4.

一键搭建ss脚本

git clone https://github.com/flyzy2005/ss-fly

ss-fly/ss-fly.sh -i flyzy2005.com 1024

或

1. yum -y install wget

2. wget -O /root/shadowsocks.sh https://raw.githubusercontent.com/teddysun/shadowsocks\_install/master/shadowsocks.sh

3. sh /root/shadowsocks.sh install



重启阿帕奇服务器：sudo /etc/init.d/apache2 restart

vi /etc/apache2/apache2.conf

vi /etc/apache2/sites-enabled/000-default.conf

sudo cp /root/exempt-front /var/www/html

**创建sql用户**

CREATE USER 'debian-sys-maint'@'localhost' IDENTIFIED BY 'su4B1V2efWUxbRSm';

GRANT ALL ON \*.\* TO 'debian-sys-maint'@'localhost';

flush privileges;

**github推新项目：**

git init

git add .

git commit -m “first-commit”

git remote add origin [git@github.com:YourName/YourRepositroy.git](mailto:git@github.com:YourName/YourRepositroy.git)

git push -u origin master(不行就git push origin master --force)

**git公私钥：**

ssh-keygen -t rsa -C [791150351@qq.com](mailto:791150351@qq.com) （接着按三个回车）

cat ~/.ssh/id\_rsa.pub

复制，打开github ，点自己头像 >> settings >> SSH and GPG keys >>New SSH key

**Mysql远程访问：**

use mysql;

update user set host = '180.102.108.30' where user = 'lhd791150351';

select host, user from user;

FLUSH PRIVILEGES;

grant all privileges on \*.\* to ' lhd791150351'@'180.102.108.30'identified by 'LHDqzmp1234.' with grant option;

FLUSH PRIVILEGES;

**服务器前台运行jar**

Java -jar XXX.jar –server.port=1024

后台运行jar

nohup java -jar XXX.jar –server.port=1024

**查看端口号：**

netstat -tln

**linux如何查看端口被哪个进程占用？**

1、lsof -i:端口号

2、netstat -tunlp|grep 端口号

都可以查看指定端口被哪个进程占用的情况

在centos下, **无法使用命令lsof,** 出现以下信息:

# lsof -i:3690

-bash: lsof: command not found

2. 解决方法

我们可以通过yum来安装:

# yum install lsof

**移动文件夹**

mv /usr/sp/git-project/web/dist /usr/sp/apache-tomcat-8.5.29/webapps

**递归拷贝文件夹**

cp -r /usr/sp/sp/web/dist /usr/sp/apache-tomcat-8.5.29/webapps

**centOS7环境下安装jdk1.8**

首先下载jdk1.8 去官网下载jdk：http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html

解压到指定的目录 解压命令

若后缀为.tar.gz，使用tar -xzvf jdk-8u201-linux-x64.tar.gz //安装到当前目录下

tar -zxvf jdk-8u201-linux-x64.tar.gz -C /usr/local/java/ //安装到指定目录下

然后修改配置文件 就相当与设置环境变量

编辑/etc/下的profile文件，配置环境变量

[root@bogon software]# vim /etc/profile

export JAVA\_HOME=/usr/local/java/jdk1.8.0\_101

export JRE\_HOME=${JAVA\_HOME}/jre

export CLASSPATH=.:${JAVA\_HOME}/lib:${JRE\_HOME}/lib

export PATH=${JAVA\_HOME}/bin:$PATH

使/etc/profile生效

[root@bogon jdk1.8.0\_101]# source /etc/profile

**CentOS7安装MySQL（完整版）**

2018年05月31日 16:36:07 Antelope24 阅读数 20051

版权声明：本文为博主原创文章，遵循 CC 4.0 by-sa 版权协议，转载请附上原文出处链接和本声明。

本文链接：https://blog.csdn.net/qq\_36582604/article/details/80526287

在CentOS中默认安装有MariaDB，这个是MySQL的分支，但为了需要，还是要在系统中安装MySQL，而且安装完成之后可以直接覆盖掉MariaDB。

1 下载并安装MySQL官方的 Yum Repository

[root@localhost ~]# wget -i -c http://dev.mysql.com/get/mysql57-community-release-el7-10.noarch.rpm

