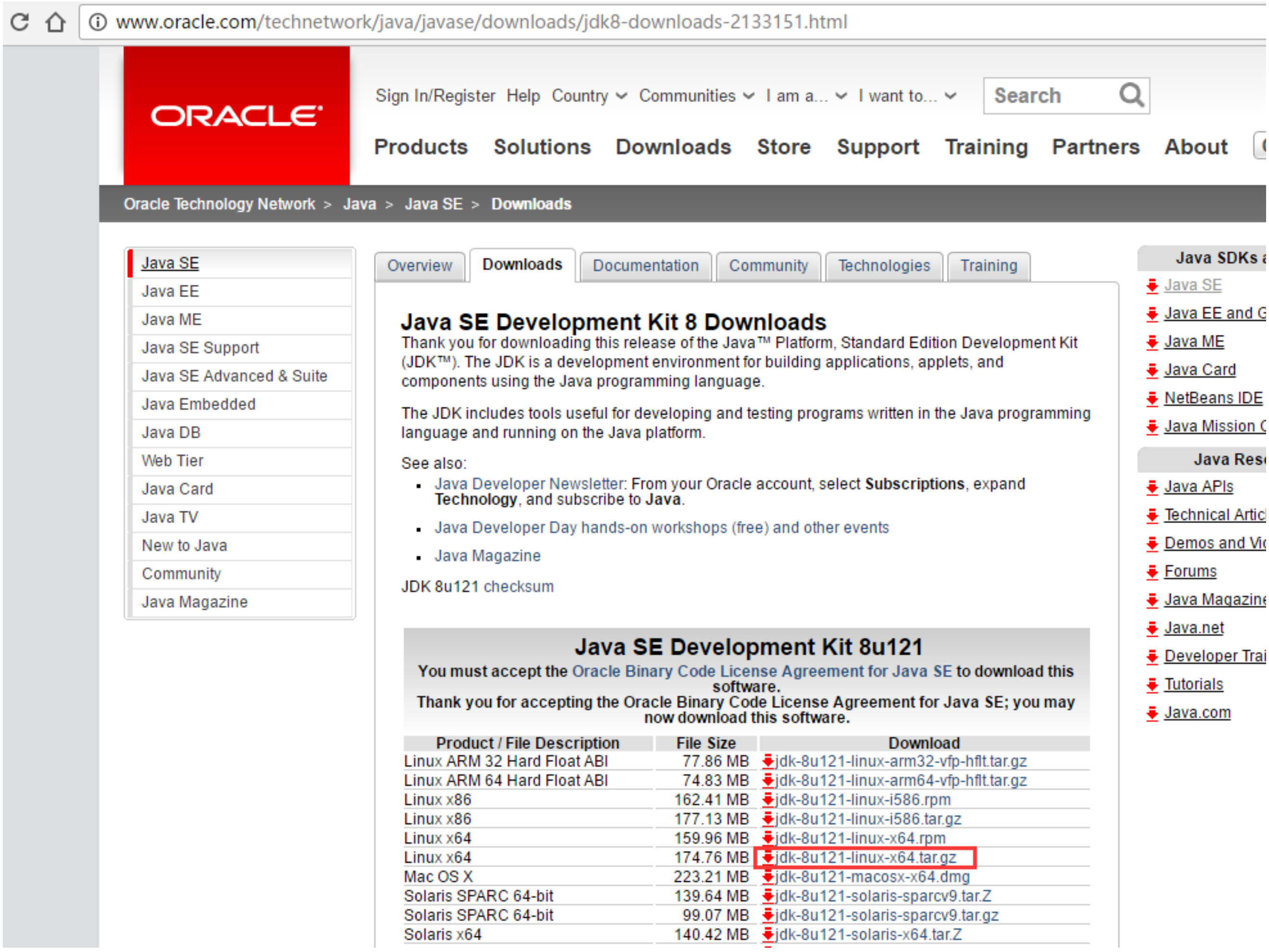
安装JDK

本节介绍如何安装 java jdk。软件包中包含的软件及版本：Tomcat：1.8.0_121

说明：这是写文档时参考的软件版本。您下载的版本可能与此不同。

- 准备工作

下载 JDK 安装包，地址：http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html



www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html

Java SE Development Kit 8 Downloads

Thank you for downloading this release of the Java™ Platform, Standard Edition Development Kit (JDK™). The JDK is a development environment for building applications, applets, and components using the Java programming language.

