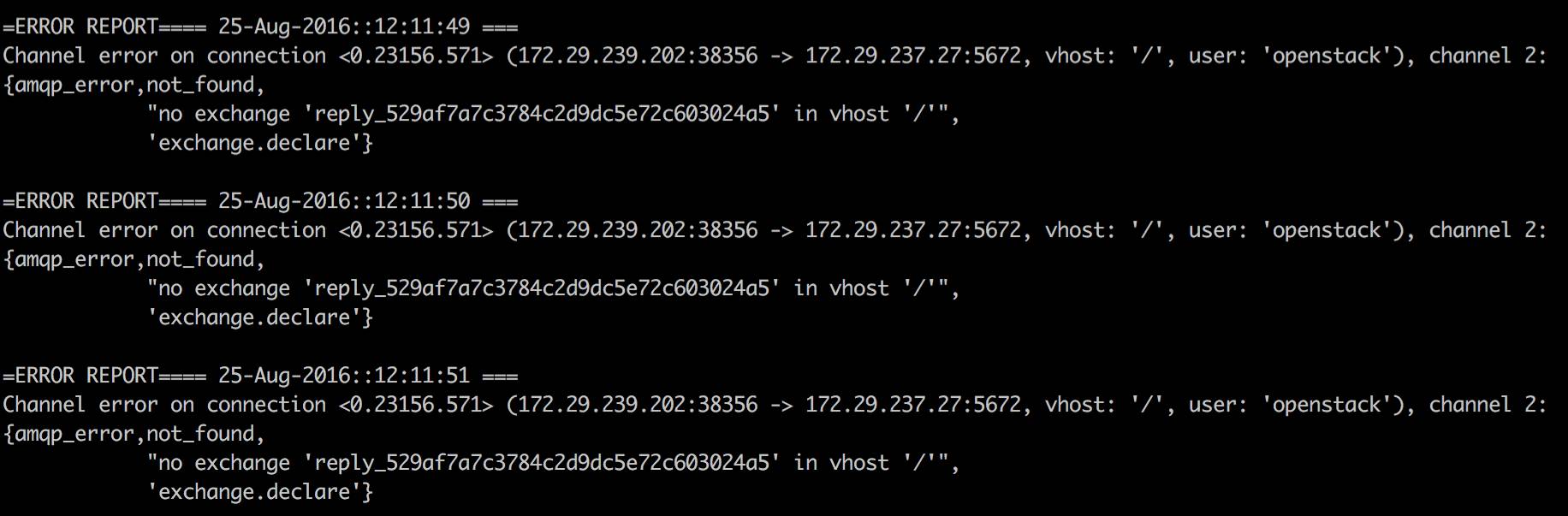
这是 OpenStack 实施经验分享系列的第 10 篇。  
  
是软件就会有 bug，OpenStack 也不例外，只要用它就一定会遇到故障。Troubleshooting（故障排除）是运维 OpenStack 等开源项目的重要技能，遇到问题后一定要**借助社区的力量**定位、搜索、分析并解决问题。  
  
