

云风的 BLOG

思绪来得快去得也快，偶尔会在这里停留

[« skynet 下的用户登陆问题](#) | [返回首页](#) | [回调还是消息队列](#) »

Hive 增加了 socket 库

按计划给 [Hive](#) 增加了非阻塞的 socket 库。这样，它就可以用 lua 完成 skynet 中的 gate 以及其它 tcp 连接相关的功能了。

我比较纠结的是 listen 这个 api 的设计。传统的 bind/listen/accept 模型不太适合这样的 actor 模式。一个尝试过的方案是 skynet 中的 gate。也就是在 listen 的端口上每收到一个新连接，就转发到新的 cell 中去。

为了灵活起见，我把转发控制交给了回调函数。这也是用 lua 的灵活之处。

用户可以选择启动新的 cell，然后把新连接 forward 到新 cell 去，也可以 fork 一个 coroutine，forward 回自己处理。整套 socket api 是非阻塞的，如果 IO 上没有数据，coroutine 都会被挂起。

这次 Hive 这个项目算是业余的尝试，skynet 在我们的项目中已经积累了太多相关代码，不太容易迁移到新的框架下来。而且用 lua 实现大部分以前用 C 实现的代码，性能损失未知。不过这几天写下来觉得，C 没有很好的 coroutine 支持，在做异步消息处理时非常麻烦，以至于用 C 写的模块都不能很好的提供服务的同时保持代码简洁。用 Lua 重新实现它们要简单的多。

这个 socket 库我顺手把 epoll / kqueue / select 三种模型都支持了。其中 select 实现起来最麻烦，而且性能不高。所以，如果想在 Windows 跑的同学，姑且认为它仅仅是想运行起来罢了。生产环境还是不要用 windows 的好。

为什么不用现成的 lua socket 库呢？

因为对于这种多虚拟机的用法，现有的 lua 库接口都不太合适。我需要在 VM 中处理 socket 上的消息，并把消息传递到其它 VM 中进一步处理。所以把网络数据打包为 lua 的 string 或 userdata 都不合适。

最好的方法是直接用 malloc 分配一块内存，复制消息，然后以 lightuserdata 指针的形式传递给后续处理的 VM。这样可以避免许多数据拷贝。甚至于可以将消息多次转发，直到有 VM 可以指针处理它为止。

至于 Hive 这个玩具暂时就放在这里了。有兴趣完善它的同学可以和我联系。

比如，可以利用已有的 socket 库增加一个调试协议，允许从外部连入一个控制台，去查询 hive 的内部状态。就像 skynet 的 debug 模型那样工作。

还可以利用 socket 库进一步的处理多个进程的不同 hive 相互交互的工作。

我认为这些 skynet 有的特性，移植到 hive 中都不是太难的事情。

ps. 本来想做一个 rockspec 提交到 luarocks 中去的，做了一半遇到些困难。而且整个代码也还需要经常更新，还是搁置一下吧。

云风 提交于 July 2, 2013 03:08 PM | [固定链接](#)

COMMENTS

skynet 的 debug 模型指的是什么呢？

Posted by: lkp | (4) [July 8, 2013 11:21 PM](#)

coroutine确实是个非常让人眼馋的特性；目前我们公司（非网游行业）还是用c++做服务端，用lambda表达式加消息队列（用function<>对象作为消息）和异步io，基本能满足低开发成本和高性能的需求；如果能有coroutine那就完美了。

Posted by: zcpro | (3) [July 2, 2013 04:54 PM](#)

老是以hive是<http://baike.baidu.com/view/699292.htm>这个

Posted by: wiml | (2) [July 2, 2013 04:47 PM](#)

老是以hive是<http://baike.baidu.com/view/699292.htm>这个

Posted by: wiml | (1) [July 2, 2013 04:47 PM](#)

POST A COMMENT

非这个主题相关的留言请到: [留言本](#)

名字:

Email 地址:

为了验证您是人类, 请将六加一的结果 (阿拉伯数字七) 填写在下面:

URL:

☐ 记住我的信息?

留言:

(不欢迎在留言中粘贴程序代码)

提交