The JDK includes tools useful for developing and testing programs written in the Java programming language and running on the Java platform.

See also:

- Java Developer Newsletter: From your Oracle account, select **Subscriptions**, expand **Technology**, and subscribe to **Java**.
- Java Developer Day hands-on workshops (free) and other events
- Java Magazine

JDK 8u121 checksum

Java SE Development Kit 8u121

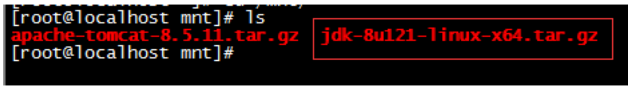
You must accept the Oracle Binary Code License Agreement for Java SE to download this software.

Thank you for accepting the Oracle Binary Code License Agreement for Java SE; you may now download this software.

Product / File Description	File Size	Download
Linux ARM 32 Hard Float ABI	77.86 MB	jdk-8u121-linux-arm32-vfp-hflt.tar.gz
Linux ARM 64 Hard Float ABI	74.83 MB	jdk-8u121-linux-arm64-vfp-hflt.tar.gz
Linux x86	162.41 MB	jdk-8u121-linux-i586.rpm
Linux x86	177.13 MB	jdk-8u121-linux-i586.tar.gz
Linux x64	159.96 MB	jdk-8u121-linux-x64.rpm
Linux x64	174.76 MB	jdk-8u121-linux-x64.tar.gz
Mac OS X	223.21 MB	jdk-8u121-macosx-x64.dmg
Solaris SPARC 64-bit	139.64 MB	jdk-8u121-solaris-sparcv9.tar.Z
Solaris SPARC 64-bit	99.07 MB	jdk-8u121-solaris-sparcv9.tar.gz
Solaris x64	140.42 MB	jdk-8u121-solaris-x64.tar.Z

下载JDK安装包

将下载好的 JDK 包上传到 Linux 上面：

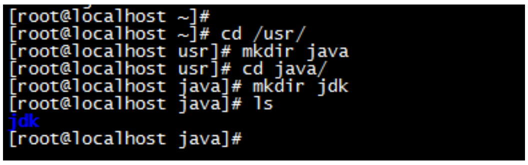


```
[root@localhost mnt]# ls
apache-tomcat-8.5.11.tar.gz  jdk-8u121-linux-x64.tar.gz
[root@localhost mnt]#
```

上传到Linux

- 操作步骤

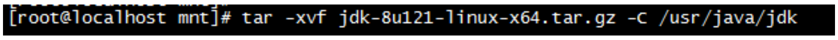
1. 新建一个目录。
   a. [root@localhost ~]# cd /usr/
   b. [root@localhost usr]# mkdir java
   c. [root@localhost usr]# cd java/
   d. [root@localhost java]# mkdir jdk
   e. [root@localhost java]# cd jdk
   f. [root@localhost jdk]#



```
[root@localhost ~]#
[root@localhost ~]# cd /usr/
[root@localhost usr]# mkdir java
[root@localhost usr]# cd java/
[root@localhost java]# mkdir jdk
[root@localhost java]# ls
jdk
[root@localhost java]#
```

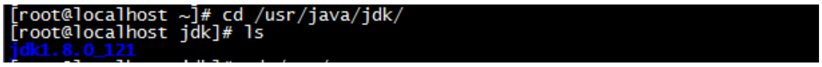
新建目录

2. 解压 jdk-8u121-linux-x64.tar.gz 到 jdk 目录下：
   a. [root@localhost mnt]# tar -xvf jdk-8u121-linux-x64.tar.gz -C /usr/java/jdk/



```
[root@localhost mnt]# tar -xvf jdk-8u121-linux-x64.tar.gz -C /usr/java/jdk
```

解压

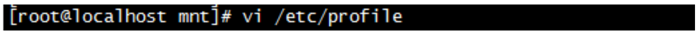


```
[root@localhost ~]# cd /usr/java/jdk/
[root@localhost jdk]# ls
jdk1.8.0_121
```

