接  # ln -s /usr/local/python3/bin/virtualenv /usr/bin/virtualenv  #######################################################  find / -name "virtualenv"  ln -s /zls/python372/bin/virtualenv /usr/bin/virtualenv  mkvirtualenv py3django  workon py3django  deactivate  rmvirtualenv py3django  lsvirtualenv  cdvirtualenv  pip install django |

参考网址：

<https://blog.csdn.net/a987929281/article/details/81636838>

<https://www.cnblogs.com/octo/p/centos7-install-python3.html>

<https://blog.csdn.net/m0_37714245/article/details/82086250>

<https://www.cnblogs.com/mqxs/p/9103031.html>

### Python 安装 Tornado 库

过程很简单：

|  |
| --- |
| mkvirtualenv py3tornado  pip install tornado |

Tornado 简易教程：

<https://blog.csdn.net/belalds/article/details/80575755>

**注意**：

1、Tornado应该运行在类Unix平台，在线上部署时为了最佳的性能和扩展性，仅推荐Linux和BSD（因为充分利用Linux的epoll工具和BSD的kqueue工具，是Tornado不依靠多进程/多线程而达到高性能的原因）。

2、对于Mac OS X，虽然也是衍生自BSD并且支持kqueue，但是其网络性能通常不太给力，因此仅推荐用于开发。

3、对于Windows，Tornado官方没有提供配置支持，但是也可以运行起来，不过仅推荐在开发中使用。

**整合**：

tornado + django 开发网站

tornado 整合 django Windows

<https://blog.csdn.net/sinat_29957455/article/details/78759894>

## 安装 PyCharm 软件

其实就是下载和解压缩，并不需要安装过程：

|  |
| --- |
| wget https://download.jetbrains.8686c.com/python/pycharm-professional-2018.2.4.tar.gz  tar -zxvf pycharm-professional-2018.2.4.tar.gz  mv pycharm-2018.2.4 pycharm  mv pycharm /zls/  cd /zls/pycharm/bin  ./pycharm.sh |

然后修改 hosts 文件：

|  |
| --- |
| gedit /etc/hosts  0.0.0.0 account.jetbrains.com |

添加快捷方式：

|  |
| --- |
| ln -s /zls/pycharm/bin/pycharm.sh /usr/bin/pycharm |

到这里就结束了。

给 pycharm.sh 加入应用程序列表，并创建快捷方式：

|  |
| --- |
| gedit /usr/share/applications/pycharm.desktop |

加入如下代码：

|  |
| --- |
| #!/usr/bin/env xdg-open  [Desktop Entry]  Encoding=UTF-8  Name=pycharm  Comment=pycharm-2018.3.4  Exec=/zls/pycharm/bin/pycharm.sh  Icon=/zls/pycharm/bin/pycharm.png  Terminal=false  StartupNotify=true  Type=Application  Categories=Application; |

然后执行

|  |
| --- |
| chmod a+x /usr/share/applications/pycharm.desktop |

这样就完成了。

## 安装 TeamViewer 软件

下载 TeamViewer：

|  |
| --- |
| wget https://download.teamviewer.com/download/linux/teamviewer.x86\_64.rpm |

安装 TeamViewer：

|  |
| --- |
| yum install teamviewer\_13.2.26559.x86\_64.rpm |

出现如下错误：

|  |
| --- |
| Error: Package: teamviewer-13.2.26559-0.x86\_64 (/teamviewer\_13.2.26559.x86\_64)  Requires: libQt5WebKit.so.5()(64bit) >= 5.5  Error: Package: teamviewer-13.2.26559-0.x86\_64 (/teamviewer\_13.2.26559.x86\_64)  Requires: libQt5WebKitWidgets.so.5()(64bit) >= 5.5 |

打开 https://pkgs.org/download/libQt5WebKitWidgets.so.5()(64bit) 网址，找到 CentOS 7 下面的链接并打开，如下链接：https://centos.pkgs.org/7/epel-x86\_64/qt5-qtwebkit-5.9.1-1.el7.x86\_64.rpm.html 在该页面上找到 Install Howto 这里就是安装步骤： 打开 https://fedoraproject.org/wiki/EPEL 页面，找到对应版本的 EPEL 安装方法：

|  |
| --- |
| yum install https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm  yum install qt5-qtwebkit  yum install teamviewer\_13.2.26559.x86\_64.rpm |

图形界面启动 TeamViewer 软件：

|  |
| --- |
| Application → Internet → TeamViewer 13 #单击该快捷方式即可启动，不需要通过命令行启动。 |

以下内容不需要操作：

查找 TeamViewer 快捷方式：

|  |
| --- |
| [root@localhost ~]# find / -name teamviewer  /opt/teamviewer  /opt/teamviewer/tv\_bin/script/teamviewer  /etc/teamviewer  /usr/bin/teamviewer |

快捷方式存在于 /usr/bin/ 目录下，可直接输入 teamviewer 启动软件

|  |
| --- |
| [root@localhost ~]# teamviewer |

把 TeamViewer 加入开机启动：

|  |
| --- |
| cp /usr/bin/teamviewer /etc/rc.d/init.d/teamviewer  chmod +x /etc/rc.d/init.d/teamviewer  gedit /etc/rc.d/init.d/teamviewer |

在 #!/bin/bash 语句下添加如下两条语句，包括 # 部分

|  |
| --- |
| #chkconfig:345 61 61  #description:TeamViewer teamviewer |

然后添加系统服务并开机启动：

|  |
| --- |
| chkconfig --add teamviewer  chkconfig teamviewer on |

开机启动好像没有起作用。

# Apache相关

## Apache性能优化

Apache优化：修改最大并发连接数

<https://www.cnblogs.com/jukan/p/5830068.html>

处理高并发、高访问之Apache优化

<https://blog.csdn.net/tai532439904/article/details/78484342>

Apache 性能配置优化

<https://www.cnblogs.com/nongchaoer/p/6792767.html>