

# Windows 系统下 buildozer 工具配置

## 关键词说明:

**Kivy:** kivy 是开源 [Python](#) 函式库，用于开发行动应用程序和其它采用自然用户界面的多点触控应用软件。它可以在 [Android](#), [iOS](#), [Linux](#), [OS X](#) 和 [Windows](#) 执行。采用 [MIT](#) 授权条款，Kivy 是自由并且开源的软件——摘自《百度百科》

**Buildozer:** Buildozer 是一款旨在轻松打包移动应用程序的工具。它自动化整个构建过程，下载 python-for-android, Android SDK, NDK 等先决条件。Buildozer 在您的应用程序目录中管理一个名为 buildozer.spec 的文件，描述您的应用程序要求和设置，如标题，图标，包含的模块等。它将使用规范文件为 Android, iOS 等创建一个安装包。-----译自 Buildozer 日志网站 <https://buildozer.readthedocs.io/en/latest/>

不幸的是，buildozer 工具只工作于 linux 系统，所以在 Windows 系统就需要安装 linux 虚拟环境。我试过 ubuntu18.04 桌面版，但是可能由于我的系统配置太低，总感觉整个配置过程特别慢而且感觉笨重，在这里推荐使用 xubuntu，去官网就可以下载，使用起来相对更加轻便。（如果电脑上已安装 ubuntu 虚拟机可仍然按照此教程进行配置）

## 目的:

在 Windows 系统下的 xubuntu 虚拟机上安装 buildozer 工具，以及其他相关工具，将 kivy (python) 程序打包成可以在手机上安装的 apk 文件。

## 配置环境:

Windows 10 下虚拟机 xubuntu18.04

## 工作流程:

1、安装 vitural Box，下载 xubuntu18.04 桌面版（官网）。

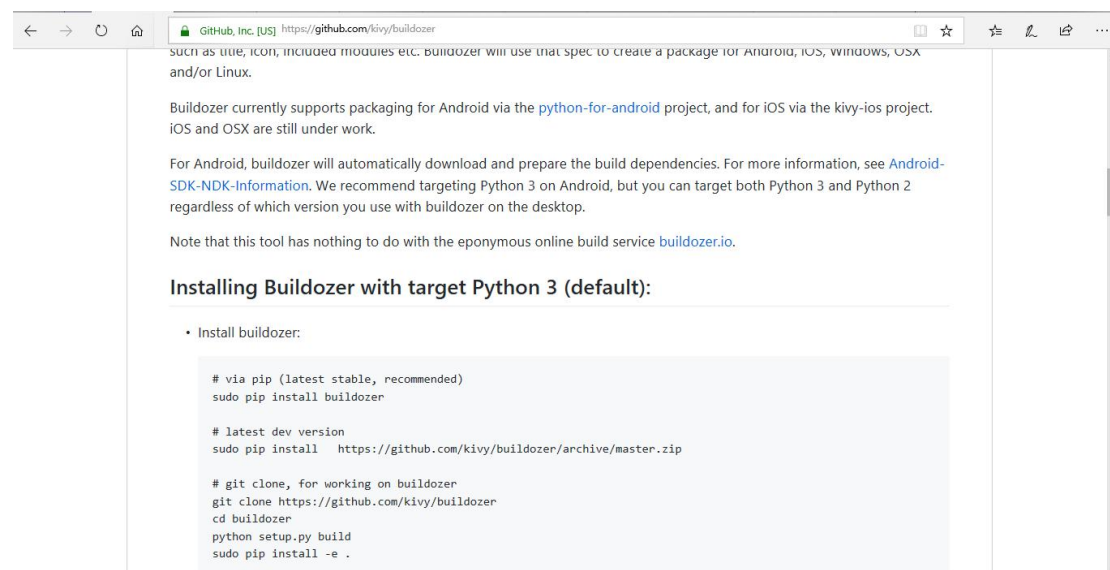
在虚拟机管理器 vitural Box 中安装 xubuntu 以及设置主机系统和虚拟机系统共享文件夹可参考 <https://blog.csdn.net/u012732259/article/details/70172704>这篇博文。

2、点击这个网址 <https://github.com/kivy/buildozer> 如图，按照网页中的要求在终端中输入相关指令，建议先执行到截图中的指令（即执行到 `sudo pip3 install -e .`）

- 由于需要 `pip` 工具，我们在终端中输入 `sudo apt-get install python3-pip` 下载 `pip3` 工具，而且我们之后用到的所有涉及 `pip` 指令都必须相应替换为 `pip3`，`python` 指令也必须相应替换为 `python3`

