CentOS6.4服务器环境配置

1. 安装JDK
2. su用户获取root权限：

root用户登录。

方法一：vim /etc/sudoers 修改sudoers文件找到下面一行，把前面的注释（#）去掉

432x36

417x37

然后修改用户，使其属于root组（wheel）,命令如下：

usermod –g root su 修改完毕，用su用户登录，然后用su –切换，即可获得root权限进行操作。

方法二：vim /etc/sudoers 修改sudoers文件，找到下面一行，在root下面添加一行，如下所示：

345x52

修改完毕，用su用户登录，然后用命令su -，即可获得root权限进行操作。

方法三：vim /etc/passwd 修改/etc/passwd文件，找到如下行，把用户ID修改为0，如下所示：

329x18

308x18

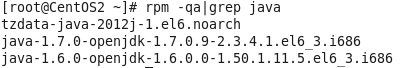
修改后保存，用su用户登录后，直接获取的说就是root用户的权限。

1. 删除默认OpenJDK,安装sun公司的JDK:

第一步：查看自带JDK是否已安装：用命令java –version查出下面信息：



第二步：先卸载OpenJDK，先查看rpm –qa|grep java



卸载: rpm –e –nodeps java-1.6.0-openjdk-1.6.0.0-1.50.1.11.5.el6\_3.i686

rpm –e –nodeps java-1.7.0-openjdk-1.7.0.9-2.3.4.1.el6\_3.i686

第三步：安装sun公司的JDK。

进入usr下lib，cd /usr/lib。解压安装tar –zxvf /home/su/下载/jdk-7u21-linux-i586.tar.gz

第四步：修改环境变量

vim ~/.bashrc

添加

export JAVA\_HOME=/usr/lib/jdk1.7.0\_21  
export JRE\_HOME=${JAVA\_HOME}/jre  
export CLASSPATH=.:${JAVA\_HOME}/lib:${JRE\_HOME}/lib  
export PATH=${JAVA\_HOME}/bin:$PATH

保存退出，并输入以命令source ~/.bashrc使之立即生效。

第五步：配置默认JDK



update-alternatives --install /usr/bin/java java /usr/lib/ jdk1.7.0\_21/bin/java 300  
update-alternatives --install /usr/bin/javac javac /usr/lib/ jdk1.7.0\_21/bin/javac 300  
update-alternatives --install /usr/bin/jar jar /usr/lib/ jdk1.7.0\_21/bin/jar 300  
update-alternatives --config java

第六步：测试

Java –version 如下所示，表示安装成功



1. 安装配置Tomcat

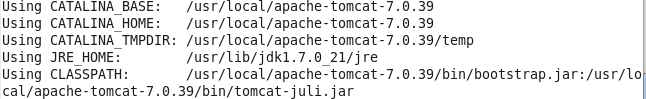
cd /usr/local

tar –zxvf apache-tomcat-7.0.39.tar.gz

启动tomcat, /usr/local/apache-tomcat-7.0.39/bin/startup.sh

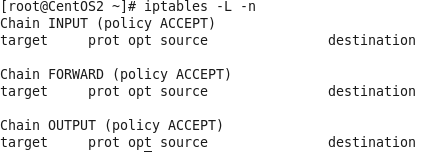


如果出现下图所示信息说明已成功启动。



清除防火墙原有规则：

iptables –F （清除预设表filter中的所有规则链的规则）iptables –X （清除预设表filter中使用者自定链中的规则）。Iptables –L –n查看如下图所示：



