常见操作系统：Windows ，Mac os，Linux

Windows：闭源，收费，稳定性差，安全性低，易用性好，用户体验好一些，图形界面做的比较好。主要应用在个人电脑上，其服务器版本Windows Server 市场占有率较低。

Linux：开源，免费，安全性、稳定性较高，易用性较差，图形界面做的不太好。主要应用在服务器上，特别是在中国。安装工具：xshell：用来远程连接linux系统，操作linux系统的工具。Xftp：用来远程连接linux系统，向linux系统上传文件的工具。

一、常用命令

1，Linux中的路径

/ 表示根目录，绝对路径

./ 表示当前目录

../ 表示上一级目录

/root root用户的主目录

/bin存放公用的命令文件，root管理员账号，它的某些命令普通用户是没有的，那些管理员有并且普通用户也有的命令脚本文件就保存在此目录下。

/dev linux系统中把所有的硬件包括外接设备使用文件来表示，那么这些代表了硬件的文件就保存在此目录下。

/home 普通用户的家（根目录），那么每新建一个用户，系统就会自动在该目录下为新用户创建一个用户目录（名字为用户名）

/sbin 存放管理员的命令文件。

/var: 存放一些日志文件的目录。

/etc 存放系统的各种配置文件，包括系统环境变量的配置文件、防火墙、网络等的配置文件。

/lib 存放一些类库。

/opt 存放用户暂时使用的应用软件的目录。

/usr 存放用户的自定义的常用的软件，各种应用软件的linux版本安装包默认的安装目录会在usr目录下。

/tmp 存放系统中软件运行的时候使用到的临时文件。

2，静态配置ip地址

1）打开ifcfg-eth0文件并进入编辑模式：

vi /etc/sysconfig/network-scripts/ifcfg-eth0

2）在文件中配置以下信息：

ONBOOT=yes #是否在系统启动的时候启动网络

BOOTPROTO=static #将网络ip地址改为静态的

IPADDR=ip地址 #指定静态的ip地址

NETMASK=255.255.255.0 #子网掩码

GATEWAY=192.168.80.1 #网关

3）敲击esc键，退出编辑模式，输入:wq命令，保存退出

5）重启网卡：service network restart

3，常用命令

## ls :显示指定目录下的文件及文件夹。

ls 具体目录：展示指定目录下的文件及文件夹

ls -l:展示文件及文件夹的详细信息

ls -a:展示当前目录下所有的文件及文件夹，包括隐藏文件或文件夹

ls -R :显示该目录及其子目录的内容

## cd:切换目录

cd 具体目录切换到指定目录

cd ..返回上层目录

## ifconfig：查看ip地址

## dhclient：动态获取ip地址

## 操作文件的命令：

1）创建文件：

vi 文件名：如果指定的文件名存在，就打开该文件，如果不存在，就创建该文件并打开文件。

a或i：开启文件编辑模式。

Esc：退出文件编辑模式。

:w ：保存

:q ：退出

:wq ：保存加退出

:wq! ：强制保存退出

注意：使用vi命令打开文件后，输入命令:set nu可以显示文件中的行号

touch 文件名：直接创建一个文件

cat > 文件名：创建文件，可以直接在文件中输入内容，CTRL+D保存并结束。

cat 文件名：查看文件的内容

2）删除文件：

rm 文件名：删除指定的文件

3）复制文件：

cp 源文件 目标路径

4）移动文件：

mv 源文件 目标路径

5）创建快捷方式：

ln -s 源文件 快捷文件

##目录操作的命令：

1）创建目录：

mkdir 目录名

mkdir 目录名1 目录名2 目录名3… 同时创建多个目录

mkdir -p 父目录/子目录/孙目录 递归的创建目录

2）删除目录：

rmdir 目录名 删除空目录

rm -rf 目录名 强行删除非空目录

3）复制目录：

cp -r 源目录 目标目录

4）移动目录/重名了文件夹：

mv 源目录 目标目录

5）查看当前用户：

命令：who is my

6）查看当前目录：

命令：pwd

##涉及权限操作的命令：

关于用户组：

1）创建用户组：

groupadd 用户名

2）查看用户组：

cat /etc/group

3）删除用户组：

groupdel 用户组名

4）修改用户组名：

groupmod -n 新名字 旧名字

关于用户：

1）新建用户：