解压

3. 配置环境变量。

a. [root@localhost mnt]# vi /etc/profile

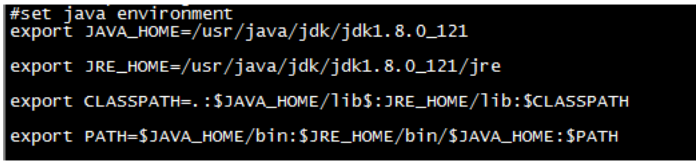


```
[root@localhost mnt]# vi /etc/profile
```

配置环境变量

4. 在底部添加以下内容。

```
#set java environment
export JAVA_HOME=/usr/java/jdk/jdk1.8.0_121
export JRE_HOME=/usr/java/jdk/jdk1.8.0_121/jre
export CLASSPATH=.:$JAVA_HOME/lib$:JRE_HOME/lib:$CLASSPATH
export PATH=$JAVA_HOME/bin:$JRE_HOME/bin/$JAVA_HOME:$PATH
```

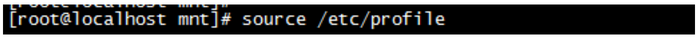


```
#set java environment
export JAVA_HOME=/usr/java/jdk/jdk1.8.0_121

export JRE_HOME=/usr/java/jdk/jdk1.8.0_121/jre

export CLASSPATH=.:$JAVA_HOME/lib$:JRE_HOME/lib:$CLASSPATH

export PATH=$JAVA_HOME/bin:$JRE_HOME/bin/$JAVA_HOME:$PATH
```

添加内容

5. 保存后执行以下命令:

a. [root@localhost mnt]# source /etc/profile

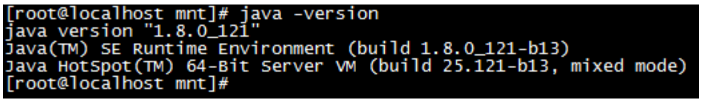


```
[root@localhost mnt]# source /etc/profile
```

执行命令

6. 验证安装。

a. [root@localhost ~]# java -version



```
[root@localhost mnt]# java -version
java version "1.8.0_121"
Java(TM) SE Runtime Environment (build 1.8.0_121-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.121-b13, mixed mode)
[root@localhost mnt]#
```