/etc/rc.d/init.d/iptables save保存



设定预设规则，对于filter表中的INPUT、FORWARD链，超出规则的数据包放弃，对于OUTPUT链，操作规则的数据包通过。



添加规则：

为了采用远程SSH登录，开启22端口

iptables –A INPUT –p tcp –dport 22 –j ACCEPT

iptables –A OUTPUT –p tcp –sport 22 –j ACCEPT

WEB服务器，开启80端口

iptables –A INPUT –p tcp –dport 80 –j ACCEPT

iptables –A OUTPUT –p tcp –sport 80 –j ACCEPT

允许icmp包通过，也就是允许ping

iptables –A OUTPUT –p icmp –j ACCEPT

iptables –A INPUT –p icmp –j ACCEPT

允许loopback,不然会导致DNS无法正常关闭等问题

iptables –A INPUT –i lo –p all –j ACCEPT

iptables –A OUTPUT –o lo –p all –j ACCEPT

开放8080端口：

iptables –A INPUT –p tcp –dport 8080 –j ACCEPT

iptables –A OUTPUT –p tcp –sport 8080 –j ACCEPT

开启转发功能（在做NAT时，FORWARD默认规则是DROP时，必须做）

iptables –A FORWARD –i eth0 –o eth1 –m state –state RELATED,ESTABLISHED –j ACCEPT

iptables –A FORWARD –i eth0 –o eh0 –j ACCEPT

丢弃坏的TCP包

iptables –A FORWARD –p TCP ! –-syn –m state --state NEW –j DROP

处理IP碎片数量，防止攻击，允许每秒100个

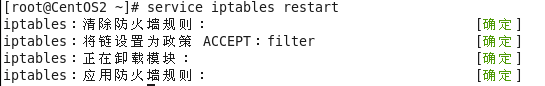
iptables –A FORWARD –f –m limit –-limit 100/s –-limit-burst 100 –j ACCEPT

设置icmp包过滤，允许每秒1个包，限制触发条件是10个包

iptables –A FORWARD –p icmp –m limit –-limit 1/s –-limit-burst 10 –j ACCEPT

/etc/rc.d/init.d/iptables save保存

service iptables restart重启防火墙



ifconfig查看ip地址，如ip地址为192.168.150.129，如果<http://192.168.150.129:8080> 看到tomcat系统界面，说明安装成功。

停止tomcat, /usr/local/ apache-tomcat-7.0.39/shutdown.sh

配置tomcat自启动：

安装jsvc

cd /usr/local/apache-tomcat-7.0.39/bin

tar zxvf commons-daemon-native.tar.gz

cd commons-daemon-1.0.14-native-src/unix

./support/buildconf.sh

如果提示没有找到autoconf命令，下载autoconf-2.61.tar.gz安装。

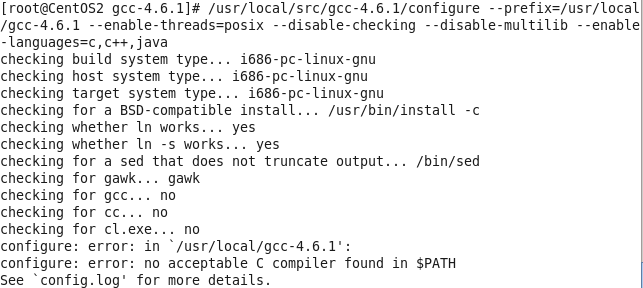
如果提示没有找到gcc命令，下载gcc-4.6.1.tar.bz2安装。

gcc安装方法:

bzcat gcc-4.6.1.tar.bz2|tar xvf – 解压下载的安装包，mv gcc-4.6.1 /usr/local/src 移动源文件到/usr/local/src目录，mkdir /usr/local/gcc-4.6.1建立安装目录。mkdir /usr/local/obj mkdir /usr/local/obj/ gcc-4.6.1建立目标目录

cd /usr/local/obj/gcc-4.6.1 进入目标目录

配置安装文件/usr/local/src/gcc-4.6.1/configure –-prefix=/usr/local/gcc-4.6.1 –-enable-threads=posix –-disable-checking –-disable-multilib –-enable-languages=c,c++,java 出现如下错误：



