**linux 下篇**

**Linux自有服务（2）**

一，设置主机名

回顾：#hostname

#hostname -f FQDN(全限定域名)

1，设置临时主机名，需要切换用户使之生效

#hostname 设置的主机名

2，永久设置主机名，需要重启

先找到一个文件

/etc/sysconfig/network [主机名的配置文件]

修改其中的HOSTNAME 来设置永久主机名

修改linux服务器的hosts文件，将 修改的名字 指向本地(设置FQDN)

hosts文件的位置：/etc/hosts

vim /etc/hosts

然后修改的名字加到 第一行(本机的地址，经过DNS的解析)的最后

问题：

不设置 FQDN会怎么样？

1，很多开源服务器软件（例如apache）则无法启动，或出现错误

2，影响本地域名的解析

二，chkconfig 服务配置

作用：相当于window下“安全卫士”，“电脑管家”之类的安全辅助工具，提供“开机启动项”的一个管理服务。

在linux下该服务不一定自启

1，开机启动服务查询 # chkconfig -list

其中 0-6 表示 各个启动级别

例如：以httpd为例，其3级别为关闭（off),则表示其在3级别下默认开机不启动

2，删除服务

# chkconfig --del 服务名

例如删除 httpd ： #chkconfig --del httpd

3,添加开机启动服务，把服务加到开机项里，对其具体的开启关闭，还需手动设置

#chkconfig --add 服务名

查询：#chkconfig --list|grep httpd

4,设置服务在某个级别下开机启动/不启动

#chkconfig --level 35(连在一起的启动级别) 服务名 on/off

案例：

设置httpd在 35 级别下 设置开启启动

#chkconfig --level 35 httpd on

三，ntp 服务

作用： 主要用于对计算机的时间同步管理操作

获取当前linux 时间：#date + "%F %T"

同步服务器时间方式有两个：一次性同步(手动同步) 通过服务自动同步

时间服务器上游的概念

1，一次性同步时间（简单）

# ntpdate 时间服务域名或ip地址

ip地址查看可以访问 http:[www.ntp.org.cn/pool.php](http://www.ntp.org.cn/pool.php)

2,设置时间同步的服务

服务名：ntpd

启动 ntpd服务

#service ntpd start 或者 /etc/init.d/ntpd start

设置ntpd 开机启动

#chkconfig --list|grep ntpd

#chkconfig --level 35 ntpd on

四，防火墙服务

防火墙作用： 防范网络攻击，选择性让请求通过保证网络安全

（防水墙作用：bbs论坛，灌水，水军）

防火墙两种：

软件防火墙

硬件防火墙

centos6.5 中防火墙的名称：iptables

centos7.x中防火墙的名称：firewalld

1，查看iptables服务是否开机启动

#chkconfig --list|grep iptables

2，查看是否有防火墙服务进程

#ps -ef|grep iptables

3,iptables服务启动/重启/关闭

#service iptables start/restart/stop

/etc/init.d/iptables start/restart/stop

4，查看防火墙的状态（规则）

#service iptables status

查看规则

#iptables -L -n

含义：

-L：表示列出规则

-n：表示将单词表达形式改成数字显示

5，简单设置防火墙规则

例如：需要允许80端口通过防火墙， 则规则可以用下面的命令来设置

#iptables -A INPUT -p tcp --dport 80 -j ACCEPT #允许访问80端口

iptables：主命令

-A：add，添加规则

INPUT:进站请求（出战请求 output）

-p：protocol，指定协议（icmp/tcp/udp）

--dport：指定端口号

-j:指定行为结果，允许（accept）/禁止（reject）

添加完成后保存操作

/etc/init.d/iptables save

五，rpm管理

作用： 对linux服务器上的软件包进行管理操作(查询，卸载，安装)

