TTAutoVerify 的设计和实现

1. 准备知识

RD 在处理TT bug的时候，通常在收到bug的mail通知是开始处理，然后check in code，接下来就要对TT bug 做VerifyBugClosed 给SQ 验证。而verifybugclose的时候需要填一些信息：

1. RD developer
2. Fixed in version
3. Action for result(包含root cause和solution)

其中1和2 是必填的，3的部最好填gerrit的change link 可以给大家review。于是问题就来了，fixed in version，这一栏改怎么填？这就牵涉的到bug的种类：

1. AMAXQ bug
2. ProjectQ bug

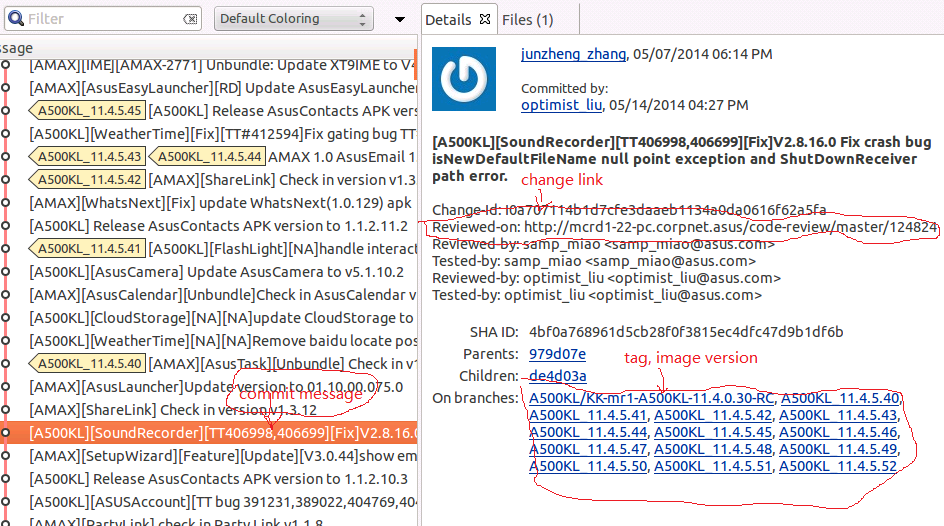
其中AMAXQ bug比较单纯，直接填apk的version 就可以，而projectQ的bug 比较复杂，除了apk version之外，image version是必填项。不管怎样，apk version 都必须是gerrit上已经merged 的change。就引发我们要解决的三个问题：

1. Change 通过Git push 之后，怎样得到gerrit的change link？

2. 怎样知道gerrit change 是否被merge？

3. 怎样知道 project的image 已经release？

其实这些都是可以从git log中找到答案。



好，有了这些准备知识，我们就可以开始做TTAutoVerify 的tool了。

1. 目的

因为RD 可以收到TT 开出来的bug，change commit 后什么时候merge和image release 的schedule基本上都不能及时把握，所以很多bug不能及时的verify，对RD，PM，还有SQ都是不小的人力浪费。所以TTAutoVerify 的tool就是要帮RD 来及时的verify bug，节省不必要的人力成本而提高效率。

最朴素的想法：开发一个script，每天自动运行，根据User 在TT上的name和password，使用web api 模拟http request来自动verify TT bug。只要运行一次，解bug的时候只需要在gerrit上填上规范的commit message，剩下的TT update 等琐事，这个tool 会自动帮你搞定。

流程这样的：



接下来就是设计和实现了。

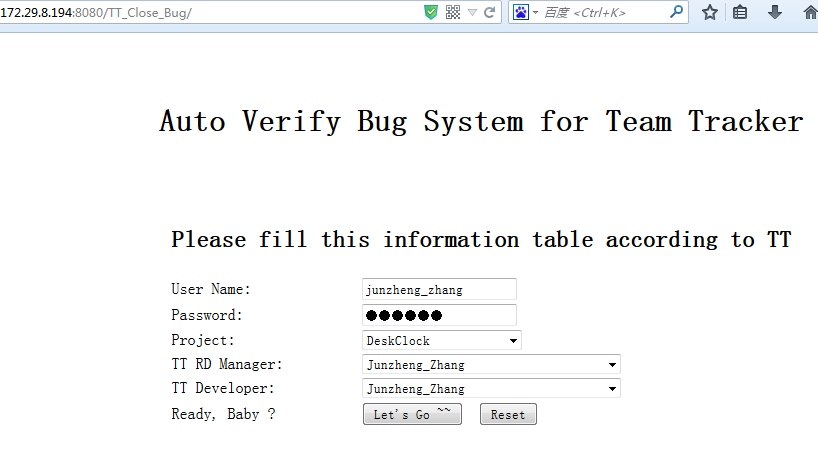
1. 详细设计和实现
2. 开发和运行平台：linux
3. 使用语言： perl，j2ee的hibernate 和struts2，jsp。
4. 使用的数据库： mysql
5. Website:  **http://172.29.8.194:8080/TT\_Close\_Bug/**
6. Commit message format：

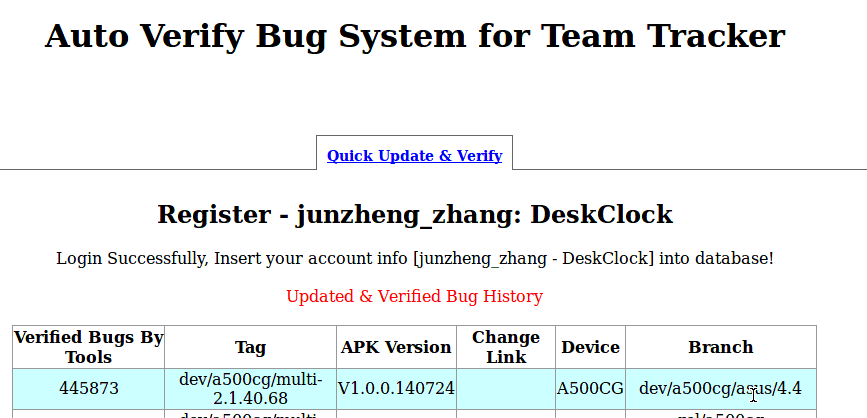
**[device\_name][project\_name][TT\_xxxxxx,xxxxxx][fix] release note:XXXX**

多个TT number 使用逗号分隔。

分为两个部分来实现整个逻辑：

1. Web 前端，提供user 登录的page，根据User 在TT上的username和password，以及负责的project name，显示tool 自动关闭的bug 列表。





1. 后端自动verify，根据前端的信息，对每个user负责的project，对每个git repository 进行git log的parse，整理verify bug 所需的字段，然后使用http api auto verify,流程如下：

