skynet相关笔记

# skynet服务地址、名称、handle

skynet分为lua层与C代码层，在lua层会涉及到服务地址、名称等概念。

例如在lua层：

* 可以使用skynet.self()获取服务自己的址址。
* skynet.register(name) 可以为自己注册一个别名。（别名必须在 16 个字符以内）
* skynet.name(name, address) 为一个地址命名。skynet.name(name, skynet.self()) 和 skynet.register(name) 功能等价

而在C代码层发送消息时是根据handle来路由的，即使传入的目的是一个别名，在C语言中也会被转换成对应的handle。

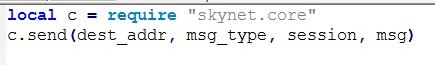
***未完，待续...***

# skynet.core模块API

skynet.core模块使用C语言实现，在lua-skynet.c中实现

## 使用方法

skynet.core模块导出一些API以供Lua中调用，在lua中的使用方法如下所示：



skynet.core模块通过luaopen\_skynet\_core函数导出lua API

## API说明

### send函数

该函数在C语言中的实现对应lsend函数。

send函数的参数解析如下所示：

* 参数1：addr

即目的地址，addr可以为数字或字符串;

如果addr为数字，则addr就是目标服务的handle，即skynet\_context->handle。

如果addr为字符串，***未完，待续...***

* 参数2：msg\_type
* 参数3：session
* 参数4：msg
* 参数5：msg\_len

该参数只在msg的数据类型为LUA\_TLIGHTUSERDATA时传入

### command函数

该函数在C语言的实现对应lcommand函数。

command函数的参数解析如下所示：

* 参数1：cmd

要执行的命令

* 参数2：param

该参数为可选参数，如果要执行的cmd没有参数，则不需要传入该参数。

lua中可以通过command函数执行指定的命令，这些命令通过skynet\_server.c中的cmd\_funcs数组定义命令的名称与执行函数，目前支持的命令如下所示：

* TIMEOUT
* REG

该命令可以为服务注册一个本地名称,即类似”.test”的名称。在lua中可以使用c.command(“REG”, “.test”)调用

该命令可以查询本服务的名称。此时该命令不需要传参数或者传入空字符串作为参数。

* QUERY
* NAME

该命令为一个地址命名，它接收一个参数，参数格式：”name addr“。在lua中可以使用c.command(“NAME”, name .. “ ” .. skynet.address(handle))调用

* EXIT
* KILL
* LAUNCH
* GETENV
* SETENV
* STARTTIME
* ABORT
* MONITOR
* STAT
* LOGON
* LOGOFF
* SIGNAL