1，查询某个软件的安装情况

#rpm -qa|grep 关键词

-q：query 查询

-a：全部 all

实例：查询是否安装firefox

#rpm -qa|grep firefox

2，卸载某个软件

#rpm -e 软件的名称

实例：卸载firefox

#rpm -e firefox

没有依赖关系可直接卸载，有依赖关系的情况下，不允许直接卸载

强制卸载：rmp -e httpd --nodeps

3，软件的安装

装软件需要先找到安装包

安装包的获取方式：

1，官网下载

2，不介意老版本的话，可以从光盘(或者镜像文件)中读取

此处以光盘文件为例：

查看块状设备的信息：

#lsblk

Name:名称 Size:设备大小 Type:类型 MountPoint：挂载点（类似windows下盘符）

扩展：光盘的挂载和解挂

a,解挂操作：相当于u盘在windows上已经被弹出，但是还没有拔下来

命令：umount

语法：#umount 当前设备的挂载点（路径）

b,挂载光盘

命令：mount

语法：#mount 设备原始地址 要挂载的位置路径

设备原始地址：地址统一都在/dev下，然后根据大小确定name值，拼凑组成

原始地址，如：“/dev/sr0"

要挂载的位置路径：挂载目录一般都在mnt下，可以在mnt下新建目录

安装软件的命令：

#rpm -ivh 软件包完整安装名称

选项：

-i :install 安装

-v：显示进度条

-h：表示以 # 形式显示进度条

查找安装包完整安装名称

#rpm -qa|grep firefox

六，cron/crontab 计划任务

作用： 计划任务程序，定时执行软件及服务

语法：#crontab 选项

常用选项：

-l：list，列出指定用户的计划任务列表

-e：edit，编辑指定用户的计划任务列表

-u：user，指定的用户，默认当前用户

-r：remove，删除指定的用户的计划任务列表

1，编辑计划任务：

计划任务的规则语法格式：一行为单位，一行为一个计划

分 时 日 月 周 需要执行的命令：

0 0 \* \* \* reboot 每天的0点0分执行reboot指令

取值范围：

分：0-59

时：0-23

日：1-31

月：1-12

周：0-7 0和7表示星期天

四个符号：

\*：表示取值范围中的每个数字

-：做连续区间表达式的，要想表示1~7，则可写成 1-7

/：表示每多少个，例如：要想每10分钟执行一次，则可以在分的位置写成

\*/10

,：表示多个取值比如想在1点，2点，6点执行，则可以在时的位置写：1,2,6

问题：

1，每月1，10，22日的4:45重启network服务

45 4 1,10,22 \* \* service network restart

2，每天18:00-23:00 之间每隔30分钟重启network服务

\*/30 18-23 \* \* \* service network restart

3，每隔两天的上午8点到11点的第3和第15分钟执行一次重启

3,15 8-11 \*/2 \* \* reboot

例子：

每隔1分钟往root家目录中的RT.txt 中输入当前的时间信息，为了看到效果使

用追加输出

\*/1 \* \* \* \* date+"%F %T">>/root/RT.txt

Cromtab 权限问题：超级管理员可通过配置文件设置某些用户不可执行这个操作

配置文件位于：

/etc/cron.deny 里面写用户名 一行一个

还有一个配置文件：（白名单）

/etc/cron.allow

注意：白名单的优先级较高

**Linux权限管理操作**

一，权限概述

1，权限定义

Linux系统一般将文件可存/取访问的身份分为3个类别：owner，group，others，且

3种身份各有read,wirte,excute等权限

读，写，执行对于文件夹，和文件有不同的定义

2，身份介绍

Owner身份：文件所有者

Group身份：与文件所有者同组的用户

Others身份：其他人

Root用户：超级用户

二，Linux的权限介绍

要设置属性，就要知道文件的属性和权限的分配规则，在linux中，ls命令常用来查看文件的属性用于显示文件的文件名和相关属性

#ls -l 路径 【ls -l 等价于 ll】

文档的权限属性

r:read 可读

w:write 可写

e:execute 可执行文件

-：没有对饮权限

**drwxr-x---**

第一位：表示文档类型，常见取值有“d表示文件夹，“-表示文件”,"I表示连接”，“s表示套接字等等”

第2-4位：表示文档所有者的权限情况，取值有“r",-；第3位表示写权限的情况，w表示可写，-表示不可写，第4位表示执行权限的情况，取值有x,-.

