



请输入搜索条件...

搜索 高级搜索

热门搜索: JQuery 插件 Struts Spring Hibernate 数据库 Linux Android Iphone 设计模式

[首页](#) [编程语言](#) [Web开发](#) [前端技术](#) [移动开发](#) [数据库](#) [操作系统](#) [嵌入开发](#) [网络安全](#) [办公软件](#) [全部列表>>](#)

今天是: 2013年04月23日 Tue 网站状态: 本站已备案, 备案号(粤ICP备12030714)

Ubuntu 10.04 (64位) 下载并编译 Android 2.2 源码[只有11条命令]

时间: 2012-12-08 02:17:03.0

分类: [首页](#) / [风向标](#) / [热门文章](#)

Ubuntu 10.04 (64位) 下载并编译 Android 2.2 源码[只有11条命令]

为了方便, 我把所有操作写成了命令, 按顺序(软件安装—源码下载—编译)执行成功即可:

```
1. sudo apt-get install git-core gnupg flex bison gperf build-essential zip curl zlib1g-dev gcc-multilib g++-multilib
libncurses5-dev ia32-libs x11proto-core-dev
libx11-dev lib32readline5-dev lib32z-dev java-common unixodbc
2. sudo apt-get install sun-java6-jdk
3. cd
4. mkdir -p software/bin
5. curl http://android.git.kernel.org/repo >software/bin/repo
6. chmod a+x software/bin/repo
7. mkdir -p software/android/src/new
8. cd software/android/src/new
9. repo init -u git://android.git.kernel.org/platform/manifest.git -b froyo
10. repo sync
11. make -j4
```

1. 前言

本文叙述了如何从零开始搭建基于Google官方Android 2.2.1 (froyo) 源码的开发环境, 包括如何下载源代码、编译源代码, 以及在模拟器中运行编译生成的镜像。文中开发机是基于Ubuntu 10.04的, 同样适用于其他版本的Ubuntu。现在需要弄清楚一下几点, 防止后面出现误操作: 下载源代码是用参数: `repo init -u git://android.git.kernel.org/platform/manifest.git -b froyo` (下载的是2.2.1版本, 编译环境操作系统必须是32位的, jdk1.5版本的——jdk1.5没有64位版本)。如果是没有带参数的下载源代码, 将会下载所有的源代码, 编译环境要求: 操作系统64位, jdk1.6, (偶是2010年10月15日做) 但很有可能会出现交叉编译工具错误: “/bin/bash: prebuilt/linux-x86/toolchain/arm-eabi-4.4.0/bin/arm-eabi-gcc: 没有那个文件或目录” (实际上是有arm-eabi-gcc这个文件的, 偶怀疑是版本问题——32位的, 而环境要求是64位的—— 下载全部代码)

2. 下载JDK 1.5 (不建议此下载方式, 防止配置环境参数出错。建议用apt-get install方式, 请跳转到: 第4步)

编译Android源码需要JDK1.5版(主要是需要其中的javadoc), 1.6版是不兼容的。JDK1.5已经停止支持, 其最终的版本是JDK 5.0 Update 22。

可从Sun官网上下载:

<http://java.sun.com/javase/downloads/5u22/jdk>

选择平台Linux, 勾选同意许可协议:

单击Continue, 在弹出的页面中一共有两个下载, 选第一个jdk-1_5_0_22-linux-i586.bin下载:

3. 部署JDK 1.5

3.1 解压缩JDK

以下命令均在root账户上执行，非root账户执行某些命令可能需要加sudo。

将jdk-1_5_0_22-linux-i586.bin放在root目录下。进入该目录：

```
cd ~
```

然后赋予该bin文件可执行属性：

```
chmod +x jdk-1_5_0_22-linux-i586.bin
```

再执行安装(注意前面有./)：

```
./jdk-1_5_0_22-linux-i586.bin
```

跳过协议阅读。

最后一行：

```
Do you agree to the above license terms? [yes or no]
```

键入yes，然后回车，开始解压。默认会解压到当前目录下的jdk1.5.0_22目录下（此处是~/jdk1.5.0_22）。

3.2 JDK版本切换

如果同时需要使用JDK1.6版做应用开发，则系统中将同时存在JDK1.5和JDK1.6。因此需要实现能够在不同版本间切换。如果不使用JDK1.6，请做完1.5版的设置后跳至3.3节。