验证安装

我们可以看到 JDK 已经成功安装了。

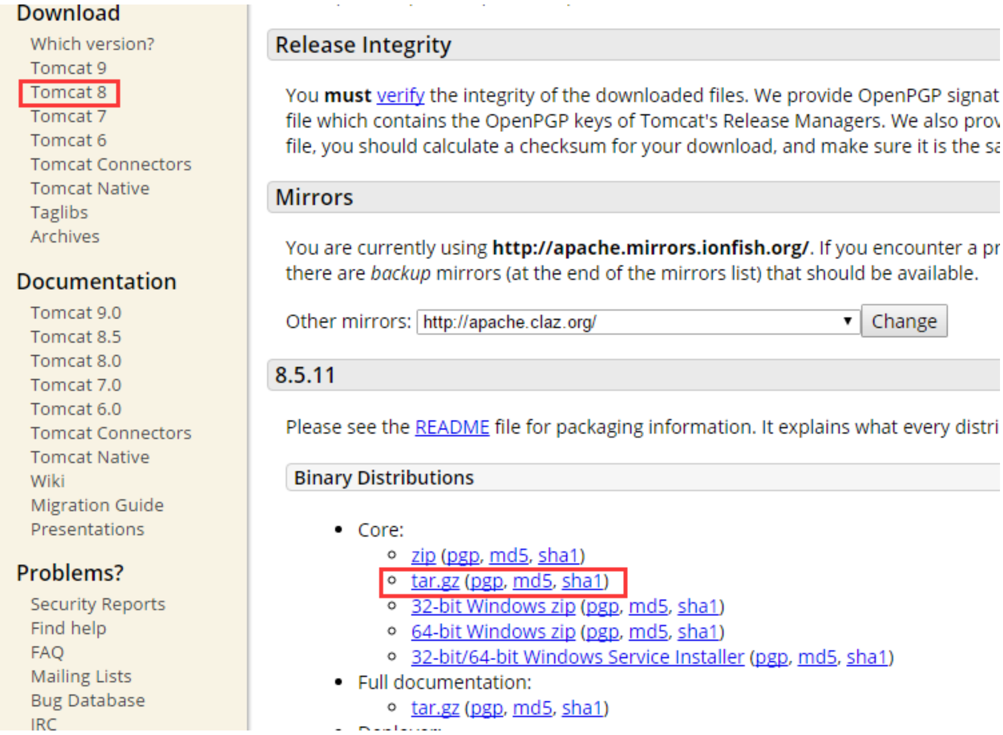
步骤 3 安装 Tomcat8.0

本节介绍如何部署Tomcat环境。软件包中包含的软件及版本：Tomcat：8.5.11 。

说明：这是写文档时参考的软件版本。您下载的版本可能与此不同。

- 准备工作

下载 tomcat linux 的包，地址：http://tomcat.apache.org/download-80.cgi



Download

Which version?
Tomcat 9
Tomcat 8
Tomcat 7
Tomcat 6
Tomcat Connectors
Tomcat Native
Taglibs
Archives

Documentation

Tomcat 9.0
Tomcat 8.5
Tomcat 8.0
Tomcat 7.0
Tomcat 6.0
Tomcat Connectors
Tomcat Native
Wiki
Migration Guide
Presentations

Problems?

Security Reports
Find help
FAQ
Mailing Lists
Bug Database
IRC

Release Integrity

You **must** verify the integrity of the downloaded files. We provide OpenPGP signat
file which contains the OpenPGP keys of Tomcat's Release Managers. We also prov
file, you should calculate a checksum for your download, and make sure it is the sa

Mirrors

You are currently using **http://apache.mirrors.ionfish.org/**. If you encounter a pr
there are *backup* mirrors (at the end of the mirrors list) that should be available.

Other mirrors: http://apache.claz.org/ ▾ Change

8.5.11

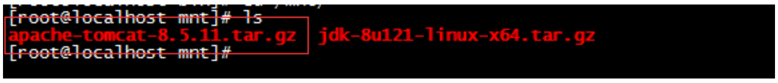
Please see the README file for packaging information. It explains what every distri

Binary Distributions

- Core:
  - zip (pgp, md5, sha1)
  - tar.gz (pgp, md5, sha1)
  - 32-bit Windows zip (pgp, md5, sha1)
  - 64-bit Windows zip (pgp, md5, sha1)
  - 32-bit/64-bit Windows Service Installer (pgp, md5, sha1)
- Full documentation:
  - tar.gz (pgp, md5, sha1)

下载

将下载好的 Tomcat 包上传到 Linux 上面：

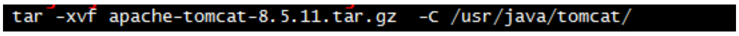


```
[root@localhost mnt]# ls
apache-tomcat-8.5.11.tar.gz  jdk-8u121-linux-x64.tar.gz
[root@localhost mnt]#
```

上传

- 操作步骤

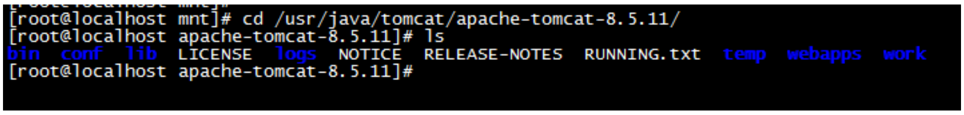
1. 解压 apache-tomcat-8.5.11.tar.gz
   a. # tar -xvf apache-tomcat-8.5.11.tar.gz -C /usr/java/tomcat/



```
 tar -xvf apache-tomcat-8.5.11.tar.gz  -C /usr/java/tomcat/
```

解压

2. 解压之后，我们进入解压的文件：
   a. [root@localhost mnt]# cd /usr/java/tomcat/apache-tomcat-8.5.11/

