利用nginx执行shell脚本

1、安装openresty


```
wget https://openresty.org/download/openresty-1.15.8.2.tar.gz
tar -zxvf openresty-1.15.8.2.tar.gz
cd openresty
./configure --prefix=/Data/apps/openresty
make
make install
```

2、安装sockproc

> sockproc 是一个服务器程序, 侦测unix socket 或者 tcp socket , 并把收到的命令,传递给子进程执行,并且侦测/tmp/shell.sock 的套接口，子执行完毕后,把结果返回给客户端。

```
git clone https://github.com/juce/sockproc.git
cd sockproc
make
# 最好不要选用/tmp目录
./sockproc /tmp/sockproc.sock
chmod 0666 /tmp/sockproc.sock
```

3、安装lua-resty-shell

```
git clone https://github.com/juce/lua-resty-shell.git
cd lua-resty-shell
# 原目录下有shell.lua文件，备份，覆盖
cp ./lib/resty/shell.lua /Data/apps/openresty/lualib/resty/
```

4、nginx配置文件配置

```
    server {
        listen       80;
        server_name  localhost;

        location / {
            root   html;
            index  index.html index.htm;
        }

        location = /video.php {
            content_by_lua_file /Data/apps/test.lua;
        }

        error_page   500 502 503 504  /50x.html;
        location = /50x.html {
            root   html;
        }
    }
```

5、lua脚本

```
local shell = require "resty.shell"
local cjson = require("cjson")

ngx.req.read_body()
-- 获取json
local args = ngx.req.get_body_data()

if args==nil then
    ngx.exit(ngx.HTTP_BAD_REQUEST)
end
-- 打印下请求方式
-- ngx.say(ngx.var.request_method)
-- 打印postman发送过来的json全貌
-- ngx.say(ngx.req.get_body_data())

-- 用cjson的decode方法处理json，使我们可以获取到具体的json中包含的值
local res_tab = cjson.decode(args)

Rcommand = "/Data/apps/nginx/sbin/"

-- ngx.say(path .. res_tab["id"])

function string.split(input, delimiter)
    input = tostring(input)
    delimiter = tostring(delimiter)
    if (delimiter=='') then return false end
```

```
    local pos,arr = 0, {}
    for st,sp in function() return string.find(input, delimiter, pos, true) end
do
        table.insert(arr, string.sub(input, pos, st - 1))
        pos = sp + 1
    end
    table.insert(arr, string.sub(input, pos))
    return arr
end

local dest_port = string.split(res_tab["id"],":")[2];

local camera = {}
camera['554'] = " 10.44.35.211 ";
camera['555'] = " 10.44.35.214 ";
camera['556'] = " 10.44.35.215 ";
camera['557'] = " 10.44.35.216 ";

-- ngx.say(camera[dest_port])

if camera[dest_port]==nil then
    ngx.exit(ngx.HTTP_BAD_REQUEST)
end

Rcommand = Rcommand .. " 80 " .. camera[dest_port] .. " " .. res_tab["tilt"]

ngx.say(Rcommand)
-- local args = ngx.req.get_post_args()
-- if not args or not args.info then
--    ngx.exit(ngx.HTTP_BAD_REQUEST)
-- end

-- local args = {
--    socket = "unix:/tmp/sockproc.sock",
-- }
-- local status, out, err = shell.execute("ls", args)
-- ngx.header.content_type = "text/plain"
-- ngx.say("Result:\n" .. out)
```

6、postman测试结果