useradd 用户名 创建一个用户，系统会在home目录下自动创建一个跟用户同名的目录，还要创建一个同名的用户组

useradd -g test -G dev Maria 新增用户Maria，主组为test，并附加到dev中

2）删除用户

userdel 用户名

userdel -r 用户名 删除用户的同时会删除home下的用户目录

修改用户的用户组：

3）修改用户

usermod -g 新用户组 用户名

4）查看用户

cat /etc/passwd

liu:x:502:500::/home/liu:/bin/bash

liu表示用户名

502表示用户id

500表示用户组名

5）查看用户所在的用户组

groups 用户名

6）切换登录用户：

su – 用户名

7）设置用户密码

passwd 用户名

6）权限编辑：

语法：chmod 指定权限 文件或目录

方式一：chmod u+rwx,g+rx,o+r 目标文件或目录

方式二：chmod 数字1数字2数字3 文件名

数字1：表示所有者的权限

数字2：表示同组用户的权限

数字3:表示其他用户的权限

注意：如果什么权限都没有就为0.

方式三：chmod u=rwx,g=r

-d：表示目标是目录

-：表示目标是一个普通文件

-l：表示目标是一个快捷方式

u:文件或目录的所有者

g:文件或目录的所有者的同组用户

o:文件或目录的所有者以及同组用户之外的用户

a:所有用户

r:表示读权限 对应数字为4

w:表示写权限 对应数字为2

x:表示执行权限 对应数字为1

7）更改文件或目录的所有者：

chown 用户名 文件或目录名

chown -R 用户名 文件或目录名（wang）

8）文件搜索

find 起始目录 参数（查找类型） 指定条件

find . –name “\*test\*”

9）查看系统运行的进程：

ps -ef 查看所有运行的进程

ps -ef | grep tomcat(指定的进程)

kill -9 pid（进程id） 强制关闭指定的进程

二、常用软件的安装

1，安装方式：

1）绿色安装，解压缩安装，安装好之后，如果要卸载，可以非常干净、方便的卸载。

tar -zxvf 压缩包名 -C 目录名 ：解压到指定目录

tar –cvf 压缩包名 被压缩的文件或目录名 ： 压缩命令

tar –xvf 压缩包名 解压命令

tar –zxvf 压缩包名 解压到当前目录

2）通过运行安装文件的形式安装，在linux中，卸载的时候比较麻烦。

rpm –ivh xxxx.rpm 安装一个rpm包

rpm -qa|grep gcc 查找和gcc相关的rpm安装包，即检测系统中是否有gcc的残留。

rpm -qpl xxxx.rpm 查看rpm安装程序的默认安装目录

rpm -e xxxx 卸载一个rpm包

3）在线安装，不用下载任何的安装包或安装程序文件，直接在线安装。本质上是rpm

yum install 软件名

2，jdk的安装

1）下载jdk的linux版本的安装包

2）运行rpm安装包，安装

命令：rpm -ivh jdk-8u131-linux-i586.rpm

3）配置环境变量，在/etc/profile文件中配置：

export JAVA\_HOME=/usr/java/jdk8（jdk安装目录）

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

4）运行命令使配置生效

source profile

2，tomcat绿色安装：

1）下载压缩包

2）解压

tar –zxvf 压缩包名

3）配置防火墙

service iptables stop|start|restart 操作防火墙停止|启动|重启

在/etc/sysconfig/iptables配置文件中，配置8080端口。增加配置如下

-A INPUT -m state --state NEW -m tcp -p tcp --dport 8080 -j ACCEPT

4）重启防火墙。

3，mysql安装：

1）下载压缩包

2）解压压缩包

tar –zxvf 压缩包名字

3）移动解压出来的文件夹到/usr/local/目录下，并改名为mysql

mv 解压出来的目录名 /usr/local/mysql

4）进入mysql目录，创建目录data

mkdir data

5）创建mysql用户，并且设置mysql目录的权限

useradd mysql

chown -R mysql .