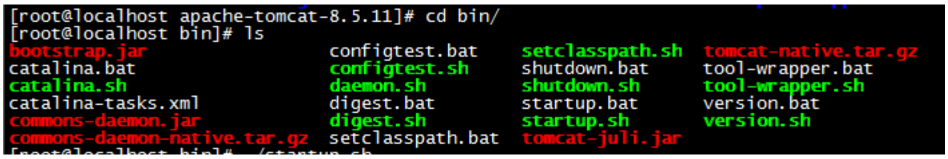


```
[root@localhost mnt]# cd /usr/java/tomcat/apache-tomcat-8.5.11/
[root@localhost apache-tomcat-8.5.11]# ls
bin  conf  lib  LICENSE  logs  NOTICE  RELEASE-NOTES  RUNNING.txt  temp  webapps  work
[root@localhost apache-tomcat-8.5.11]#
```

进入解压的文件

   - bin 目录中存放 Tomcat 的一些脚本文件，最重要的也是用的最多就是启动和关系 tomcat 服务脚本。
   - conf：存放 Tomcat 服务器的各种全局配置文件，其中最重要的是 server.xml 和 web.xml。
   - webapps：Tomcat 的主要 Web 发布目录，默认情况下把 Web 应用文件放于此目录。
   - logs：存放 Tomcat 执行时的日志文件

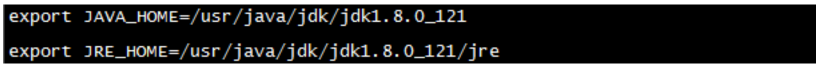
3. 进入到 tomcat 的 bin 目录：
   a. [root@localhost apache-tomcat-8.5.11]# cd bin/



```
[root@localhost apache-tomcat-8.5.11]# cd bin/
[root@localhost bin]# ls
bootstrap.jar                 configtest.bat     setclasspath.sh  tomcat-native.tar.gz
catalina.bat                  configtest.sh      shutdown.bat     tool-wrapper.bat
catalina.sh                   daemon.sh          shutdown.sh      tool-wrapper.sh
catalina-tasks.xml            digest.bat         startup.bat      version.bat
commons-daemon.jar            digest.sh          startup.sh       version.sh
commons-daemon-native.tar.gz  setclasspath.bat   tomcat-juli.jar
```

bin目录

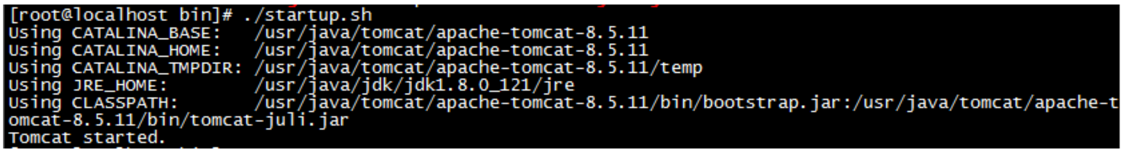
4. 编辑 setclasspath.sh 脚本，添加以下内容：
   a. export JAVA_HOME=/usr/java/jdk/jdk1.8.0_121
   b. export JRE_HOME=/usr/java/jdk/jdk1.8.0_121/jre



```
export JAVA_HOME=/usr/java/jdk/jdk1.8.0_121

export JRE_HOME=/usr/java/jdk/jdk1.8.0_121/jre
```

