<https://blog.csdn.net/u013377887/article/details/62429457>

# 前言

在linux下，使用socket进行编程，需要到服务器上进行获取数据，服务器使用的php编程，需要使用http的方式进行获取数据。

# 代码

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

#include <errno.h>

#include <sys/types.h>

#include <sys/socket.h>

#include <unistd.h>

#include <netinet/in.h>

#include <arpa/inet.h>

#include <netdb.h>

#define DEST\_PORT 80

#define DEST\_IP\_ADDR "10.0.9.139"

#define DEST\_IP\_BY\_NAME "demo.git.com"

void process\_info(int fd)

{

int send\_num;

char send\_buf [] = "helloworld";

char recv\_buf [4096];

char str1[4096];

while (1)

{

printf("begin send\n");

memset(str1,0,4096);

strcat(str1, "POST http://demo.git.com/sum.php HTTP/1.1\r\n");

strcat(str1,"Host: demo.git.com\r\n");

strcat(str1,"Content-Length: 65\r\n");

strcat(str1,"Content-Type: application/x-www-form-urlencoded\r\n");

strcat(str1,"\r\n");

strcat(str1,"mathod=adb\_signe&token=0E1FEECD0EE54E3B8568A536A7036D78B1AC7EEE");

strcat(str1,"\r\n\r\n");

printf("str1 = %s\n",str1);

send\_num = send(fd, str1,strlen(str1),0);

if (send\_num < 0)

{

perror("send error");

exit(1);

}

else

{

printf("send successful\n");

printf("begin recv:\n");

int recv\_num = recv(fd,recv\_buf,sizeof(recv\_buf),0);

if(recv\_num < 0){

perror("recv");

exit(1);

} else {

printf("recv sucess:%s\n",recv\_buf);

}

}

break;

sleep(5);

}

}

int main()

{

int sock\_fd;

struct sockaddr\_in addr\_serv;

sock\_fd=socket(AF\_INET, SOCK\_STREAM, 0);

if (sock\_fd < 0)

{

perror("sock error");

exit(1);

}

else

{

printf("sock successful");

}

struct hostent\* hostInfo = gethostbyname(DEST\_IP\_BY\_NAME);

if(NULL == hostInfo){

printf("hostInfo is null\n");

return -6;

}

memset(&addr\_serv, 0, sizeof(addr\_serv));

addr\_serv.sin\_family = AF\_INET;

addr\_serv.sin\_port = htons(DEST\_PORT);

//addr\_serv.sin\_addr.s\_addr = inet\_addr(DEST\_IP\_ADDR);

printf("Ip address = %s \n",inet\_ntoa(\*((struct in\_addr\*)hostInfo->h\_addr)));

memcpy(&addr\_serv.sin\_addr, &(\*hostInfo->h\_addr\_list[0]), hostInfo->h\_length);

if (connect(sock\_fd, (struct sockaddr\*)(&addr\_serv), sizeof(addr\_serv)) < 0)

{

perror("connect error\n");

exit(1);

}

else

{

printf("connect successful\n");

}

process\_info(sock\_fd);

}

# 总结

* 需要使用c语言发送http的请求，关键点在构建http的格式，在这个过程中，通过尝试发现在strcat后，每次都需要使用 \r\n 进行分割。没有进行分割的时候，返回了400错误。 这个 \r\n 具体该写不该写，需要具体问题具体分析，多多尝试。
* 在linux c 的编程中。通常使用的是ip地址，例如 10.0.9.139 这样的地址。如果服务提供的是域名，这是就需要将域名利用gethostbyname进行转换。

# 参考文献

[Linux下用c语言实现发送http请求 方式可以Get或者Post](https://www.oschina.net/code/snippet_176076_5908)  
[C socket 发送HTTP请求](http://blog.csdn.net/u011680118/article/details/49287031)

版权声明：本文为博主原创文章，未经博主允许不得转载。 https://blog.csdn.net/u013377887/article/details/62429457