说明$PATH里没有找到C的编译器，通过whereis gcc 查看系统是否有gcc编译器，如果系统有gcc编辑器就设置下环境变量，比如export PATH=$PATH:/usr/local/bin,如果没有，先从/media/CentOS\_6.4\_Final/Packages安装gcc低版本rpm包，按下面次序依次安装

rpm –ivh /media/CentOS\_6.4\_Final/Packages/cpp-4.4.7-3 el6.i686.rpm –force --nodeps

rpm –ivh /media/CentOS\_6.4\_Final/Packages/kernel-headers-2.6.32-358.el6.i686.rpm –force --nodeps

rpm –ivh /media/CentOS\_6.4\_Final/Packages/glibc-headers-2.12-1.107.el6.i686.rpm –force --nodeps

rpm –ivh /media/CentOS\_6.4\_Final/Packages/glibc-devel-2.12-1.107.el6.i686.rpm –force --nodeps

rpm –ivh /media/CentOS\_6.4\_Final/Packages/libgomp-4.4.7-3.el6.i686.rpm –force --nodeps

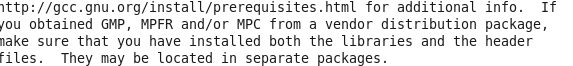
rpm –ivh /media/CentOS\_6.4\_Final/Packages/gcc-4.4.7-3.el6.i686.rpm –force --nodeps

再次配置安装文件/usr/local/src/ gcc-4.6.1/configure –-prefix=/usr/local/gcc-4.6.1 –-enable-threads=posix –-disable-checking –-disable-multilib –-enable-languages=c,c++,java发生错误，查看config.log如图所示：

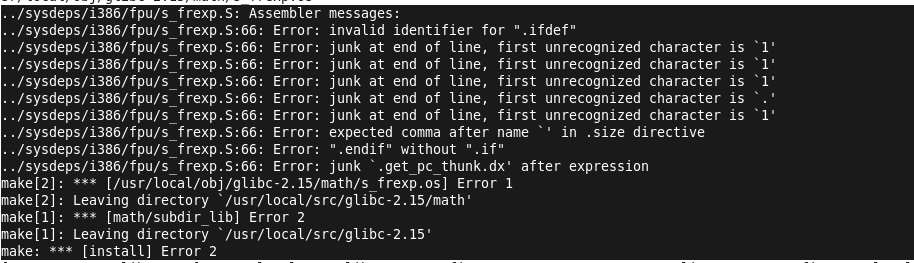


rpm –ivh /media/CentOS\_6.4\_Final/Packages/mpfr-2.4.1-6.el6.i686.rpm –force –nodeps

再次配置安装文件/usr/local/src/configure –-prefix=/usr/local/gcc-4.6.1 –-enable-threads=posix –-disable-checking –-disable-multilib –-enable-languages=c,c++,java发生错误，如图所示：



需要GMP、MPFR、MPC三个库，按顺序安装GMP、MPFR、MPC。中间可能需要安装glibc，安装配置glibc-2.15时可能会报错，如报下图所示错误：



修改sysdeps/unix/sysv/linux/i386/sysdep.h在include后面加上

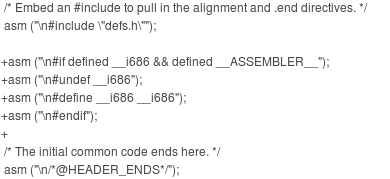
#if defined \_\_i686&&defined \_\_ASSEMBLER

#undef \_\_i686

#define \_\_i686 \_\_i686

#endif

修改/nptl/sysdeps/pthread/pt-initfini.c如下图所示：



安装好gcc后，重新安装jsvc

chmod 755 configure

./configure –with-java=/usr/lib/jdk1.7.0\_21

make

添加脚本让tomcat自动启动

首先为tomcat编辑一段启动脚本，并置于/etc/rc.d/init.d目录下，让tomcat作为一项系统服务。并随系统自动启动。

#!/bin/bash

#  
# Startup script for the tomcat  
#  
# chkconfig: 345 80 15  
# description: Tomcat is a Servlet+JSP Engine.  
export JDK\_HOME=/usr/lib/ jdk1.7.0\_21    
export JAVA\_HOME=/usr/lib/ jdk1.7.0\_21