这里使用update-alternatives命令实现。

首先为JDK1.5版生成可选配置选项：

```
update-alternatives --install /usr/bin/jdk jdk /root/jdk1.5.22 255
```

```
(sudo update-alternatives --install /usr/bin/jdk jdk /home/ynb/jdk1.5.0_22 255)
```

/usr/bin/jdk是符号链接，它指向/etc/alternatives/jdk（由第2个参数给出的），/root/jdk1.5.22(/home/ynb/jdk1.5.0_22)是实际的链接，255是优先级（任意整数，在automatic mode自动模式时，数字大的优先级较高，被选为默认配置选项）。

然后再为JDK1.6版生成（如果有JDK1.6的话。不使用JDK1.6的请跳至3.3节）：

```
update-alternatives --install /usr/bin/jdk jdk /root/jdk1.6.20 0
```

这样，默认jdk版本即为1.5版。需要切换到1.6版时，请使用update-alternatives -config jdk

3.3 配置JDK环境变量

下面开始配置环境变量。

在终端中执行：sudo gedit ~/.bashrc

修改.bashrc文件，添加下面语句：

```
#set java environment
```

```
export JAVA_HOME=/usr/bin/jdk
```

```
export JRE_HOME=JAVA_HOME/jre
```

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

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

该段代码最好设置在文件的尾部。

保存文件。在终端中执行source ~/.bashrc

重启系统请使用java -version检查JDK是否配置正确。

4. 用apt-get install方式安装JDK1.5:

Android 要求的 jdk 1.5, 而Ubuntu10.4的源里只有JDK1.6。如果安装了jdk1.6请先卸载：

```
$sudo apt-get remove remove openjdk-6-jdk
```

```
$sudo apt-get remove remove openjdk-6-jre
```

```
$sudo apt-get remove remove openjdk-6-jre-headless
```

安装jdk1.5目前网上解决方法有两种:

解决方法一(建议): 安装JDK1.5

- 1) 根据官方文档里所说, 源代码的编译必须使用JDK5才能编译的了, 所以这里偶使用jdk5
- 2) 需要先更新源, 在终端里执行 `sudo vim /etc/apt/sources.list`
- 3) 在source.list里把以下2行拷贝到文档最后面:

```
deb http://us.archive.ubuntu.com/ubuntu/ jaunty multiverse
```

```
deb-src http://us.archive.ubuntu.com/ubuntu/ jaunty multiverse
```

```
deb http://us.archive.ubuntu.com/ubuntu/ jaunty-updates multiverse
```

```
deb-src http://us.archive.ubuntu.com/ubuntu/ jaunty-updates multiverse
```

保存退出。

- 4) 执行 `sudo apt-get update` 更新源.
- 5) 配置编译环境并下载JDK5: `sudo apt-get install sun-java5-jdk`

解决方法二: 安装JDK 1.6 , 修改Android源文件。

- 1) 安装jdk 1.6, 修改<src_top/build/core/make.mk>(如果2.0版本源码请修改<src_top/build/core/main.mk>)

在文件中搜索并分别把:

```
java_version := $(shell java -version 2>&1 | head -n 1 | grep '[ "]1.5[. "$$"]')
```

```
javac_version := $(shell javac -version 2>&1 | head -n 1 | grep '[ "]1.5[. "$$"]')
```

两行改成:

```
java_version := $(shell java -version 2>&1 | head -n 1 | grep '[ "]1.6[. "$$"]')
```

```
javac_version := $(shell javac -version 2>&1 | head -n 1 | grep '[ "]1.6[. "$$"]')
```

即可。

推荐使用第一种解决方法。

5. 安装其他必备组件

在终端上, 使用下面的命令安装: (前面的1、2、3步骤都可以省略啦)

```
sudo apt-get install git-core gnupg flex bison gperf libssl-dev libbsd0-dev libxgtk2.6-dev build-essential zip curl  
libncurses5-dev zlib1g-dev
```