chmod -R 755 ../mysql

6）安装一些mysql依赖的语言环境：

C++：yum -y install make gcc-c++ cmake bison-devel ncurses ncurses-devel libaio-devel

num：yum -y install numactl

7）初始化mysql：

./bin/mysqld --initialize --user=mysql --basedir=/usr/local/mysql --datadir=/usr/local/mysql/data

正常运行过后生成初始密码，需暂时保存，后面要修改

8）在/usr/local/mysql/supports-files目录下把mysql.server文件复制到/etc/init.d目录下，并且改名为mysql，然后打开该文件，配置里边的basedir和datadir两项。

basedir=/usr/local/mysql

datadir=/usr/local/mysql/data

9）在/etc/my.cnf文件中配置mysql信息。

datadir=/usr/local/mysql/data

socket=/usr/local/mysql/mysql.sock

user=mysql

symbolic-links=0

[client]

socket=/usr/local/mysql/mysql.sock

[mysqld\_safe]

#log-error=/var/log/mysqld.log

#pid-file=/var/run/mysqld/mysqld.pid

10）启动。

/etc/init.d/mysql start

11）配置环境变量，在/etc/profile中

export PATH=/usr/local/mysql/bin:$PATH

12）连接

mysql –u root –p

13）改密码，授权远程主机连接

mysql>alter user 'root'@'localhost' identified by 'root';

mysql>use mysql;

msyql>update user set user.Host='%' where user.User='root';

mysql>flush privileges;

mysql> grant all privileges on \*.\* to 'root'@'%' ;

mysql>flush privileges;

mysql>quit;

14）配置防火墙，打开3306端口，重启防火墙。

在/etc/sysconfig/iptables配置文件中，配置3306端口。增加配置如下

-A INPUT -m state --state NEW -m tcp -p tcp --dport 3306 -j ACCEPT

4，nginx安装：

#服务器程序：

Tomcat：servlet容器，运行servlet，可以处理静态、动态资源的请求，但是tomcat处理静态资源的请求效率较低，并发量低。Tomcat并发处理请求的量级在百位数。依赖jdk环境。

Nginx：http服务器，处理http请求，可以处理静态资源的请求，而且效率非常高，不能运行servlet，并发性特别好，并发处理请求的量级在万位数。搭建服务器集群、实现动静态分离部署的最佳选择。

Apache：http服务器，处理静态资源请求，而且效率较高，不能运行servlet，并发性较好。

Weblogic、JBoss：应用服务器，只能运行动态资源。收费。

安装：

1）下载安装包

2）解压安装包

tar -zxvf 安装包名字

3）安装依赖

yum -y install make zlib zlib-devel gcc-c++ libtool openssl openssl-devel pcre pcre-devel

4）在任意目录创建一个文件夹，取名nginx

mkdir nginx

5）编译安装

在解压出来的目录中执行

./configure --prefix=指定安装目录（刚才创建的名为nginx的目录）

make

make install

6）配置防火墙，打开80端口

在/etc/sysconfig/iptables配置文件中，配置3306端口。增加配置如下

-A INPUT -m state --state NEW -m tcp -p tcp --dport 3306 -j ACCEPT

7）然后在nginx/sbin目录下找到nginx的文件，启动。

./nginx

5，服务器集群的搭建：

随着互联网项目的兴起，对于系统服务端的高并发、高可用等要求越来越高，单体应用部署在一台tomcat上，可以处理的并发请求数较低，当在某些特定时段，访问量瞬间提高，有可能造成服务器的响应效率低下，甚至宕机。那么有一种解决方案就是通过将多台服务器组合到一起，共同分担服务器负载，安照一定比例合理的分配请求，从而提高系统整体的并发性、可用性、安全性。

系统为了提高并发性有两种解决方案：

1）纵向扩展系统，提高服务器硬件的配置，通过安装其他的并发性能高的服务器。

2）横向扩展系统，通过使用多台普通的服务器，搭建一个负载均衡的服务器集群，每个服务器上安装一个tomcat，每个tomcat中部署一个相同的项目，处理相同的请求，只是每台分担一定数量的请求。