第5-7位：表示与所有者同在一个组的用户的权限情况，

第8-10位：表示处理上面的前2部分的用户之外的其他用户的权限情况

三，Linux的权限设置

语法：#chmod 选项 权限模式 文档

常用选项：

-R：地柜设置权限（当文档类型为文件夹的时候）

权限模式：就是该文档需要设置的权限信息

文档：可以是文件或文件夹，可以是相对路径或绝对路径

注意：只有root用户和文档所有者才能给文档设置权限

1，字母形式

选项 字母 介绍

谁 u 用户

谁 g 所属群体

谁 o 其他人

谁 a 所有人

作用 + 增加权限

作用 - 减少权限

作用 = 确定权限

作用 r 可读

作用 w 可写

作用 x 执行

给谁设置：

u：表示所有者身份 owner,user

g：表示给所有者同组用户设置（group)

o：表示others

a：表示all，

权限字符：

r：读

w：写

x：执行

权限分配方式：

+：表示给具体的用户新增权限

-：表示删除用户的权限（相对当前）

=：表示将权限设置成具体的值（注重结果）

例如：需要给anaconda-ks.cfg文件（-rw--------）设置权限，要求所有者拥有全部的权限，同组用户拥有读和执行权限，其他用户只读权限

1，#chmod u+x,g+rx,o+r anaconda-ks.cfg

2，#chmod u=rwx,g=rx,,o=r anaconda-ks.cfg

2，数字形式

数字形式的权限，用数字代表具体的权限赋予操作

读：r 4

写：w 2

执行：x 1

无权限：0

数值 权限 目录列表

0 不能读，不能写，不能执行 ---

1 不能读，不能写，可执行 --x

2 不能读，可写，不能执行 -w-

3 不能读，可写，可执行 -wx

4 可读，不能写，不能执行 r--

5 可读，不能写，可执行 r-x

6 可读，可写，不能执行 rw-

7 不可，可写，可执行 rwx

例如：给anaconda-ks.cfg设置权限权限要求所有者拥有全部权限，同组用户拥有读执行权限，，其他用户只读

全部权限：u，读+写+执行=4+2+1=7

读和执行：g,读+执行=5+1=5

读o: 读=4

则权限为754

#chmod 754 anaconda-ks.cfg

面试题：用超级管理员设置文档的权限命令是#chmod -R 731 aaa,请问这个命令有什么问题？

拥有者：7=4+2+1=读+写+执行

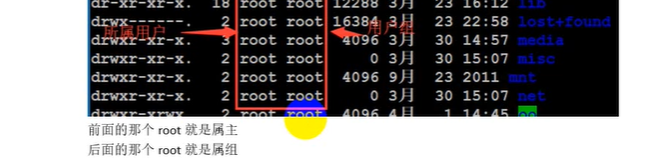
同组用户：3 = 2+1=写+执行------无法打开

其他用户：1=1 =执行

四，属主与属组设置

属主：所属的用户（文件的主人）

属组：所属的用户组



1，chown

作用：更改文档的所属用户

语法：#chown -R username 文档路径

案例：将刚才root用户创建的oo目录，所有者更改为test

# chown test oo/

2，chgrp

作用：更改文档的所属用户组

语法：#chgrp -R groupname 文档的路径

案例：将刚才root用户创建的oo目录，所有者更改为test并且将所属用户组也改为test

#chgrp test oo/

思考，如何通过一个命令实现既可以更改所属的用户，也可以修改所属的用户组呢？

答： #chown -R username:groupname 文档路径

案例：要求只使用chown指令，将oo目录的所属用户和用户组改回成root并且包含其子目录

# chown -R root:root oo/

五，扩展：sudo操作

问题：reboot,shutdown,init,halt,uer管理，在普通用户身份上无法操作，但又需要执行该操作，又不能让root用户把自己的密码告诉普通用户，这个问题该如何解决？

答：可使用sudo(switch uesr do) 命令来进行权限设置sudo可以让管理员（root）用户事先定义某些特殊命令谁可以执行

默认sudo中是没有除root之外的用户的规则，要想使用则先配置sudo

sudo配置文件：/etc/sudoers



b.配置普通用户的权限

clipboard.png

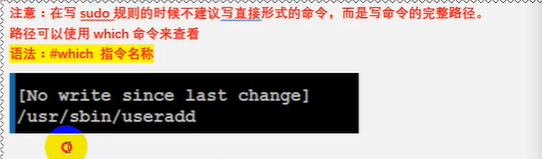
root 表示用户名，如果是用户组，，则可以写成“%组名”