添加内容

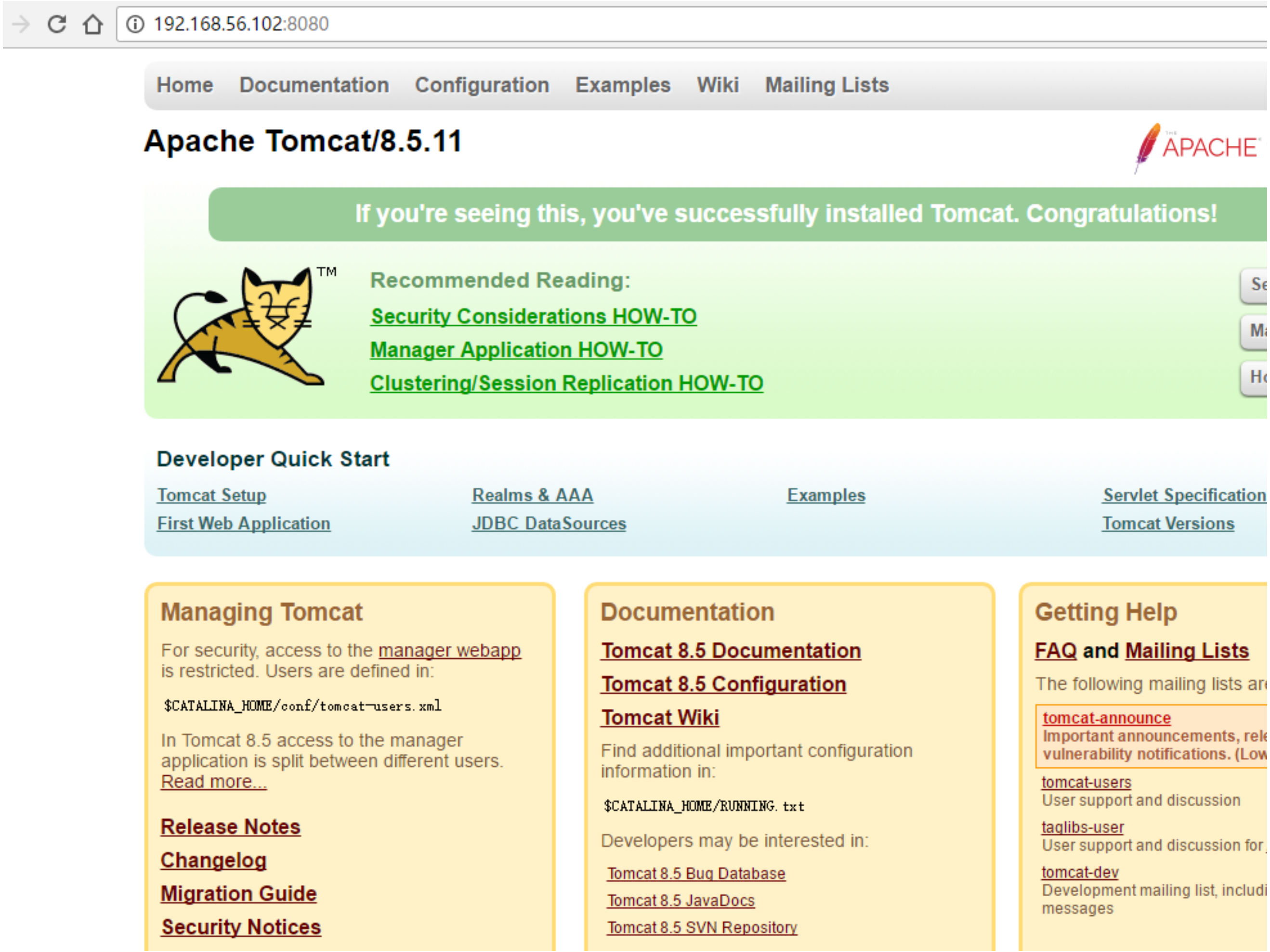
5. 保存后输入以下命令启动 tomcat。
   a. [root@localhost bin]# ./startup.sh



```
[root@localhost bin]# ./startup.sh
Using CATALINA_BASE:   /usr/java/tomcat/apache-tomcat-8.5.11
Using CATALINA_HOME:   /usr/java/tomcat/apache-tomcat-8.5.11
Using CATALINA_TMPDIR: /usr/java/tomcat/apache-tomcat-8.5.11/temp
Using JRE_HOME:        /usr/java/jdk/jdk1.8.0_121/jre
Using CLASSPATH:       /usr/java/tomcat/apache-tomcat-8.5.11/bin/bootstrap.jar:/usr/java/tomcat/apache-t
omcat-8.5.11/bin/tomcat-juli.jar
Tomcat started.
```

启动

6. 从浏览器访问，输入http://ip:8080 即可访问，如果外部无法访问，但是 Linux 内部却可以访问，这种情况一般是防火墙的问题，关闭防火墙就可以了。



192.168.56.102:8080

Apache Tomcat/8.5.11

If you're seeing this, you've successfully installed Tomcat. Congratulations!

Recommended Reading:

Security Considerations HOW-TO

Manager Application HOW-TO

Clustering/Session Replication HOW-TO

Developer Quick Start

Tomcat Setup
First Web Application
Realms & AAA
JDBC DataSources
Examples
Servlet Specification
Tomcat Versions

Managing Tomcat

For security, access to the manager webapp is restricted. Users are defined in:

$CATALINA_HOME/conf/tomcat-users.xml

In Tomcat 8.5 access to the manager application is split between different users. Read more...