使用上面的命令就直接下载了安装用的Yum Repository，大概25KB的样子，然后就可以直接yum安装了。

[root@localhost ~]# yum -y install mysql57-community-release-el7-10.noarch.rpm

之后就开始安装MySQL服务器。

[root@localhost ~]# yum -y install mysql-community-server

这步可能会花些时间，安装完成后就会覆盖掉之前的mariadb。

至此MySQL就安装完成了，然后是对MySQL的一些设置。

2 MySQL数据库设置

首先启动MySQL

[root@localhost ~]# systemctl start mysqld.service

查看MySQL运行状态，运行状态如图：

[root@localhost ~]# systemctl status mysqld.service

此时MySQL已经开始正常运行，不过要想进入MySQL还得先找出此时root用户的密码，通过如下命令可以在日志文件中找出密码：

[root@localhost ~]# grep "password" /var/log/mysqld.log

如下命令进入数据库：

[root@localhost ~]# mysql -uroot -p

输入初始密码（是上面图片最后面的 no;e!5>>alfg），此时不能做任何事情，因为MySQL默认必须修改密码之后才能操作数据库：

mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY 'new password';

其中‘new password’替换成你要设置的密码，注意:密码设置必须要大小写字母数字和特殊符号（,/';:等）,不然不能配置成功

3 开启mysql的远程访问

执行以下命令开启远程访问限制（注意：下面命令开启的IP是 192.168.0.1，如要开启所有的，用%代替IP）：

grant all privileges on \*.\* to 'root'@'192.168.0.1' identified by 'password' with grant option;

然后再输入下面两行命令

mysql> flush privileges;

mysql> exit

4 为firewalld添加开放端口

添加mysql端口3306和Tomcat端口8080

[root@localhost ~]# firewall-cmd --zone=public --add-port=3306/tcp --permanent

[root@localhost ~]# firewall-cmd --zone=public --add-port=8080/tcp --permanent

然后再重新载入

[root@localhost ~]# firewall-cmd --reload

5 更改mysql的语言

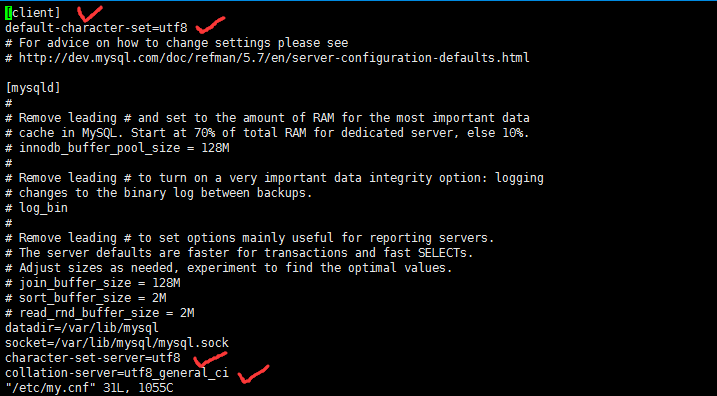
首先重新登录mysql，然后输入status：

可以看到，绿色箭头处不是utf-8

因此我们先退出mysql，然后再到、etc目录下的my.cnf文件下修改一下文件内容

vi /etc/my.cnf

进入文件后，新增四行代码：



character-set-server=utf8

collation-server=utf8\_general\_ci

保存更改后的my.cnf文件后，重启下mysql，

service mysqld restart

然后输入status再次查看，你就会发现变化啦

最后，到Windows下用cmd命令启动mysql啦！！

**Tomcat 部署项目**

下载 Tomcat 服务器

　　①、官网下载地址：http://tomcat.apache.org/

　　②、tomcat 8.0 64位百度云下载地址：http://pan.baidu.com/s/1slbKPsx 密码：ewui

　　③、tomcat 8.0 32位百度云下载地址：http://pan.baidu.com/s/1o8G28rS 密码：k11n

2、启动并部署 Tomcat 服务器

①、解压 tomcat 安装包到一个非中文目录下

解压到指定的目录 解压命令

若后缀为.tar.gz，使用tar -xzvf apache-tomcat-8.5.29.tar.gz //安装到当前目录下

tar -zxvf apache-tomcat-8.5.29.tar.gz -C /usr/local/java/ //安装到指定目录下

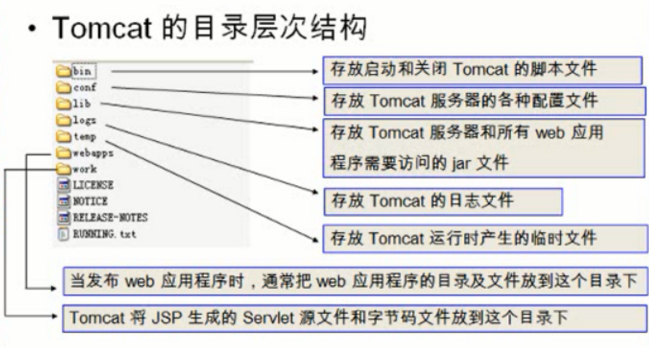
　　②、配置环境变量。JAVA\_HOME(指向 JDK 安装的根目录)