下面 CloudMan 将分享一个真实的案例，还原当时 Troubleshooting 的过程，希望能给大家一些启发。

**问题描述**

某天客户的 OpenStack 突然全线瘫痪：任何操作都无法正常完成，一直处于正在执行状态，界面上也不报错，就是无法完成操作。

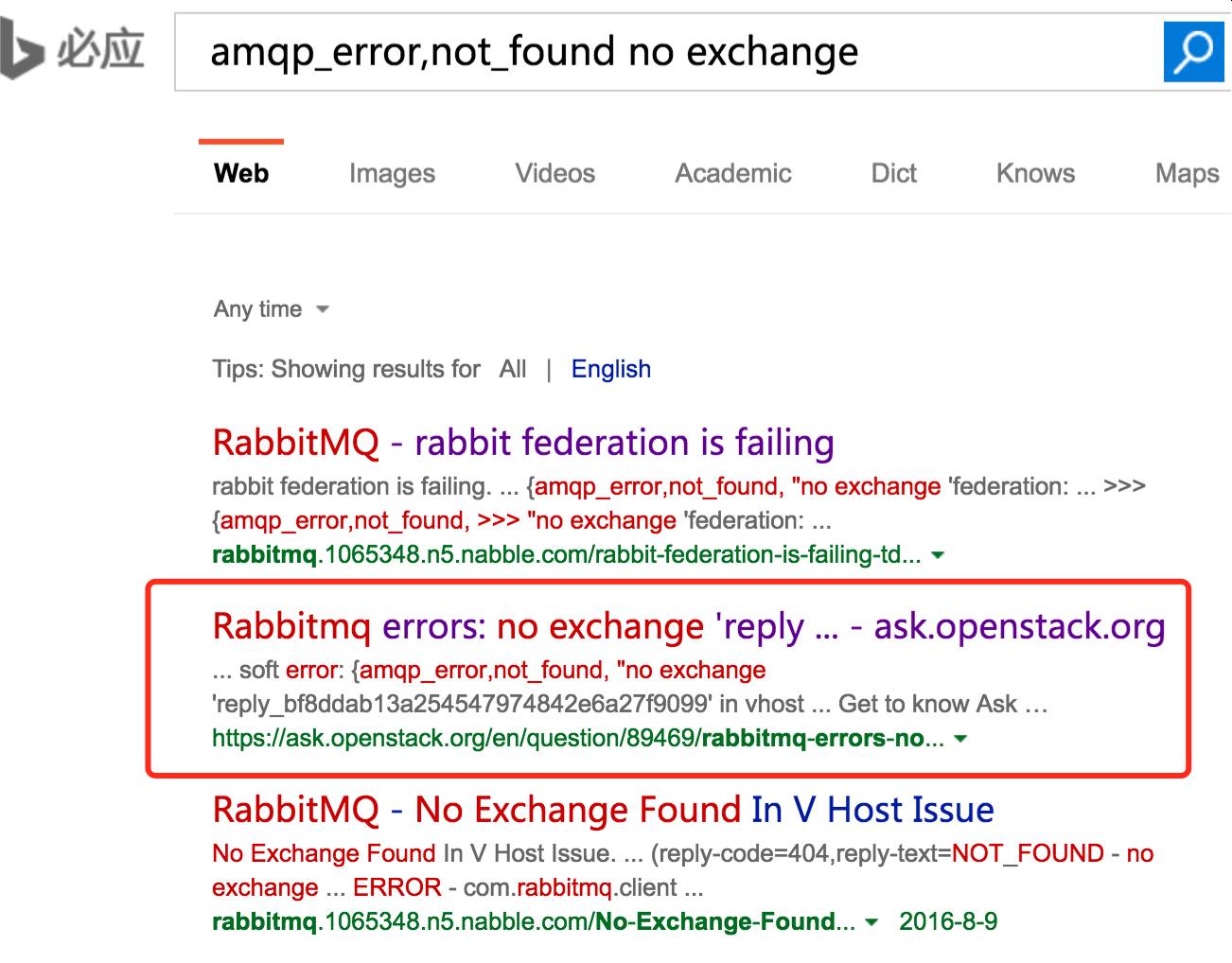
**问题分析**

这是一个全局性的问题，首先查看 nova 日志，无报错，再看 MySQL 和 RabbitMQ 日志，在 RabbitMQ 中发现大量重复报错：