# Source function library.  
. /etc/rc.d/init.d/functions  
start(){  
if [ -z $(/sbin/pidof java) ]; then  
echo "Starting tomcat"  
/usr/lib/ jdk1.7.0\_21/bin/startup.sh  
touch /var/lock/subsys/tomcat  
else  
echo "tomcat allready running"  
fi  
}  
stop(){  
if [ ! -z $(/sbin/pidof java) ]; then  
echo "Shutting down tomcat"  
/usr/lib/ jdk1.7.0\_21/bin/shutdown.sh  
until [ -z $(/sbin/pidof java) ]; do :; done  
rm -f /var/lock/subsys/tomcat  
else  
echo "tomcat not running"  
fi  
}  
case "$1" in  
start)  
start  
;;  
stop)  
stop  
;;  
restart)  
stop  
start  
;;  
status)  
/usr/lib/ jdk1.7.0\_21/bin/catalina.sh version  
;;  
\*)  
echo "Usage: $0 {start|stop|restart|status}"  
esac

exit 0

chmod 755 /etc/rc.d/init.d/tomcat 将启动脚本属性设置为可执行

chkconfig --add tomcat 将tomcat加入到chkconfig中

chkconfig tomcat on 将tomcat设置为自启动

chkconfig --list tomcat 确认tomcat服务状态,如图所示，如果2~5都是启用，OK



/etc/rc.d/init.d/tomcat start 启动tomcat

1. 安装Apache

先卸载自带的apache，rpm –qa|grep httpd 查看与httpd相关的软件包



rpm –e –nodeps httpd-tools-2.2.15-26.el6.centos.i686

rpm –e –nodeps httpd-2.2.15-26.el6.centos.i686

下载apr-1.4.6.tar.gz,apr-util-1.5.2.tar.gz,httpd-2.4.4.tar.gz

cd /usr/local/src

tar –zxvf /home/su/下载/apr-1.4.6.tar.gz

mkdir /usr/local/apr-1.4.6

mkdir /usr/local/obj/apr-1.4.6

cd /usr/local/obj/apr-1.4.6

/usr/local/src/apr-1.4.6/configure -–prefix=/usr/local/apr-1.4.6

make

make install

cd /usr/local/src

tar –zxvf /home/su/下载/apr-util-1.5.2.tar.gz

mkdir /usr/local/apr-util-1.5.2

mkdir /usr/local/obj/apr-util-1.5.2

cd /usr/local/obj/apr-util-1.5.2

/usr/local/src/apr-util-1.5.2/configure -–prefix=/usr/local/apr-util-1.5.2 -–with-apr=/usr/local/apr-util-1.5.2

make

make install

cd /usr/local/src

tar –zxvf /home/su/下载/httpd-2.4.4.tar.gz

mkdir /usr/local/httpd-2.4.4

mkdir /usr/local/obj/httpd-2.4.4

cd /usr/local/obj/httpd-2.4.4

/usr/local/src/httpd-2.4.4/configure -–prefix=/usr/local/httpd-2.4.4 –-with-apr=/usr/local/apr-1.4.6 –-with-apr-util=/usr/local/apr-util-1.5.2 -–enable-so –-enable-mods-shared=most -–enable-rewrite=shared –-enable-proxy=shared –-enable-proxy-ajp=shared –-enable-proxy-balancer=shared –-enable-speling=shared

make

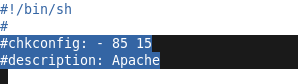
make install

启动 /usr/local/httpd-2.4.4/bin/apachectl start

设置自启动apache

cp /usr/local/httpd-2.4.4/bin/apachectl /etc/init.d/httpd

vim /etc/init.d/httpd 在文件中加入如图所示：



保存退出后 chkconfig httpd on

1. 整合Apache和Tomcat

vim /usr/local/httpd-2.4.4/conf/httpd.conf 修改apache的httpd.conf,将下面两行内容取消注释

#LoadModule proxy\_module modules/mod\_proxy.so  
#LoadModule proxy\_ajp\_module modules/mod\_proxy\_ajp.so