因为JDK1.5在前面已经单独安装过了, 因此此处apt-get install命令行中没有加入sun-java5-jdk。

6. 下载Android 源代码

6.1 安装repo脚本

首先安装repo。在当前用户: ~目 录下新建一个bin目录。在终端中执行:

```
cd ~
```

```
mkdir bin
```

然后, 向PATH中添加bin路 径:

```
export PATH=$PATH:~/bin
```

然后执行下面两行指令, 下载repo脚本并修改可执行属性:

```
curl https://android.git.kernel.org/repo >~/bin/repo
```

```
chmod a+x ~/bin/repo
```

6.2 初始化repo

首先建立一个目录用来存放Android源代码:

```
mkdir mydroid
```

```
cd mydroid
```

然后用最新的列表执行初始化。这里偶下载android 2.2.1(froyo)，因此加了参数 -b froyo: (android 2.2.2(froyo)源码下载:2G多

```
repo init -u git://android.git.kernel.org/platform/manifest.git -b froyo )
```

(下载2.1_r2源码

```
repo init -u git://android.git.kernel.org/platform/manifest.git -b android-2.1_r2
```

下载全部源码: repo init -u git://android.git.kernel.org/platform/manifest.git)

(下载2.0源码:

```
repo init -u git://android.git.kernel.org/platform/manifest.git -b eclair )
```

(上面下载2.0时有个莫名其妙的问題:

```
ynb@ynb-pc:~/bin$ repo init -u git://android.git.kernel.org/platform/manifest.git -b eclair
```

```
Usage: repo init -u url [options]
```

出现上面问题，删除mydroid目录: rmdir mydroid ; mkdir mydroid2.0 ; cd mydroid2.0;

再: repo init -u git://android.git.kernel.org/platform/manifest.git -b eclair 就可以了)

(如果能通过浏览器浏览到android源码的主页，但运行repo init或repo sync的时候，却没法下载源码，形如:

只需要把git开头的地址改为http即可:

[1]

```
repo init -u http://android.git.kernel.org/platform/manifest.git -b froyo
```

[2]修改manifest.git里的地址，也是把git://改为http://

```
vim .repo/manifest.git
```

如:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<manifest>
```

```
<remote name="korg"
```

```
fetch="http://android.git.kernel.org/"
```

```
review="review.source.android.com" />
```

```
<default revision="froyo"
```

```
remote="korg" /> )
```

这中间会问到你的名字和Email。如果你打算上传代码，Email须是一个Google账户。

repo初始化成功后会看到类似于下面的提示:

```
repo initialized in /home/(当前用户名)/mydroid
```

最后执行\$repo sync开始同步代码。过程比较漫长，下载速度取决于你的网速 (偶的3-4小时左右)，中间有可能遇到断线，重新执行repo sync即可断点续传。

7. 编译Android源代码

7.1 编译生成镜像

切换到android源码根目录下，执行\$sudo make即可。首次编译非常耗时，取决于机器配置，大约需要2-3小时。有可能会出现：“Permission denied (权限拒绝的提示)” 请命令赋予权限:

\$cd 无权限文件的当前目录 回车; \$sudo chmod -R +rwx * 回车; 完成后镜像生成在/root/mydroid/out/target/product/generic下: android源码编译后得

到system.img,ramdisk.img,userdata.img映像文件。其中， ramdisk.img是emulator的文件系统，system.img包括了主要的包、库等文件，userdata.img包括了一些用户数

据，emulator加载这3个映像文件后，会把 system和 userdata分别加载到 ramdisk文件系统系统中的system和 userdata目录下。

7.2 编译、配置Android SDK

make编译完成后，执行\$sudo make sdk命令，生成对应于该版本源代码的sdk，此过程用的时间偶的是10多分钟。注意这个sdk不能用于生产环境。

用于生产环境的最好用如下命令: \$sudo make PRODUCT-sdk-sdk命令，生成对应于该版本源代码的用于生产环境的sdk，此过程用的时间偶的是1个多小时。

Android官方的SDK和偶自己编译Android源码时使用make sdk编译出的SDK有区别，模拟器版本不同，加载的默认镜像也不同 (一个是官方的， 一个是自己编译的)。下面将展示如何能够自由切换SDK版本。

首先是自己编译的SDK版本，实际位置是/root/mydroid/out/host/linux-x86/sdk/android-sdk_eng.root_linux-x86。

因为它比较常用，我们给它高优先级:

```
update-alternatives --install /usr/bin/AndroidSDK AndroidSDK /root/mydroid/out/host/linux-x86/sdk/android-sdk_eng.root_linux-x86 255
```

事先下载的官方SDK位置是/root/android-sdk-linux_86，执行:

```
update-alternatives --install /usr/bin/AndroidSDK AndroidSDK /root/android-sdk-linux_86 0
```

然后使用update-alternatives --display AndroidSDK查看当前配置情况:

如果要切换配置，使用update-alternatives --config AndroidSDK

配置AndroidSDK环境变量。终端中执行gedit ~/.bashrc

在文件最后添加下面三行：

```
# set android environment

export ANDROID_SDK_HOME=/usr/bin/ AndroidSDK

export PATH=$ANDROID_SDK_HOME/tools:$PATH
```

保存文件。在终端中执行source ~/.bashrc

8. 在模拟器中运行编译的镜像

打开终端，执行android脚本：

```
$android
```

出现下图所示窗口：

选择左边第一项Virtual Devices，然后在右边选择New，新建一个AVD。

Name：AVD的名称，随便取，但只能包含字母和数字以及点、下划线和连字符， 这里取名test。

Traget：目标SDK版本，这里选自己编译的2.1-r1版Android SDK。

SD Card：SD卡。暂且不填，待需要时再设置。

Skin：皮肤（模拟器屏幕分辨率）。

Hardware：使用默认即可

然后点Create AVD，就创建好了一个AVD。点击Start即可启动模拟器，其运行的是自己编译的镜像。

若以后要再次启动此AVD，请在终端中运行：

```
emulator -avd test 或者 emulator @test
```

评论：0

1099人已阅读

相关文章	
Ubuntu 10.04（64位）下载并编译 Android 2.2 源码[只有11条命令]【转】	Ubuntu 10.04（64位）上载并编译 Android 2.2 源码[只有11条命令]【转】
Ubuntu 12.04（64位）下载并编译 Android 4.1 源码[只有2条命令]	Ubuntu 12.04（64位）下载并编译 Android 4.1 源码
32位Ubuntu 11.10下android2.3.7源码下载与编译小结	32位Ubuntu 11.10下android2.3.7源码下载与编译小结
Ubuntu10.10(64位)编译Android2.3源码 遇到的错误	Ubuntu10.10(64位)编译Android2.3源码 遇到的异常
ubuntu10.04下编译android2.3源码	ubuntu10.04停编译android2.3源码
32位Ubuntu 10.10下android2.3.7源码下载及编译	32位Ubuntu 10.10停android2.3.7源码下载及编译
Ubuntu 10.04系统 32位机上安装编译Android2.3源码	ubuntu 11.10（32位）停android2.2 源码编译
Ubuntu 11.10[64位] 编译Android源码	Ubuntu 10.10（64位）编译Android 2.3
Ubuntu10.10（64位）编译Android2.3	（64位）ubuntu 11.10编译android 2.3预备的工作
（64位）ubuntu 11.10编译android 2.3准备的工作	ubuntu 11 64 android 2 编译 java
android2.2源码编译—Ubuntu10.04（X86）+ android2.2 + JDK1.6（转）	【Android】Android 4.2源码下载（ubuntu 12.10）
Ubuntu 10.04 下Android 源码下载与编译	ubuntu 10.04和10.10上下载编译环境配置android2.2-froyo

💬 文章评论 以下网友留言只代表其个人观点，不代表本网站的观点和立场。

目前还没有留言，还不快抢！

💬 参与留言 网名和内容都不能为空。

* 网名:

* 内容:

* 验证码:

提交

版权声明:本站大部分内容均为原创内容, 部分文章摘自互联网, 如果侵犯了您的版权, 请联系管理员删除 (联系邮箱: 421584397@qq. com)
Copyright © 2012, WWW. VERYDEMO. COM, All Rights Reserved | 粤ICP备12030714