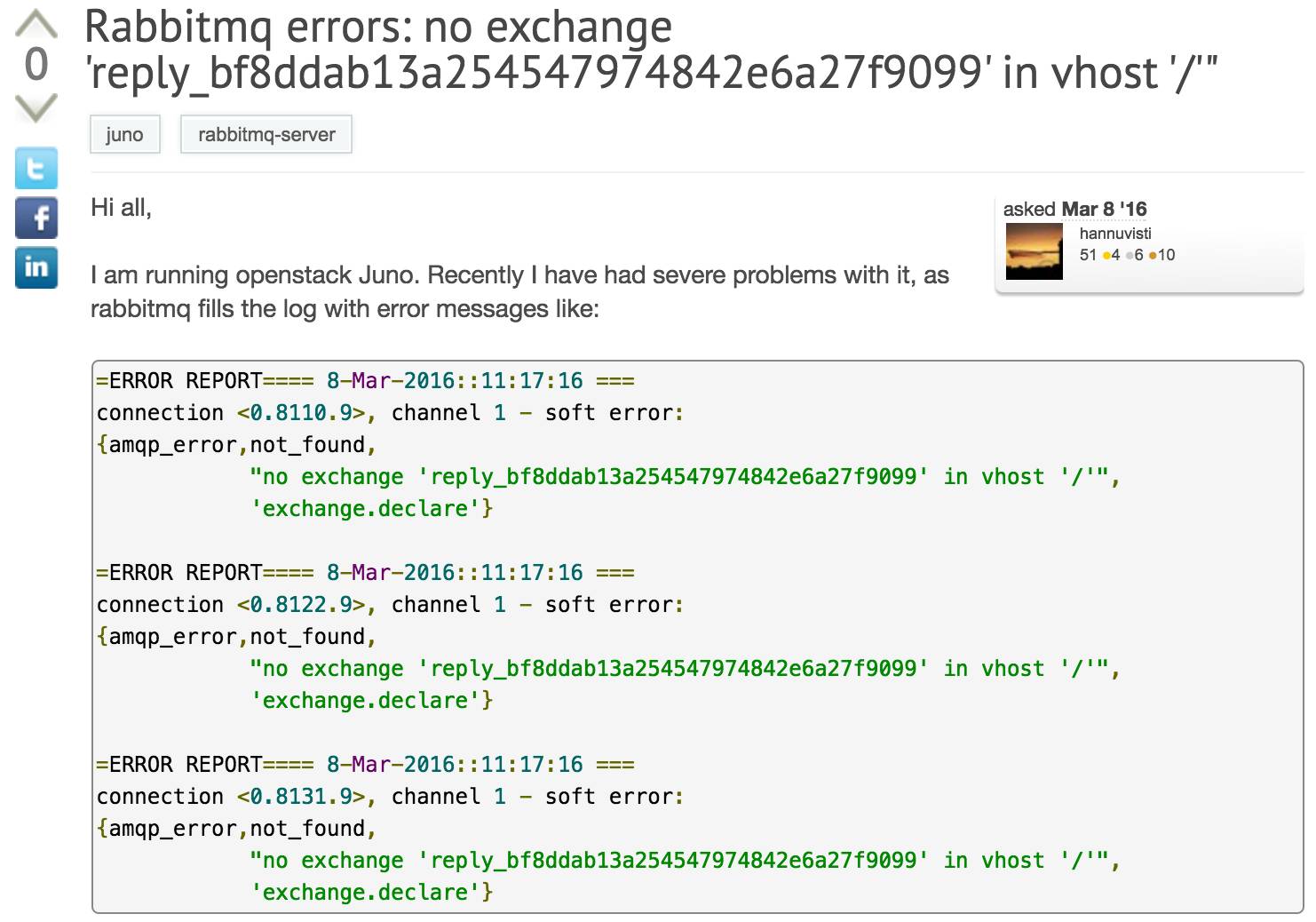


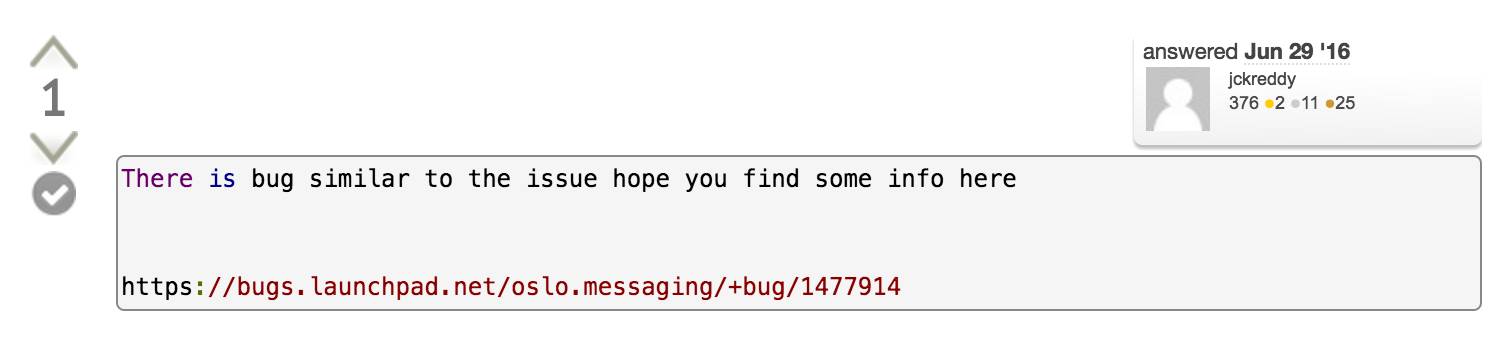
一直报 reply\_529af7a7c3784c2d9dc5e72c603024a5 这个 exchange 找不到。 这些 reply\_XXX 的都是 OpenStack 自己维护的，之前运行得好好的，为什么突然找不到，应该是发生了异常，跟配置没有关系，估计是 bug。  
  
先 google 一下吧。搜索技术问题，google 是首选，翻不了墙就用 bing，度娘嘛还是让她专注中文吧 :-)