ALL 表示允许登录的主机（地址白名单）

（ALL）：表示以谁的身份执行，ALL表示root身份

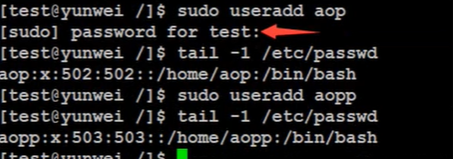
ALL:表示当前用户可以执行的命令，多个命令可以使用“，”分割

案例：本身test用户不能添加用户，要求使用sudo配置，将其设置为可以添加用户，并且可以修改密码（但是不能修改root用户密码）



clipboard.png

sudo需要执行的命令



在输入sudo指令后需要输入当前的用户密码进行确认的操作（表示root用户密码），输入之后在接下来5分钟内再次执行sudo指令不需要密码

**Linux的网络基础**

一，网络相关基础

1，网络发展

网络分类（记忆）

局域网(Local Area Network,LAN)

城域网(Metropolitan Area Network)

广域网(Wide Area Network,WAN)

2,ip地址

ip:internet protocol,网络之间互连的协议，也就是为计算机网络相互连接进行通信而设计的协议

ip地址分为公有地址，私有地址

3，网卡

4，网线

5，交换机

6，路由器

7，拓扑结构图（扩展）

星型拓扑结构

总线型拓扑结构

环形拓扑结构

树型拓扑结构

网状拓扑结构

混合型拓扑结构

蜂窝型拓扑结构

二，网络相关命令

1，ping

作用：检测当前主机与目标主机之间的连通性（不是100%准确，有的服务器是禁ping）

语法：#ping 主机地址（ip地址，主机名，域名等）

#ping [www.baidu.com](http://www.baidu.com)

和windows下的区别：

linux下默认一致发送

windows下默认发送4个数据包

2，netstat

作用：表示查看网络的连接信息

语法：#netstat -tnlp(-t:tcp协议，-n：将字母装化成数字，-l：列出状态为监听,-p:显示进程相关信息)

#netstat -an (-a:表示全部，-n:将字母转化为数字)

TCP/IP协议需要使用这个命令

3，traceroute

作用：查找当前主机与目标主机之间的所有网关（路由器，会给沿途各个路由器发送icmp数据包，路由器可能会不给响应）

该命令不是内置命令，需要安装

语法：#traceroute 主机地址

扩展：windows下的命令是 tracert

比较好用的网站用来进行网络操作：

<http://tool.chinaz.com/>

4，arp

地址解析协议，即ARP（Address Resolution Protocol)，是根据IP地址获取（MAC）物理地址的协议。

常用语法：

#arp -a 查看本地缓存mac表

#arp -d 主机地址 删除指定的缓存记录

5，tcpdump

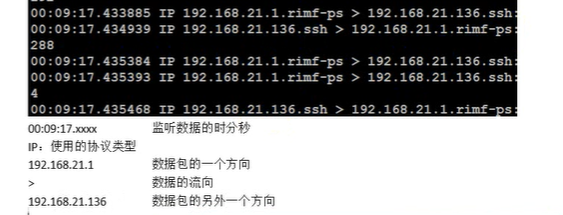
作用：抓包，抓取数据表

语法：

#tcpdump 协议 port 端口

#tcpdump 协议 port端口 host地址

#tcpdump -i 网卡设备名



三，项目上线流程

1，服务器选配购买

2，域名购买

3，域名备案

4，域名解析

5，配置生产环境

6，上传代码

此时需要上传工具：pscp,filezilla

**Shell基础**

一，关于shell

1，什么是shell

shell是一个用c语言编写的程序，是用户使用linux的桥梁，shell既是一种命令语言，也是一种程序设计语言

shell是指一种应用程序，这个应用程序提供了一个界面，用户通过这个界面访问操作系统内核的服务

脚本：javascript，VBScript，ASP，JSP，PHP，SQL，Perl,Shell,python,Ruby,JavaFX,Lua等。

shell种类，linux默认的shell是/bin/bash，流行的shell有ash，bash，ksh，csh,zsh等

2,shell入门

编写规范：

代码规范：#!/bin/bash 指定系统告知这个脚本要是用的shell脚本解释器

Shell 相关指令

文件命名规范：

文件名.sh .sh是linux下的bash shell的默认后缀

使用流程

1，创建.sh文件 touch/vim

2，编写shell代码

3，执行shell脚本 脚本必须得有执行权限

案例1：创建test.sh实现一个shell脚本程序，输出helloworld

代码：#touch test.sh

vim test.sh

#!/bin/bash

echo 'hello world'