③、双击 apache-tomcat-6.0.16\bin 目录下的 startup.bat，启动服务器(如果一闪而过，那就是没有配置 JAVA\_HOME 的环境变量)

cd apache-tomcat-8.5.29/bin

./startup.sh

④、在浏览器中输入 <http://localhost:8080>



1、将编写并编译好的web项目(注意要是编译好的，如果是 eclipse，可以将项目打成 war 包放入)，放入到 webapps 中

　2、启动tomcat服务器（双击 apache-tomcat-6.0.16\bin 目录下的 startup.bat，启动服务器）

3、在浏览器输入：http://localhost:8080/项目名/访问的文件名

重启（只能先关闭再启动）

cd apache-tomcat-8.5.29/bin

./shutdown.sh

./startup.sh

或者绝对路径

/usr/sp/apache-tomcat-8.5.29/bin/shutdown.sh

/usr/sp/apache-tomcat-8.5.29/bin/startup.sh

**Centos7搭建Git及安装使用**

1、查看是否安装了git

rpm -qa|grep git

若已经安装，需要先卸载。卸载命令如下：

rpm -e --nodeps git 或者 rpm -e git

2、安装Git

yum install git

输入y,并回车

再使用 rpm -qa|grep git 来查看是否已经安装好了Git。

**npm安装(Centos7)——CentOS7常用软件**

yum install gcc gcc-c++

wget https://npm.taobao.org/mirrors/node/v10.14.1/node-v10.14.1-linux-x64.tar.gz

然后执行下面命令

tar -zxvf node-v10.14.1-linux-x64.tar.gz

//重命名

mv node-v8.1.4-linux-x64 node

配置环境变量

vim /etc/profile

在文件中添加，目录写自己的实际目录

#set for nodejs

export NODE\_HOME=/usr/local/node

export PATH=$NODE\_HOME/bin:$PATH

生效配置文件

source /etc/profile

node -v

npm -v

**npm install遇到权限不足情况Error: EACCES: permission denied, mkdir '/usr/sp/apache-tomcat-8.5.29/webapps/sp/web/node\_modules/node-sass/.node-gyp'**

**解决：**

npm install --unsafe-perm

**SEVERE: Could not contact localhost:8005. Tomcat may not be running error while shutting down tomcat**

2018年11月03日 16:47:15 兔子不爱吃草 阅读数 2041

起因：大概是因为tomcat启动时server.xml配置有问题导致没启动成功，然后就shutdown不了tomcat了，网上找了许多结束进程的但都没有用，重启了阿里云的服务器也没有用。

（抱歉，报错详情丢失，所以便不上传了）

以下为解决过程tomcat环境是部署在阿里云centos7上，提供命令参考

此处为谷歌翻译转载 原地址

解决：大概是jdk的一个bug，我用的是jdk1.8

过程例如，我的jdk路径是“/usr/java/jdk1.8.0\_144”。（这里如果你和我一样contos7系统，yum安装tomcat不知道路径问题查看yum安装路径

1、cd /usr/java/jdk1.8.0\_144/jre/lib/security/

2、找到名为“java.security”的文件，

3、打开进入：vi java.security

4、找到“securerandom.source = file：/ dev / random”。

5、修改“securerandom.source = file：/ dev /./ urandom”。

6、然后，转到Tomcat / bin目录，执行./start.sh并./shutdown.sh执行成功。

**Tomcat在url根路径访问项目**

在conf/server.xml配置文件

<Host name="localhost" debug="0" appBase="d:/wwwroot"

unpackWARs="true" autoDeploy="true">

<Context path="" docBase="d:/wwwroot" debug="0"

reloadable="true" crossContext="true"/>

</Host>

配置二级域名、https证书

<https://blog.erestu.top/index.php/archives/191.html>

我们经常需要在服务器还是那个配置域名，  
所以出一个简略版教程。

* 首先在阿里云域名解析添加一个A记录指向应用服务器
* 安装Nginx，可以百度
* cd /etc/nginx/conf.d
* vim your.sub.domain.conf输入如下，端口号记得改

server{

listen 80;

server\_name your.sub.domain;

location / {

proxy\_set\_header Host $host:$server\_port;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto $scheme;

proxy\_pass http://127.0.0.1:3000;

root /the/directory/of/your/project;# 如果只有端口，不需要文件路径（如go应用）则可注释掉

index index.html;