- 由于需要 `git` 工具，我们在终端中输入 `sudo apt-get install git` 下载 `git` 工具

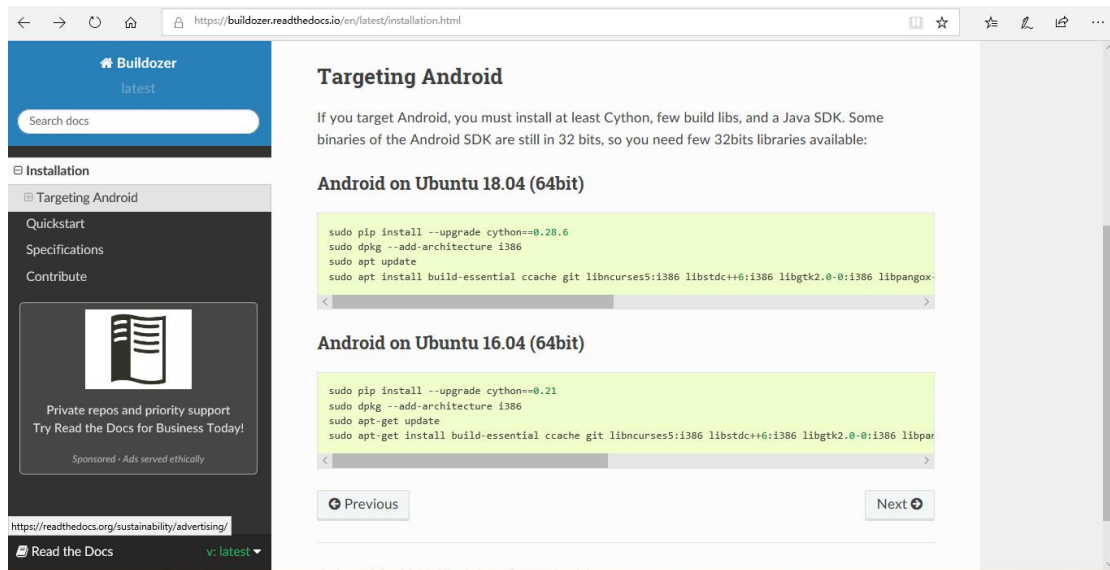
**Note:** #后的语句是解释性语句，不用管。而且最后一条命令最后有一个句点”。“要注意。



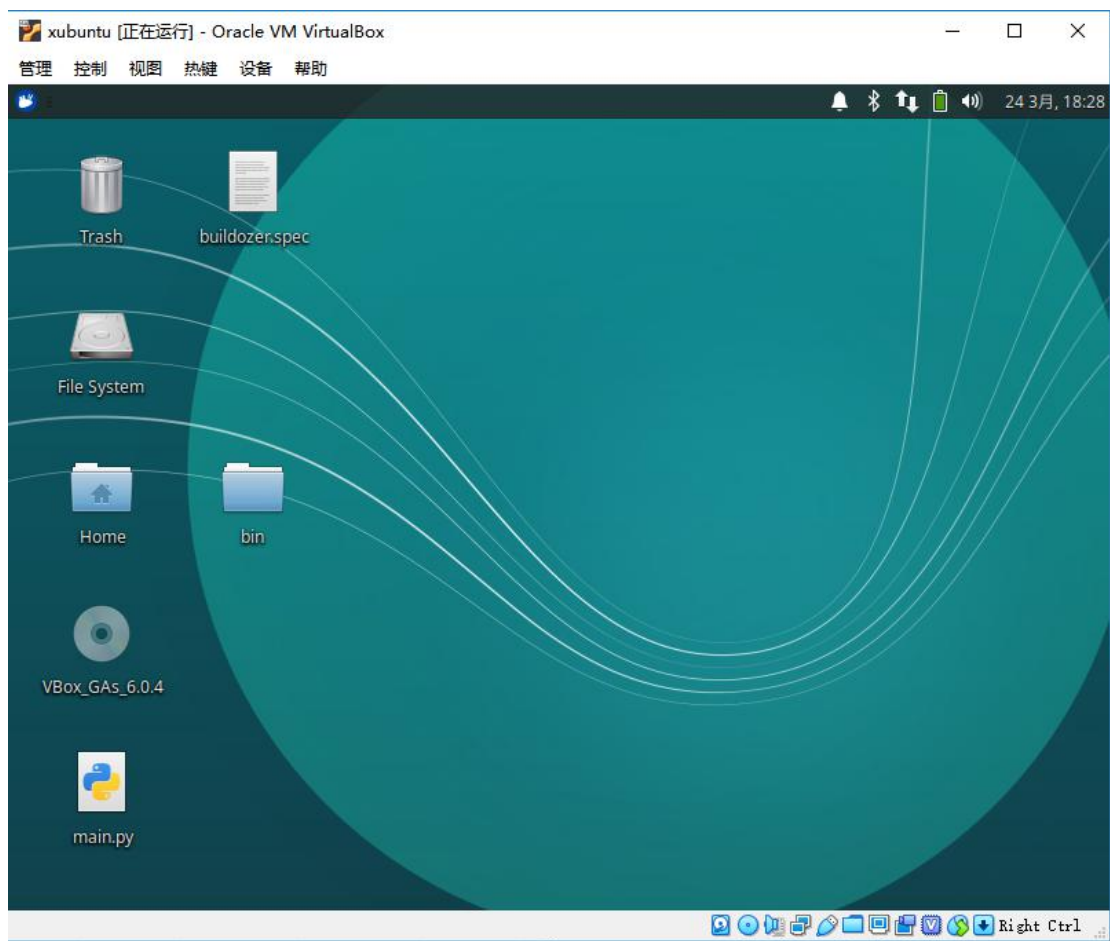
3、接下来我们点击 <https://buildozer.readthedocs.io/en/latest/installation.html> 这个网址如下图，按照 18.04 板块的命令，在终端中一一输入进行执行，耐心等待，直到所有文件下载成功。