这里贴出 bing 的搜索结果：

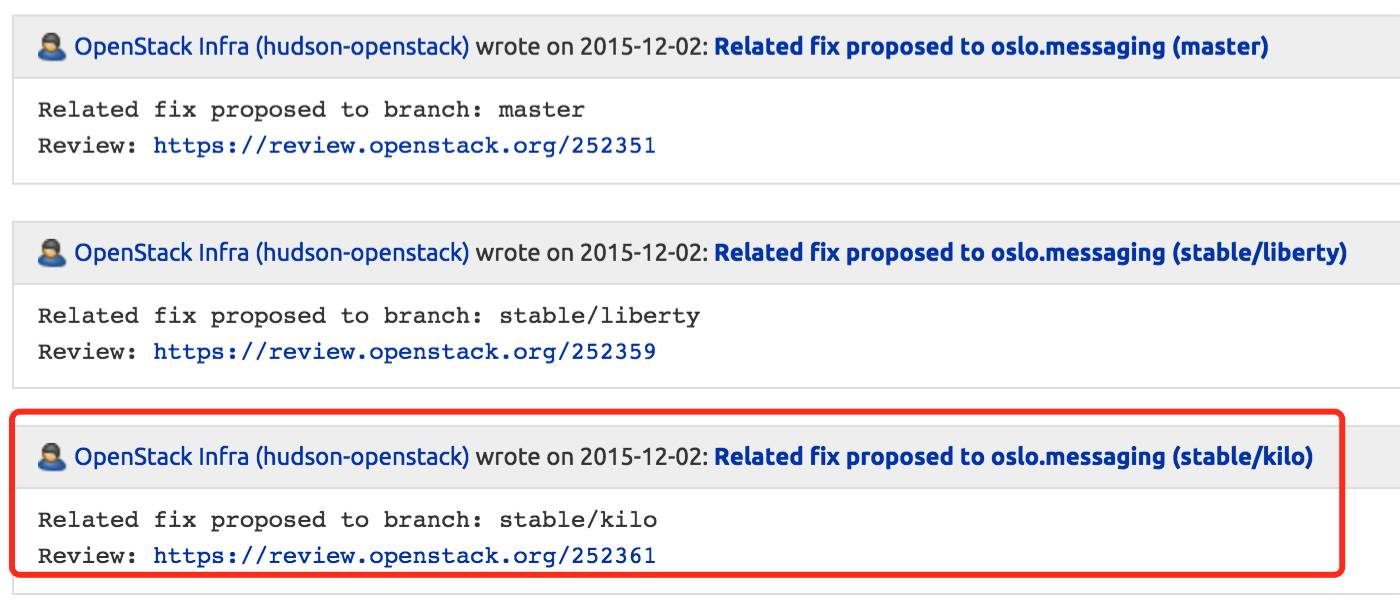


看上去第二个比较靠谱，点进去发现跟我们的情况完全一样，而且还提到一个相关 bug。

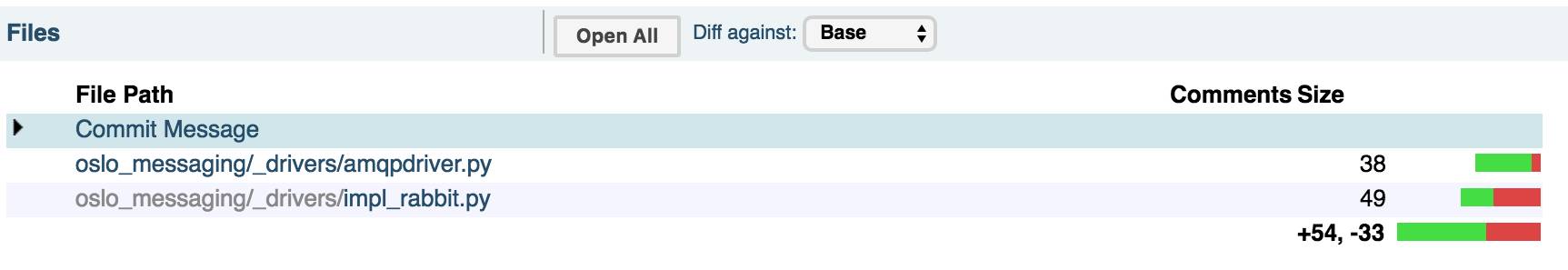




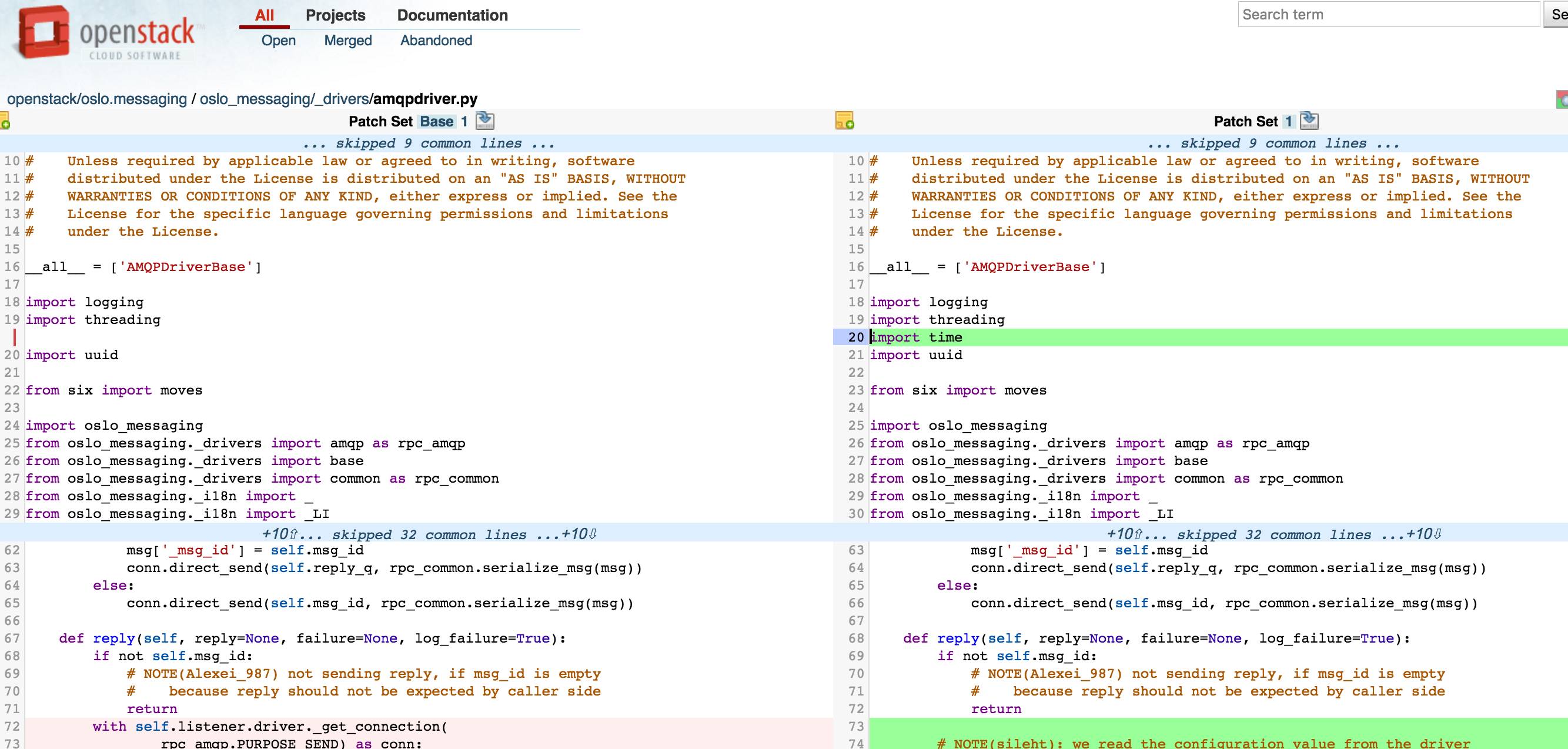
浏览一下 bug 的内容，确实是我们遇到的问题，这是一个 oslo.messaging 的 bug，而且已经 fix 了。



因为客户 OpenStack 版本是 kilo, 所以点击 kilo 对应的 review 链接看看 fix 都修改了哪些地方。



一共改了两个文件，点开 amqpdriver.py 的链接，可以看到 diff。



对比客户系统 /usr/local/lib/python2.7/dist-packages/oslo\_messaging/\_drivers/amqpdriver.py 文件内容，确实是 fix 之前的版本。