# try\_files $uri $uri/ /index.html;

}

}

* service nginx restart
* service nginx reload  
  访问http://your.sub.domain,大功告成

<https://blog.erestu.top/index.php/archives/175.html>

**问题描述**

项目执行在3000端口，服务器部署在阿里云，所以需要先在阿里云控制台添加安全组规则，入方向添加3000端口，由于需要https，所以同时需要开放443端口。

**域名设置**

域名配置可百度。

**centos设置**

**关闭防火墙**

systemctl stop firewalld.service

**关闭Selinux**

vi /etc/selinux/config,将将SELINUX=enforcing改为SELINUX=disabled.

**nginx配置**

server{

listen 443;

server\_name your.sub.domain;

location / {

proxy\_set\_header Host $host:$server\_port;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto $scheme;

proxy\_pass http://127.0.0.1:3000;

root /the/directory/of/your/project;# 如果只有端口，不需要文件路径（如go应用）则可注释掉

index index.html;

# try\_files $uri $uri/ /index.html;

}

}

执行certbot certonly --nginx使用let's encypt生成证书，会提示cert与key存放在那个目录之下，  
写到配置文件中，即

server{

listen 443 ssl;

server\_name jianshi.erestu.top;

ssl\_certificate /etc/letsencrypt/live/jianshi.erestu.top/fullchain.pem;#cert文件路径

ssl\_certificate\_key /etc/letsencrypt/live/jianshi.erestu.top/privkey.pem;#key文件路径

ssl\_session\_timeout 5m;

ssl\_ciphers ECDHE-RSA-AES128-GCM-SHA256:ECDHE:ECDH:AES:HIGH:!NULL:!aNULL:!MD5:!ADH:!RC4;

ssl\_protocols TLSv1 TLSv1.1 TLSv1.2;

ssl\_prefer\_server\_ciphers on;

location / {

proxy\_set\_header Host $host:$server\_port;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto $scheme;

proxy\_pass http://127.0.0.1:3000;

root /the/directory/of/your/project;# 如果只有端口，不需要文件路径（如go应用）则可注释掉

index index.html;

# try\_files $uri $uri/ /index.html;

}

}

最后  
reload nginx:service nginx reload  
并  
service nginx restart,  
访问https://your.sub.domain。

**遇到的问题**

ImportError: No module named 'requests.packages.urllib3‘  
2018年05月24日 11:20:57 shengerjianku 阅读数 7797  
I use nginx on centos 7.3.

I installed certbot following https://certbot.eff.org/#centosrhel7-nginx

There is an error when running certbot --nginx:

[root@demo src]# certbot --nginx  
Traceback (most recent call last):  
File "/usr/bin/certbot", line 9, in   
load\_entry\_point('certbot==0.14.1', 'console\_scripts', 'certbot')()  
File "/usr/lib/python2.7/site-packages/pkg\_resources/**init**.py", line 564, in load\_entry\_point  
return get\_distribution(dist).load\_entry\_point(group, name)  
File "/usr/lib/python2.7/site-packages/pkg\_resources/**init**.py", line 2662, in load\_entry\_point  
return ep.load()  
File "/usr/lib/python2.7/site-packages/pkg\_resources/**init**.py", line 2316, in load  
return self.resolve()  
File "/usr/lib/python2.7/site-packages/pkg\_resources/**init**.py", line 2322, in resolve  
module = **import**(self.module\_name, fromlist=['**name**'], level=0)  
File "/usr/lib/python2.7/site-packages/certbot/main.py", line 14, in   
from certbot import account  
File "/usr/lib/python2.7/site-packages/certbot/account.py", line 17, in   
from acme import messages  
File "/usr/lib/python2.7/site-packages/acme/messages.py", line 4, in   
from acme import challenges  
File "/usr/lib/python2.7/site-packages/acme/challenges.py", line 10, in   
import requests  
File "/usr/lib/python2.7/site-packages/requests/**init**.py", line 58, in   
from . import utils  
File "/usr/lib/python2.7/site-packages/requests/utils.py", line 32, in   
from .exceptions import InvalidURL  
File "/usr/lib/python2.7/site-packages/requests/exceptions.py", line 10, in   
from .packages.urllib3.exceptions import HTTPError as BaseHTTPError  
File "/usr/lib/python2.7/site-packages/requests/packages/**init**.py", line 95, in load\_module  
raise ImportError("No module named '%s'" % (name,))  
ImportError: No module named 'requests.packages.urllib3'  
解决方法：

trypip install --upgrade --force-reinstall 'requests==2.6.0' urllib3 , I've had the same problem https://niuhp.com/other/https-certbot.html

