**CTS User Guide**

**For Ubuntu**

1. 环境配置

确保ubuntu PC上面配置有 ADB 和 AAPT

ADB(Android Debug Bridge)就是安卓的环境在搭建好以后，应该就有了的

AAPT(Android Asset Packaging Tool)是要单独安装并配置的：

1/ 下载安装包：

<https://ibotpeaches.github.io/Apktool/>

apktool\_2.3.0.jar

2/

cp apktool\_2.3.0.jar /usr/local/bin/

chmod 777 /usr/local/bin/apktool\_2.3.0.jar

3/ 如果系统是64位的，则无法直接执行，因为aapt需要32位支持，因此需要先安装32位库

sudo apt-get install lib32z1

sudo apt-get install lib32stdc++6

执行过程中如有报错，则需要先执行一下：

sudo apt-get update

4/ 验证aapt是否已经配置完毕

aapt version

1. 如果要在Android 7.0及以上版本运行，则需要安装和配置jdk 8.0

1/

sudo apt-get install openjdk-8-jre

2/ 如果一台PC上同时安装有多个版本的java的话，则需要手动指定要使用的JAVA版本

sudo update-alternatives --display java

sudo update-alternatives --display javac

3/ 手动切换JAVA版本

sudo update-alternatives --config java

sudo update-alternatives --config javac

这两个要对应的切换，不然是不正确的配置

1. 下载CTS工具包，放在ubuntu下的任一路径下

<https://source.android.com/compatibility/cts/downloads>

如果要运行在手机上的话，选择 (CTS) – ARM即可，然后选择对应的手机的安卓系统版本即可

1. 解压以后，

Chmod 777 -R android-cts-7.0\_r16-linux\_x86-arm

cd android-cts-7.0\_r16-linux\_x86-arm/android-cts/tools/

./cts-tradefed

这样就启动了cts控制台，进入Cts命令行 cts-tf >

退出指令为 exit

1. 所有模块全部测试一遍的命令：

run cts –plan Java

1. 查看某个模块

list packages

list plans

1. 指定某个模块中的某个测试并运行

run cts -m Gesture --test android.gesture.cts.GestureTest#testGetStrokes

不清楚命令的情况下可以直接执行-help

-m : 要测试的模块，可通过 l m 命令查看都有哪些模块

--test : 要测试的测试名称

\*\*#\*\* : 类名#方法名 ， 要测试的类和方法名称

1. 生成的测试报告

自动保存在/android-cts-7.0\_r16-linux\_x86-arm/android-cts/下，会同时生成两个文件夹：

/logs/

/results/

/results/下的test\_result.xml就是最终生成的报告

这份报告可以用chrome或者Firefox浏览器打开，前提是必须将整个以时间命名的test\_result文件夹拷贝出来并在这个里面才能打开

因为这个里面涉及到一个.xml文件的xsl格式，这个xsl格式就类似于你在打开那个excel表格的时候所使用的格式，这个格式由于是谷歌自定义的，因此，必须要在当前文件夹中加载

这个格式在test\_result文件夹中定义，所以必须全部整个拷贝出来才能够打开

1. cts检测camera计划部分的命令为：

run cts-camera

1. CTS在运行某个Case的过程中，可以指定其运行所需的ABI，在64位操作系统中，可以指定， --force-abi 32|64， 含义为仅对32位或者64位ABI进行测试。

在运行某个测试case的过程中，可以使用如下命令强制测试在某个ABI下运行

--abi <abi\_name>

默认情况下，测试会为每个该设备所支持的ABI运行一遍

CTS单独测试某一个Testcase时候，不能只是指出要测试的test，还必须指定该测试Case所属的模块

**Building and running CTS**

* Installing Repo
  + mkdir ~/bin
  + export PATH=~/bin:$PATH
  + Download repo script to ~/bin/repo refer to

curl http://commondatastorage.googleapis.com/git-repo-downloads/repo > ~/bin/repo

* + chmod a+x ~/bin/repo
* mkdir WORKINGDIR
* cd WORKINGDIR
* repo init -u https://android.googlesource.com/platform/manifest -b android-cts-7.0\_r16
* repo sync -j8
* . build/envsetup.sh
* lunch aosp\_arm-eng
* make cts -j32

Build result is saved under folder:

out/host/linux-x86/cts/android-cts.zip

And this file is automatically unzipped under the same folder.

Detail complying instructions are based on :

<https://source.android.com/setup/downloading>

1. repo这个工具就是个谷歌自己搞的git的脚本，用于在谷歌的代码库中去下或者传安卓的源代码的
2. 如果要切换repo分支，可以执行如下命令

在AOSP的目录下，查看可以切换的所有分支

cd .repo/manifests

git branch -a | cut -d / -f 3

切换分支

repo init -u [https://android.googlesource.com/platform/manifest -b android-7.0.0\_r1](https://android.googlesource.com/platform/manifest%20-b%20android-7.0.0_r1)

同步代码

repo sync

如果这时候有如下提示的话：

build/: discarding 1 commits

dalvik/: discarding 2 commits

kernel/: discarding 6 commits

packages/apps/Calendar/: discarding 1 commits

packages/apps/Contacts/: discarding 2 commits

packages/apps/Mms/: discarding 1 commits

packages/apps/Music/: discarding 1 commits

packages/apps/Phone/: discarding 1 commits

vendor/embinux/support-tools/: discarding 1 commits

那么就要执行如下命令：

repo forall -c git reset --hard

repo init -u https://android.googlesource.com/platform/manifest -b android-7.0.0\_r1

repo sync