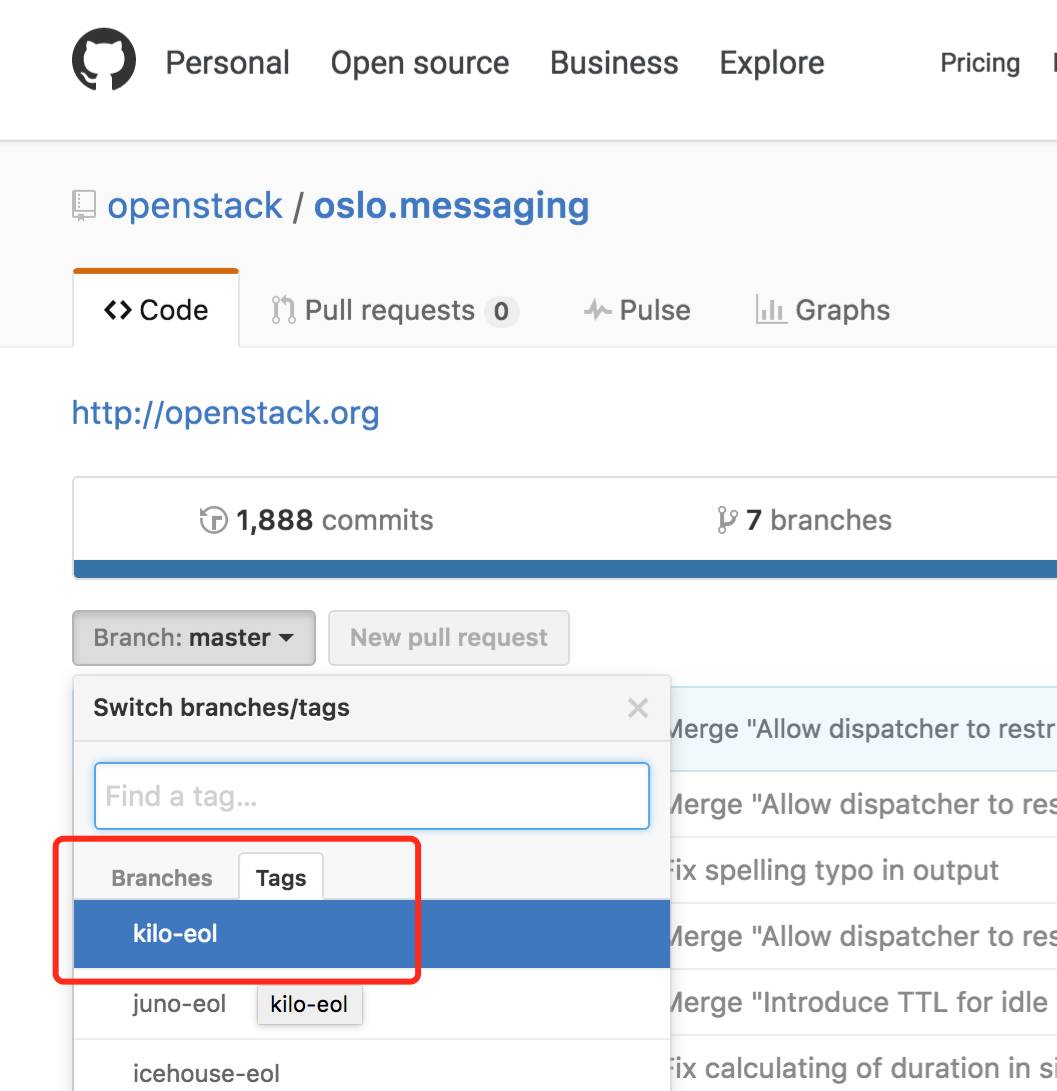
问题确定了，解决办法也有了：**更新 olso.messageing 包**。

**解决问题**

OpenStack 的源代码是在 github 上维护的，每个模块有自己的 repository。 oslo.messageing 的项目主页是 <https://github.com/openstack/oslo.messaging>

因为我们目前的版本是 kilo，所以要找 oslo.messaging 在 kilo 上的最新版本。

在 Tags 中，我们看到有 kilo-eol，eol 的意思是 “end of life”，是 kilo 的最终版本了。



可以再次确认，kilo-eol 确实包含了我们想要的 fix。后面的工作就很直接了：

1. 下载 oslo.messaging 代码库。
2. 安装 kilo-eol 版本。
3. 重启相关 OpenStack 相关服务。

下节我们会详细讨论如何更新 OpenStack 组件。

由于 oslo.messaging 是基础组件，几乎所有服务都会用到，所以不得不更新每一个节点并重启 OpenStack。工作量虽然大些，但问题终于解决了。