**Note:**

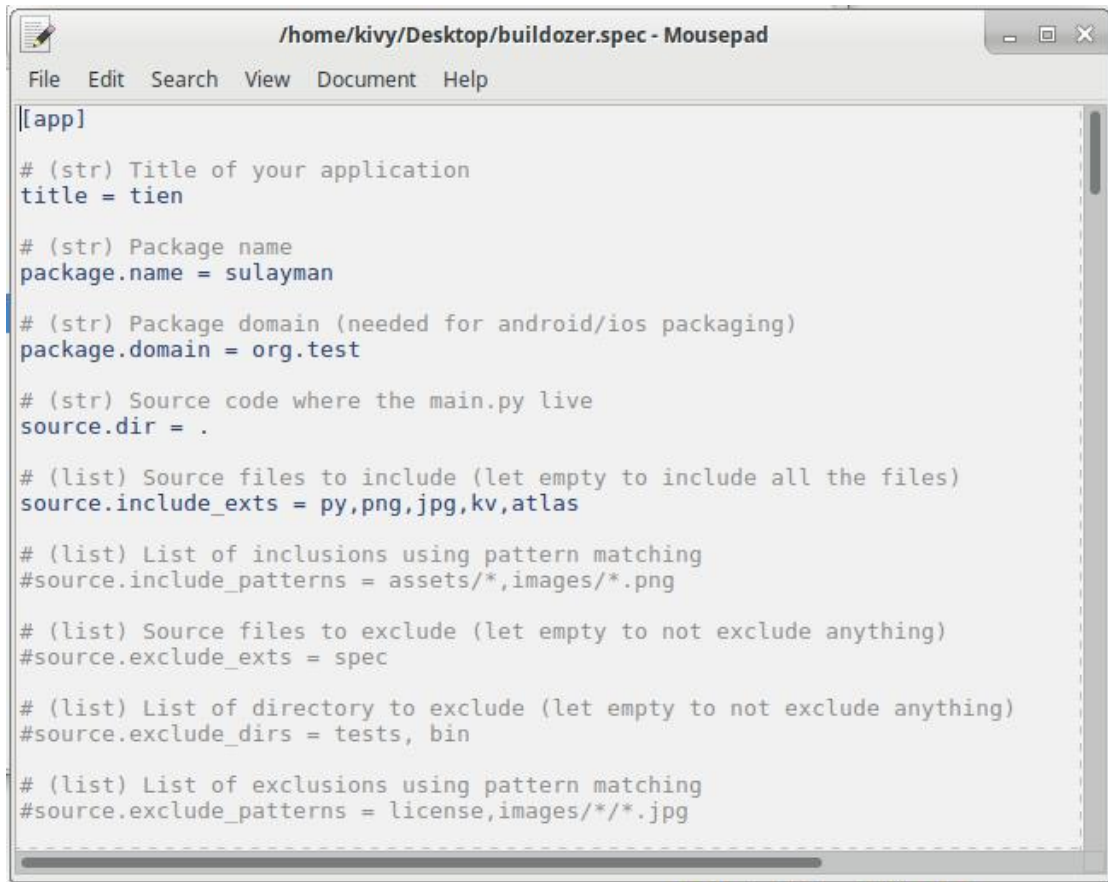
在下载工程中会遇到协议部分，但是协议后没有给出提示，只有光秃秃的光标，这时候我们输入“y”同意然后按【enter】继续即可