**Cannot download "https://github.com/sass/node-sass/releases/download/v3.8.0/win32-x64-46\_binding.node":**

这句话大家都懂的哈，我就不说了，这里我也不知道为什么会下载失败，因为我的电脑全局代理的，正常的访问任何网站都是无阻碍的，百思不得解，

好吧那我就自己下载吧（为什么自己下载？因为我在google的时候也是发现了写有用的东西，虽然没那么直观哈，但是也给了我很多帮助）

下载完了之后把它注册成全局环境变量，两种方式：

第一种：直接右键我的电脑--》属性--》高级系统设置--》环境变量--》添加

第二种：set XXX=文件路经--》set SASS\_BINARY\_PATH=F:\lishiming\tools\node-sass\win32-x64-46\_binding.node

查看环境是否合适：echo %SASS\_BINARY\_PATH%

如果打印出来您配置好的文件地址那就ok了，

最后再来试试安装：npm i -g node-sass

参照上面的答案完美解决

# 将前端项目发布到nginx

1、安装好nginx后启动

2、将前端项目复制到/nginx/html文件夹下

这里写图片描述

3、将nginx下的配置文件（/nginx/conf/nginx.conf）里的server节点配置改为要配置的域名

server {

#监听的端口，80端口是默认端口，在访问时，就无需输入端口号，其他的都需要输入端口号，比如这里访问地址就是127.0.0.1，而若是8080端口，则是127.0.0.1：8080

listen 80;

#此处localhost可改为要访问的域名或者ip地址，若有多个用空格隔开。例如 server\_name www.baidu.com baidu.com test.baidu.com

server\_name localhost;

#编码

charset utf-8;

#access\_log logs/host.access.log main;

location / {

#nginx下HTML文件夹，访问上述域名时会检索此文件夹下的文件进行访问

root html/BootStrapDemo;

#输入网址（server\_name：port）后，默认的访问页面

index index.html index.htm;

}

4、配置改完后测试配置是否正确（找到sbin文件夹打开，看到nginx文件后再命令行输入nginx -t检测）

5、配置正确后，重启nginx（nginx -s reload）

6、若访问的是域名还需去进行域名解析，网站配置域名指向ip。检测是否成功：ping一下域名，若结果为自己指向的ip则解析成功

1. 测试访问

# Node版本与npm、NODE\_MODULE\_VERSION对应关系（非常重要的环境配置）

<https://nodejs.org/zh-cn/download/releases/>

# 将vscode添加到注册表：

执行下面的文件（前提是vscode路径和文件中的一致）



Windows Registry Editor Version 5.00

[HKEY\_CLASSES\_ROOT\\*\shell\VSCode]

@="Open with Code"

"Icon"="C:\\Program Files\\Microsoft VS Code\\Code.exe"

[HKEY\_CLASSES\_ROOT\\*\shell\VSCode\command]

@="\"C:\\Program Files\\Microsoft VS Code\\Code.exe\" \"%1\""

Windows Registry Editor Version 5.00

[HKEY\_CLASSES\_ROOT\Directory\shell\VSCode]

@="Open with Code"

"Icon"="C:\\Program Files\\Microsoft VS Code\\Code.exe"

[HKEY\_CLASSES\_ROOT\Directory\shell\VSCode\command]

@="\"C:\\Program Files\\Microsoft VS Code\\Code.exe\" \"%V\""

Windows Registry Editor Version 5.00

[HKEY\_CLASSES\_ROOT\Directory\Background\shell\VSCode]

@="Open with Code"

"Icon"="C:\\Program Files\\Microsoft VS Code\\Code.exe"

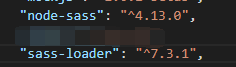
[HKEY\_CLASSES\_ROOT\Directory\Background\shell\VSCode\command]

@="\"C:\\Program Files\\Microsoft VS Code\\Code.exe\" \"%V\""

# [解决npm安装node-sass太慢及编译错误问题](https://www.cnblogs.com/charleswong/p/11439245.html)

<https://www.cnblogs.com/charleswong/p/11439245.html>

版本：



npm install -g mirror-config-china --registry=http://registry.npm.taobao.org

npm install sass-loader node-sass -D

npm install

npm run dev