Release Notes
Changelog
Migration Guide
Security Notices

Documentation

Tomcat 8.5 Documentation
Tomcat 8.5 Configuration
Tomcat Wiki

Find additional important configuration information in:

$CATALINA_HOME/RUNNING.txt

Developers may be interested in:

Tomcat 8.5 Bug Database
Tomcat 8.5 JavaDocs
Tomcat 8.5 SVN Repository

Getting Help

FAQ and Mailing Lists

The following mailing lists are

tomcat-announce
Important announcements, rele
vulnerability notifications. (Low

tomcat-users
User support and discussion

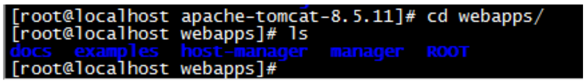
taglibs-user
User support and discussion for

tomcat-dev
Development mailing list, includi
messages

访问

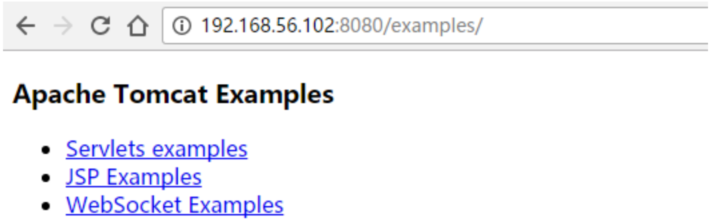
7. 安装好 Tomcat 后，在 webapps 下面自带一个 demo，通过浏览器访问即可看到已经部署好的 web 项目。

    a. [root@localhost apache-tomcat-8.5.11]# cd webapps/



```
[root@localhost apache-tomcat-8.5.11]# cd webapps/
[root@localhost webapps]# ls
docs  examples  host-manager  manager  ROOT
[root@localhost webapps]#
```

访问



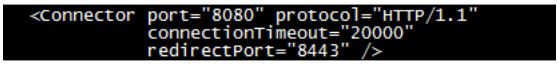
192.168.56.102:8080/examples/

Apache Tomcat Examples

- Servlets examples
- JSP Examples
- WebSocket Examples

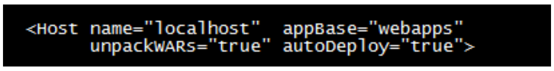
访问

8. 部署项目只需要把项目复制到 Tomcat 的 Webapps 下即可。这也是最简单的一种方式。如果您想更改您的 tomcat 服务接口，那么可以到 tomcat 下的 conf 目录，里面有个 server.xml 文件。重要的配置信息在这里：



```
 <Connector port="8080" protocol="HTTP/1.1"
            connectionTimeout="20000"
            redirectPort="8443" />
```

配置



```
 <Host name="localhost"  appBase="webapps"
       unpackWARs="true" autoDeploy="true">
```

配置

9. 在配置文件中有一个 unpackWARs 参数，如果设置为 true 即可以部署 war 包项目，tomcat 会自动解压文件。还可以在 server.xml 中使用 Context 标签指定项目在任何位置。

    a. <Context path="定web应用的虚拟路径名" docBase="要部署的Web应用的源路径（实际位置）" reloadable="true" />

安装nginx

Nginx是一个小巧而高效的Linux下的Web服务器软件，是由 Igor Sysoev 为俄罗斯访问量第二的 Rambler.ru 站点开发的，已经在一些俄罗斯的大型网站上运行多年，目前很多国内外的门户网站、行业网站也都在是使用Nginx，相当稳定。