# Virtual hosts

#Include conf/extra/httpd-vhosts.conf

注释掉DocumentRoot和ServerAdmin

编辑httpd-vhosts.conf

vim /usr/local/httpd-2.4.4/conf/extra/httpd-vhosts.conf 编辑添加如下内容：

<VirtualHost \*:80>

ServerAdmin hudiezizai82@163.com

DocumentRoot "/usr/local/apache-tomcat-7.0.39/webapps" #此虚拟主机的主目录

<Directory "/usr/local/apache-tomcat-7.0.39/webapps">

Options Indexes FollowSymLinks

AllowOverride None

Order allow,deny

Allow from all

</Directory>

ServerName www.4kingkong.com #只是测试,生产环境可以写为对应的域名

ProxyPass /images ! #以下几行的意思是对/images,/sna 和/news/images几个目录的访问请求由apache处理其它都由tomcat处理.

ProxyPass /sna !

ProxyPass /news/images !

ProxyPass / ajp://localhost:8009/

ProxyPassReverse / ajp://localhost:8009/

ErrorLog "logs/3.249-error\_log"

CustomLog "logs/3.249-access\_log" common

</VirtualHost>

1. 安装Mysql

查看自带mysql版本情况 rpm –qa|grep mysql

删除自带mysql: rpm –e –-nodeps mysql-libs-5.1.66-2.el6\_3.i686

安装cmake rpm –ivh /media/CentOS\_6.4\_Final/Packages/cmake-2.6.4-5.el6.i686.rpm –force –nodeps

rpm –ivh /media/CentOS\_6.4\_Final/Packages/ncurses-devel-5.7-3.20090208.el6.i686.rpm –force –nodeps

rpm –ivh /media/CentOS\_6.4\_Final/Packages/bison-2.4.1-5.el6.i686.rpm –force –nodeps

下载mysql-5.6.11.tar.gz安装包

groupadd mysql

useradd –r –g mysql –s /usr/sbin/nologin mysql

cd /usr/local/src

tar –zxvf /home/su/下载/mysql-5.6.11.tar.gz

cd mysql-5.6.11

下载gmock-1.6.0.zip

cp /home/su/下载/gmock-1.6.0.zip /usr/local/src/mysql-5.6.11/source\_downloads/gmock-1.6.0.zip

cmake –DCMAKE\_INSTALL\_PREFIX=/usr/local/mysql –DMYSQL\_DATADIR=/usr/local/mysql/data –DDEFAULT\_CHARSET=utf8 –DDEFAULT\_COLLATION=utf8\_unicode\_ci –DWITH\_READLINE=1 –DWITH\_READLINE=1 –DWITH\_SSL=system –DWITH\_EMBEDDED\_SERVER=1 –DENABLED\_LOCAL\_INFILE=1 –DDEFAULT\_COLLATION=utf8\_general\_ci –DWITH\_MYISAM\_STORAGE\_ENGINE=1 –DWITH\_INNOBASE\_STORAGE\_ENGINE=1 –DWITH\_DEBUG=0 –DWITH\_SSL=bundled

make && make install

复制配置文件 cp my-default.cnf /etc/my.cnf

设置权限

chmod +x /usr/local/mysqld

chown –R mysql.mysql /usr/local/mysql

chown –R mysql:mysql /usr/local/mysql/data

配置开机启动

cp support-files/mysql.server /etc/init.d/mysqld

chmod +x /etc/init.d/mysqld

chkconfig -–add mysqld

chkconfig mysqld on

修改my.cnf配置

vim /etc/my.cnf

在[mysqld]中添加

datadir=/usr/local/msql/data

default-storage-engine=MyISAM

log-error=/usr/local/mysql/data/error.log

pid-file=/usr/local/mysql/data/mysql.pid

user=mysql

tmpdir=/tmp

保存退出

初始化数据库

/usr/local/mysql/scripts/mysql\_install\_db –-user=mysql –-basedir=/usr/local/mysql –-datadir=/usr/local/mysql/data &

启动mysql

/etc/init.d/mysqld start

查看是否有进程mysql

ps –ef|grep mysql

查看端口是否运行

Netstat –tnl|grep 3306

读取mysql版本信息

/usr/local/mysql/bin/mysqladmin version