3）搭建：

#1）修改tomcat的端口号，保证两个tomcat的所有端口号都不重复。

#2）修改nginx配置文件，在nginx/conf/nginx.conf文件中：

注册tomcat服务器：

upstream test.com{

server ip地址:port;

}

配置.jsp,.do等动态请求：

location ~ \.(jsp|jspx|do|action)?$ {

proxy\_pass http://test.com;#指定负载均衡的方式跟服务器

proxy\_redirect default;

}

配置静态请求的目标资源目录：

location ~ .\*\.(html|js|css|ico|jpg|jpeg|png|JPG|JPEG|PNG|eot|svg|ttf|woff) {

root html;

}

配置没有后缀的动态资源请求：

location ~ .\*$ {#没有后缀的请求

proxy\_pass http://test.com;#指定负载均衡的方式跟服务器

proxy\_redirect default;

}

#3）在tomcat上部署web项目，重启tomcat，重启nginx。

./nginx –s reload

4）负载均衡：就是让代理服务器分发请求是按照一定的规则来分发，不至于使集群中的有些服务器负载过大，而另外一些服务器又几乎不承担负载。在nginx+tomcat组成的集群中要实现负载均衡，可以通过nginx的配置文件（在conf/nginx.conf文件中）配置不同的规则去实现。

负载均衡的规则：

#1）轮询，默认的规则。

upstream test.com {

server 192.168.80.175:18080;

server 192.168.80.175:28080;

server 192.168.80.175:38080;

}

#2）加权轮询，按照一定的权重来进行轮询

upstream test.com {

server 192.168.80.175:18080 weight=1;

server 192.168.80.175:28080 weight=2;

}

#3）热备份，就是用一台运行状态的服务器做备份，一旦服务器出现故障，备份的服务器马上自动顶替上去，处理请求。用户感受不到服务端的故障。

upstream test.com {

server 192.168.80.175:18080 ;

server 192.168.80.175:28080 backup;#备份服务器

}

#4）ip\_hash,一个用户访问系统的时候，第一个请求进入了tomcat1，tomcat1为其创建一个会话，第二个请求进入了tomcat2，tomcat2又会为其创建一个会话，这样用户就无法使用同一个会话，就会造成错误，ip\_hash,根据IP地址的前三位进行计算得到一个结果，那么第一次请求过后，计算的结果被保存，然后第二次请求进来的时候，就根据ip地址前三位来计算，如果结果一样，就把该请求分发到同一台服务器。

upstream test.com {

ip\_hash;

server 192.168.80.175:18080 ;

server 192.168.80.175:28080 ;

}

搭建一个服务器集群，session共享问题通过ip\_hash方式解决，但是这种方式随机性较高，如果我们要较好的控制请求的分发（负载均衡），我们可以采用轮询的方式进行负载均衡，通过redis存储并管理session的方式实现session共享，这个是目前比较推荐session共享解决方案。具体实现起来非常简单，下载相应的插件，配置一下，导几个jar包就可以了。

#）通过插件tomcat-cluster-redis-session-manager实现在redis中共享session：

#1）下载类库

https://github.com/ran-jit/tomcat-cluster-redis-session-manager/wiki

#2）解压

#3）jar包放入tomcat的lib目录中

#4）再把解压目录下的配置文件配置后放入tomcat的conf目录下，注意要修改相应的配置

#5）在tomcat的context.xml配置文件中添加以下两条配置信息：

<Valve className="tomcat.request.session.redis.SessionHandlerValve" />

<Manager className="tomcat.request.session.redis.SessionManager" />

#6）实现了session共享

采用集群架构对系统影响：

a：并发量提高

b：系统的稳定性或安全性提高

c：效率提高，系统扩展性或灵活性增强

d：开发的时候需要注意一些问题，如session共享

采用纵向扩展，性能提高一杯，价格提高不止一倍；采用横向扩展，性能提高一倍，价格也只提高一倍

5）安装redis：

Redis是一个非常优秀的noSQL数据库，存储数据是以key-value形式存储，数据的存取都是在内存中进行，效率非常高。特点：

#1）效率极高。

#2）相对于其他的同类产品支持的数据类型较多。

#3）相对于其他的同类产品支持持久化的（把内存中的数据存到硬盘上）。

基于内存，内存稀缺资源，所以redis一般被用做缓存。一旦redis服务器宕掉，重启时可以将持久化的数据再读入内存，不至于丢失，缓存数据更安全。

#安装:

#1）下载安装包

#2）解压压缩包

tar –zxvf 安装包名

#3）安装依赖

yum -y install gcc automake autoconf libtool make

#4）安装redis

进入redis解压目录

make

#5）启动。

进入解压目录下的src中，

./redis-server ../redis.conf

键入内容：（输入密码auth roo）t

./redis-cli

#6）解除绑定本地端口、设置守护线程和密码，在redis解压目录下找到redis.conf文件：

69行：#bind 127.0.0.1

136行：daemonize yes

507 行：requirepass root

#7）注册端口，默认为6379，重启防火墙

##redis支持的数据类型：

1）字符串String

添加：set key value

获取：get key

删除：del key

2）哈希Hash

添加：hmset key field1 value1 field2 value2 field3 value3…

获取1：hgetall key

获取2：hget key field

删除：hdel key field

3）无序集合Set

添加：sadd key value

获取：smembers key

删除：srem key value

4）有序集合Sorted Set

添加：zadd key score value

获取：zrange key start end [withscores]

删除：zrem key value

5）有序列表List

添加到列表左边：lpush key value

添加到列表右边：rpush key value

获取：lrange key start end

lpop key： 删除列表最左边的元素，并将元素返回

rpop key： 删除列表最右边的元素，并将元素返回

6）通用命令

1. keys \* : 查询所有的键

2. type key ： 获取键对应的value的类型

3. del key：删除指定的key value

7）通过Java程序访问redis服务：

Jedis.jar 是一个通过Java访问redis服务的类库。

Redis中存储的数据是要求必须进行序列化的，我们可以通过commons-lang3提供的工具类进行序列化。

步骤：

//创建连接池配置对象

JedisPoolConfig conf = new JedisPoolConfig();

conf.setMaxTotal(100);

conf.setMaxIdle(50);

conf.setMinIdle(20);

//创建一个连接池对象

pool = new JedisPool(conf, "192.168.80.187", 6379, 20000, "root", 2);

//通过连接池对象获取jedis会话

jedis = pool.getResource();

//对应String可以使用setex()方法存储可以指定过期时间的 key value

//将activecode：hehe键值对存入redis，并且20秒后自动删除该键值对

jedis.setex("activecode",20,"hehe");

// 获取hash的所有map中的数据

Map<String, String> user = jedis.hgetAll("user");

//添加，重载方法为append(byte[] key, byte[] value)

jedis.append(key, value);

//设置过期时间，单位为秒，重载方法为expire(byte[] key, int seconds);

jedis.expire(key, seconds);

##SVN安装：

1）yum安装：

yum –y install subversion

2）进入/opt目录，并创建目录svndata/projects/easybuy

cd /opt

mkdir svndata

cd svndata

mkdir projects

cd projects

mkdir easybuy(项目目录)

3）回到/opt目录下，初始化资源库：

svnadmin create svndata/projects/easybuy

成功执行后，会在conf目录中产生三个配置文件：authz、passwd、svnserve.conf

4）配置三个配置文件：

vi authz

在[groups]下边配置以下信息：

[/]

组名=权限 liu=rw

vi passwd

在[users]下边配置以下信息：

用户名=密码 liu=root

vi svnserve.conf

配置以下信息：

anon-access=none

auth-access=write (读写权限)

password-db=passwd

authz-db=authz

配置完成

5）注册防火墙

6）回到/root/opt目录，启动SVN服务：

svnserve -d -r svndata --listen-port=3690

注意端口号 svndata是资源库根目录

7）用小乌龟连接并检出或导入。

8）访问：

svn://ip:3690/projects/easybuy