Luci页面配置教程

---by llf.2016/8/8

Luci是用来做openwrt页面的，采用lua脚本语言开发。采用MVC架构，只需要写一些lua脚本，剩下的部分openwrt可以自动完成。

1. │   │   └── etc
2. │   │   │   ├── config
3. │   │   │   │  └── 配置文件名称
4. │   │   │   ├── init.d
5. │   │   │  └──执行初始化的脚本
6. │   │   └── usr
7. │   │   └── lib
8. │   │   └── lua
9. │   │   └── luci
10. │   │   ├── controller
11. │   │   │ └── lua脚本名称
12. │   │   ├── model
13. │   │   │ └── cbi
14. │   │   │ └── lua脚本名称
15. │   │   ├──view
16. │   │   └── htm文件名称

如上图所示，需要添加的几个文件已经采用符号标示出来，根据controller调用的处理方法不同及定位的文件路径不同，model和view下的文件需要作相应的调整及删减。具体步骤如下：

**第1步：**

在\usr\lib\lua\luci\controller\下建立一个lua脚本，脚本内容示例如下：

--[[ 模块入口 ]]

-- 程序模块名 /usr/lib/lua/luci/controller/njitclient.lua

1. module("luci.controller.njitclient", package.seeall)
2. function index()
3. -- 添加一个新的模块(Page)入口
4. -- entry(路径, 调用目标, \_("显示名称"), 显示顺序)
5. -- 路径 以字符串数组给定 {"admin", "network", "njitclient"} -> http://ip/cgi-bin/luci/admin/network/njitclient
6. -- 调用目标：
7. -- 执行指定方法（Action）：call("function\_name")
8. -- 访问指定页面（Views）：template("myapp/mymodule") 访问/usr/lib/lua/luci/view/myapp/mymodule.htm
9. -- 调用CBI(Configuration Binding Interface) Module：cbi("myapp/mymodule") 调用/usr/lib/lua/luci/model/cbi/myapp/mymodule.lua
10. -- 重定向：alias("admin", "status", "overview")
11. entry({"admin", "network", "njitclient"}, cbi("njitclient"), \_("NJIT Client"), 100)
12. end

**第2步：**

UCI是openwrt的配置管理机制，将配置统一放到/etc/config文件夹下。示例如下：

1. config login
2. option username ''
3. option password ''
4. option ifname 'eth0'
5. option domain ''

Section开始语法：config ‘类型’ ‘名字’

参数定义语法： option ‘键’ ‘值’

列表定义语法：list ‘集合名字’‘值’

**第3步：**

在controller里采用cbi调用Model文件夹里的lua脚本，所以此步需要在\usr\lib\lua\luci\model\cbi文件夹下建立如下lua脚本

--[[ 配置模块实际的代码 ]]

1. require("luci.sys")
2. -- 映射与存储文件的关系
3. -- m = Map("配置文件文件名", "配置页面标题", "配置页面说明")
4. -- 配置文件文件名：配置文件存储的文件名，不包含路径 /etc/config/xxx
5. m = Map("njitclient", translate("NJIT Client"), translate("Configure NJIT 802.11x client."))
6. -- 配置文件中对应的section
7. s = m:section(TypedSection, "login", "")
8. -- 不允许增加或删除
9. s.addremove = false
10. -- 不显示Section的名称
11. s.anonymous = true
12. -- 配置文件中section的交互(创建Option)
13. -- Value（文本框）
14. -- ListValue（下拉框）
15. -- Flag（选择框）
16. enable = s:option(Flag, "enable", translate("Enable"))
17. name = s:option(Value, "username", translate("Username"))
18. pass = s:option(Value, "password", translate("Password"))
19. pass.password = true
20. domain = s:option(Value, "domain", translate("Domain"))
21. ifname = s:option(ListValue, "ifname", translate("Interfaces"))
22. for k, v in ipairs(luci.sys.net.devices()) do
23. if v ~= "lo" then
24. ifname:value(v)
25. end
26. end
27. -- 判断是否点击了“应用”按钮
28. local apply = luci.http.formvalue("cbi.apply")
29. if apply then
30. io.popen("/etc/init.d/njitclient restart")
31. end
32. return m

**第4步：**

读写配置完成之后，如何根据配置进行操作？编辑/etc/init.d/下相应的文件，并添加执行权限

1. #!/bin/sh /etc/rc.common
2. START=50
3. run\_njit()
4. {
5. local enable
6. # 获取变量值
7. # config\_get\_bool 变量名 Section名 Section参数名
8. config\_get\_bool enable $1 enable
9. if [ $enable ]; then
10. local username
11. local password
12. local domain
13. local ifname
14. # 获取变量值
15. # config\_get 变量名 Section名 Section参数名
16. config\_get username $1 username
17. config\_get password $1 password
18. config\_get domain $1 domain
19. config\_get ifname $1 ifname
20. if [ "$domain" != "" ]; then
21. njit-client $username@$domain $password $ifname &
22. else
23. njit-client $username $password $ifname &
24. fi
25. echo "NJIT Client has started."
26. fi
27. }
28. start()
29. {
30. # 读取配置文件
31. config\_load njitclient
32. # 遍历配置文件中的Section
33. # config\_foreach 遍历函数名 Section类型
34. config\_foreach run\_njit login
35. }
37. stop()
38. {
39. killall njit-client
40. killall udhcpc
41. echo "NJIT Client has stoped."
42. }

**第5步：**

此部分可选，在\usr\lib\lua\luci\view页面下添加相应的htm文件。由于示例程序在controller文件下的lua脚本中并没有使用template访问相应的htm脚本，故此部分可以略去。

以上就是全部的luci添加页面的操作。各步骤不分先后，但每部分之间参数及调用必须保持一致。

参考文献：

[1]http://jphome.github.io/blog/2014/08/03/luci\_add\_page.html

[2]http://www.cnblogs.com/chengyi818/p/5094443.html

[3]http://www.leiphone.com/news/201406/diy-a-smart-router-topic-system-configuration.html

[4]https://htmlpreview.github.io/?https://github.com/openwrt/luci/blob/master/documentation/api/index.html