4、接下来我们到/mnt/share/路径下将共享文件夹里的 kivy 工程文件（也可只是简单的.py 文件）复制到桌面，在桌面打开终端，输入 `buildozer init` 命令，这时会在桌面生成 `buildozer.spec` 文件



5、打开 `buildozer.spec` 文件，基本不用改什么，应用或者安装包名字什么的随自己改。



```
[app]

# (str) Title of your application
title = tien

# (str) Package name
package.name = sulayman

# (str) Package domain (needed for android/ios packaging)
package.domain = org.test

# (str) Source code where the main.py live
source.dir = .

# (list) Source files to include (let empty to include all the files)
source.include_exts = py,png,jpg,kv,atlas

# (list) List of inclusions using pattern matching
#source.include_patterns = assets/*,images/*.png

# (list) Source files to exclude (let empty to not exclude anything)
#source.exclude_exts = spec

# (list) List of directory to exclude (let empty to not exclude anything)
#source.exclude_dirs = tests, bin

# (list) List of exclusions using pattern matching
#source.exclude_patterns = license,images/*/*.jpg
```

- 那个 `source.dir` 需要注意，`" = ."`这是刚刚我们放在桌面上的文件 `main.py` 的路径。
- `log_level` 这项参数是配置日志。可设置为 `1` 或 `2`，其中在设为 `2` 的情况下，配置日志更加详细，更容易发现错误，故在此我们一般设为 `2`。

6、返回终端，输入 `buildozer android debug` 生成 `apk` 文件，第一次会下载很多东西。接下来系统自动下载相关工具和依赖（主要有 `ANT,SDK,NDK,hostpython3` 和一些其他我也不知道的东西），这个时间会持续很久，要保持耐心，直到如下图所示【**Android packaging done**】说明大功告成，我们可以在路径下的 `bin` 文件夹查看 `apk` 文件。

```
Terminal - kivy@kivyvm: ~/Desktop
File Edit View Terminal Tabs Help

droid/dist/sulayman/private/lib/python2.7/site-packages/android/broadcast.pyo
assets/private.mp3: /home/kivy/Desktop/.buildozer/android/platform/python-for-an
droid/dist/sulayman/private/lib/python2.7/site-packages/android/_android_sound.s
o
assets/private.mp3: /home/kivy/Desktop/.buildozer/android/platform/python-for-an
droid/dist/sulayman/private/lib/python2.7/site-packages/android/billing.pyo
assets/private.mp3: /home/kivy/Desktop/.buildozer/android/platform/python-for-an
droid/dist/sulayman/private/lib/python2.7/site-packages/android/mixer.pyo
assets/private.mp3: /home/kivy/Desktop/.buildozer/android/platform/python-for-an
droid/dist/sulayman/private/lib/python2.7/site-packages/android/runnable.pyo
assets/private.mp3: /home/kivy/Desktop/.buildozer/android/platform/python-for-an
droid/dist/sulayman/private/lib/python2.7/site-packages/android/_android_billing
.so
assets/private.mp3: /home/kivy/Desktop/.buildozer/android/platform/python-for-an
droid/dist/sulayman/private/lib/python2.7/site-packages/android/_init_.pyo
assets/private.mp3: /home/kivy/Desktop/.buildozer/android/platform/python-for-an
droid/dist/sulayman/private/lib/python2.7/site-packages/android/_android.so
assets/private.mp3: /home/kivy/Desktop/.buildozer/android/platform/python-for-an
droid/dist/sulayman/private/include/python2.7/pyconfig.h
assets/private.mp3: /home/kivy/Desktop/.buildozer/android/app/main.pyo
assets/private.mp3: /home/kivy/Desktop/.buildozer/android/app/sitecustomize.pyo
# Android packaging done!
# APK file 1.0 debug.apk available in the bin directory
kivy@kivyvm:~/Desktop$
```

Note: 这期间会有一次错误（关键字：autoconf, pkg-config, automake, libtoolize），原因是如下图：

```
[INFO]: Picking the latest gcc toolchain, here 4.9
[WARNING]: Missing executable: pkg-config is not installed
[WARNING]: Missing executable: autoconf is not installed
[WARNING]: Missing executable: automake is not installed
[WARNING]: Missing executable: libtoolize is not installed
[INFO]: No existing dists meet the given requirements!
```

所以我们依次在终端输入：sudo apt-get install autoconf 等待下载完成

sudo apt-get install pkg-config 等待下载完成

sudo apt-get install automake 等待下载完成

sudo apt-get install libtool 等待下载完成

然后，在终端重新输入 buildozer android debug 直到最后终端提示【android packaging done】

至此，整个配置打包工作完成。

Written by TSJ  
2019.3.24