1、添加运行nginx服务进程的用户

```
# groupadd -r nginx
# useradd -r -g nginx  nginx
```

2、下载源码包解压编译。

```
# wget http://nginx.org/download/nginx-1.10.2.tar.gz
# tar xvf nginx-1.10.2.tar.gz -C /usr/local/src
# yum groupinstall "Development tools"
# yum -y install gcc wget gcc-c++ automake autoconf libtool libxml2-devel libxslt-devel perl-devel perl-ExtUtils-Embed pcre-devel openssl-devel
# cd /usr/local/src/nginx-1.10.2
# ./configure \
--prefix=/usr/local/nginx \
--sbin-path=/usr/sbin/nginx \
--conf-path=/etc/nginx/nginx.conf \
--error-log-path=/var/log/nginx/error.log \
--http-log-path=/var/log/nginx/access.log \
--pid-path=/var/run/nginx.pid \
--lock-path=/var/run/nginx.lock \
--http-client-body-temp-path=/var/tmp/nginx/client \
--http-proxy-temp-path=/var/tmp/nginx/proxy \
--http-fastcgi-temp-path=/var/tmp/nginx/fcgi \
--http-uwsgi-temp-path=/var/tmp/nginx/uwsgi \
--http-scgi-temp-path=/var/tmp/nginx/scgi \
--user=nginx \
--group=nginx \
--with-pcre \
--with-http_v2_module \
--with-http_ssl_module \
--with-http_realip_module \
--with-http_addition_module \
--with-http_sub_module \
--with-http_dav_module \
--with-http_flv_module \
--with-http_mp4_module \
--with-http_gunzip_module \
--with-http_gzip_static_module \
--with-http_random_index_module \
--with-http_secure_link_module \
--with-http_stub_status_module \
--with-http_auth_request_module \
--with-mail \
--with-mail_ssl_module \
--with-file-aio \
--with-ipv6 \
--with-http_v2_module \
--with-threads \
--with-stream \
--with-stream_ssl_module
# make && make install
# mkdir -pv /var/tmp/nginx/client
```

3、添加SysV启动脚本。

```
# vim /etc/init.d/nginx
#!/bin/sh
#
```

```
# nginx - this script starts and stops the nginx daemon
#
# chkconfig:   - 85 15
# description: Nginx is an HTTP(S) server, HTTP(S) reverse \
#               proxy and IMAP/POP3 proxy server
# processname: nginx
# config:      /etc/nginx/nginx.conf
# config:      /etc/sysconfig/nginx
# pidfile:     /var/run/nginx.pid
# Source function library.
. /etc/rc.d/init.d/functions
# Source networking configuration.
. /etc/sysconfig/network
# Check that networking is up.
[ "$NETWORKING" = "no" ] && exit 0
nginx="/usr/sbin/nginx"
prog=$(basename $nginx)
NGINX_CONF_FILE="/etc/nginx/nginx.conf"
[ -f /etc/sysconfig/nginx ] && . /etc/sysconfig/nginx
lockfile=/var/lock/subsys/nginx
start() {
    [ -x $nginx ] || exit 5
    [ -f $NGINX_CONF_FILE ] || exit 6
    echo -n $"Starting $prog: "
    daemon $nginx -c $NGINX_CONF_FILE
    retval=$?
    echo
    [ $retval -eq 0 ] && touch $lockfile
    return $retval
}
stop() {
    echo -n $"Stopping $prog: "
    killproc $prog -QUIT
    retval=$?
    echo
    [ $retval -eq 0 ] && rm -f $lockfile
    return $retval
killall -9 nginx
}
restart() {
    configtest || return $?
    stop
    sleep 1
    start
}
reload() {
    configtest || return $?
    echo -n $"Reloading $prog: "
    killproc $nginx -HUP
RETVAL=$?
    echo
}
force_reload() {
    restart
}
configtest() {
$nginx -t -c $NGINX_CONF_FILE
}
rh_status() {
    status $prog
}
rh_status_q() {
    rh_status >/dev/null 2>&1
}
case "$1" in
    start)
        rh_status_q && exit 0
    $1
        ;;
    stop)
        rh_status_q || exit 0
        $1
        ;;
    restart|configtest)
        $1
        ;;
    reload)
        rh_status_q || exit 7
        $1
        ;;
    force-reload)
        force_reload
        ;;
    status)
        rh_status
        ;;
    condrestart|try-restart)
        rh_status_q || exit 0
            ;;
    *)
      echo $"Usage: $0 {start|stop|status|restart|condrestart|try-restart|reload|force-reload|configtest}"
        exit 2
esac
```

4、赋予脚本执行权限。

```
# chmod +x /etc/init.d/nginx
```

5、添加至服务管理列表，设置开机自启。

```
# chkconfig --add nginx
# chkconfig  nginx on
```

6、启动服务。

```
# service nginx start
```