#ll test.sh 查看权限

#chmod +x test.sh 更改权限

# ./test.sh 执行

案例2：使用root用户账号创建并执行test2.sh,实现创建一个shelltest用户，并在其家目录中新建文件 try.html

代码：

#touch test2.sh

#vim test2.sh

#!/bin/bash

useradd shelltest

touch /home/shelltest/try.html

#chmod +x test2.sh

#/bin/bash test2.sh

shell脚本分为简单的写法和复杂的写法

简单的写法：简单命令的堆积

复杂写法:程序设计

二，shell进阶

1，变量

name="baoboahui"

echo $name

案例2：定义一个变量，输出当前时间，要求格式为：“年-月-日 时:分:秒”

#!/bin/bash

dt=`date +'%F %T'` 反引号，esc键下方的那个键

echo $dt

readonly 变量名 ----->只读变量

#!/bin/bash

a=10

readonly a

a = 20

echo $a

read -p 提示信息 变量名 ------>接收用户输入

案例：编写一个脚本test6.sh，要求执行之后提示用户输入文件的名称（路径），然后自

动为用户创建该文件

脚本内容：

#!/bin/bash

read -p '请输入需要创建的文件路径：' filepath

touch $filepath

echo '文件创建成功！‘

ls -l $filepath

添加权限并执行

# chmod +x test6.sh

# ./test6.sh

unset 变量名 --->删除变量

案例：定义变b=20，在输出b的值，随后删除b，最后在输出b

#!/bin/bash

b = 20

echo $b

unset b

echo $b

2，条件判断语句

语法1：

if condition

then

command1

command2

...

fi

单行写法（一般在命令行中执行的时候）：if[condition];then command;fi

语法2：

if condition

then

command1

command2

...

else

command

fi

语法3：

if condition1

then

command1

elif condition2

then

command2

else

commandN

fi

3，运算符

shell 中的运算符包括算数运算符，关系运算符，逻辑运算符，字符串运算符和文件算术运算符

+ 加法 `expr $a+$b` 使用反引号 ，expr是一款计算工具

-

\* `expr $a\\*$b`

\

%

=

== [$a == $b] 一定要有空格，在条件表达式中

!=

案例：两个数相加--- #expr 2+20 或者下面的

#/bin/bash

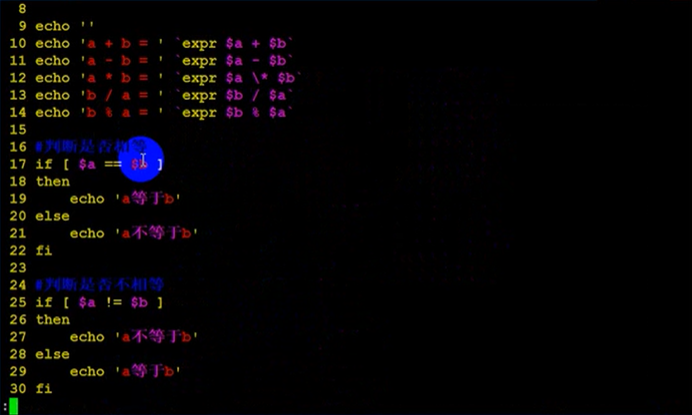
val = `expr 2 + 2`

echo "两数之和：$val"

注意：

表达式之间必须有空格

使用反引号



关系运算符

-eq 检测两个数是否相等 [ $a -eq $b ]

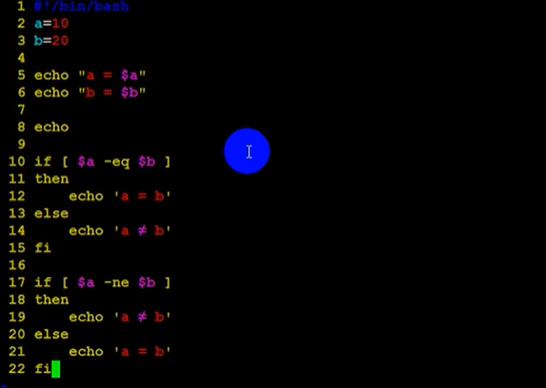
-ne 不相等，不相等返回true

-gt 检测左边的数是否大于右边的数

-lt 小于

-ge 大于等于

-le 小于等于



逻辑运算符

! 非运算 [ !false ]

-o 或运算 [ $a -lt 20 -o $b -gt 100 ] 返回true

-a 与运算

字符串运算符

= 检测两个字符串是否相等，相等返回true [ $a = $b ] 返回false

!= ，不相等返回true

-z 检测字符串长度是否为0，为0返回true

-n 不为0返回true [ -n $a ] 返回true

str 检测字符串是否为空，不为空返回true

文件测试运算符

-b file 检测文件是否是块设备文件，如果是，返回true [ -b $file ]

-c file 检测文件是否是字符设备文件，如果是，返回true [ -c file ]

-d file 目录，如果是，返回true

-f file 检测文件是否是普通文件（既不是目录，也不是设备文件）

如果是则返回true



4，shell脚本附带选项

问题描述：在linux shell中如何处理tall -10 access.log 这样的命令行选项

步骤：

调用 tail 指令

系统把后续选项传递给 tail

tail 先去打开指定的文件

取出最后 10 行

问题：自己写的shell 是否也可以像内置命令一样传递一些选项呢

答：可以传递，但是要做好接收处理

例如：

# ./test.sh a b c

接收：

在脚本中可以用 "$1" 表示a，类推

接收可以用“$”加上选项对应的序号即可

练习：创建自定义指令“user”，可以直接执行，要求该指令具备以下语法和功能

a，#user -add 用户名 [添加用户]

b. #user -del 用户名 [删除用户及其家目录]

#!/bin/bash

if [ $1 = '-add' ]

then

usesradd $2

else

userdel -r $2

fi

**MySQL基础**

**1，关于数据库**

mysql数据库（最好的RDBMS Relational Database Management System 关系数据库管理系统）和php是黄金搭档

其他数据库软件：

oracle，重量级数据库

MS SQ Server

Access

PostgreSQL

DB2

Mariadb（mysql的分支）

**2，mysql的安装与初始化**

1，linux下的软件安装方式

a，源码包

解包：

常用语法：

#tar -zxvf \*.tar.gz

#tar -jxvf \*.tar.bz2

选项含义：

-z 或 --gzip 或 --ungzip：通过zaip指令处理文件

-x 或 --extract 或 --get： 从文件中还原文件

-v：显示操作过程

-f 或 --file：指定一个文件

-j：支持bzip2解压文件

b，二进制包(rpm)

c，yum等傻瓜式安装

2，mysql的安装与初始化

**3，mysql的基本操作**

5，备份与还原

备份：

全量备份（数据+结构）：#mysqldump -uroot -p123456 -A > 备份文件路径

指定库备份(数据+结构)：#mysqldump -uroot-p123456 库名 > 备份文件路径

多个库备份（数据+结构）：

#mysqldump -yroot -p123456 --databases db1 db2 > 备份文件路径

备份成 **.sql** 文件

案例：每1分钟自动备份1次test数据库

#!/bin/bash

filename = "test\_" `date + '%Y%m%d%H%M%S` ".sql"

mysqldump -uroot -p密码 test> /root/$filename

还原：

1，还原全部数据库：

mysql 命令行：mysql> source 备份文件路径

系统命令行：#mysql -uroot -p123456 < 备份文件路径

2，还原单个数据库（需指定数据库）

1，mysql > user 库名

mysql > source 备份文件路径

2，#mysql -uroot -p123456 库名 < 备份文件路径

3，还原单个数据库的多个表（需指定数据库）

1，mysql> use 库名

mysql> source 备份文件路径

2，mysql -uroot -p123456 库名 < 备份文件路径

4，还原多个数据库（一个备份文件里有多个数据库的备份，此时不需要指定数

据库）

1，mysql 命令行：mysql> source 备份文件路径

2，系统命令行：mysql -uroot -p123456 < 备份文件路径

**4，扩展**

1，mysql 远程管理工具

B/S：PMA(phpMyAdmin)

C/S：navicat,mysql workbranch

**Yum项目上线实践（网站运维）**

**一，编译安装与卸载Nginx**

Ngnix：轻量级网站服务器软件

**二，关于LAMP**

**三，LAMP环境部署**

1，PHP与Apache的安装

2，MySql的安装与初始化

3，项目上线