

ネットワークプログラミング

第2回

ソケットプログラミング(2)

【備忘録】ファイル共有の利用

- スタート→ファイル名を指定して実行
¥¥yahiko¥linux
で、yahikoのホームディレクトリが開く。
 - ここでファイルを作成すれば、
FTPなどで転送する必要がない。
- ちなみに、
¥¥yahiko¥ユーザ名
は、「マイ ドキュメント」と同じ場所

クライアント、サーバ プログラム例2

クライアントプログラム例 2

- client.c
 - 指定したIPアドレスの80番ポートにアクセスし、`/index.html` を取得し、表示する。
 - コンパイル: `gcc client.c -o client`
 - 実行: `./client 127.0.0.1`
`./client 202.209.0.1`

【メモ】

- バイトオーダー変換関数

htons ホストからネットワークへ short型を変換

htonl ホストからネットワークへ long型を変換

ntohs ネットワークからホストへ short型を変換

ntohl ネットワークからホストへ long型を変換

サーバプログラム例2

- server.c
 - 80番ポートで接続を待ち、アクセスしてきたクライアントに、特定のHTMLデータを送信する。
 - コンパイル: `gcc server.c -o server`
 - 実行: `./server`
 - 動作を確認する場合は、80番ポートではなく、20000+名簿番号のポート番号を使うように変更すること。

おまけ

- このサーバのプログラムは、IEなどのブラウザからもアクセスできる。
`http://202.209.2.213:20000/`
 - コロン(:)のあとにポート番号を書くと、アクセスするポートを指定できる。省略時は80になっている。

【メモ】

- `unsigned long inet_addr(const char *cp)`
 - `cp`で与えられたIPアドレス形式の文字列を
`long`型の数値に変換
- `char *inet_ntoa(struct in_addr in)`
 - `in`で与えられたIPアドレスを文字列に変換
- `int inet_aton(const char *cp,
 struct in_addr *inp)`
 - `cp`で与えられたIPアドレスの文字列を引数`inp`
に変換して出力

IPアドレスではなく名前を使うには `gethostbyname`

- サーバの指定には、IPアドレスではなく、名前が使えたほうが便利。

```
struct hostent *gethostbyname(const char *hostname)
```

- hostname には、FQDNの文字列を渡す
例えば、“www.nagaoka-ct.ac.jp”
- 戻り値のhostent構造体に、そのホスト名に対応するコンピュータのIPアドレスが格納される。
 - netdb.h が必要。

hostent構造体

- struct hostent {
 char *h_name; /* 正式名称 */
 char **h_aliases; /* エイリアス(別名)テーブル */
 int h_addrtype; /* アドレスタイプ */
 int h_length; /* アドレスの大きさ */
 char **h_addr_list; /* アドレスリスト */
}

h_length --- IPv4 の場合は4

IPアドレスは、h_addr_list に入っている。
ひとつのアドレスに複数のIPアドレスがある場合があるため、
ポインタの配列という形式になっていて、取り出すのが少し複雑。

ホスト名からIPアドレスを得る例

```
/* getip.c */
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <arpa/inet.h>
#include <netdb.h>
int main(int argc, char *argv[])
{
    struct hostent *pe;
    struct in_addr addr;
    int i;
    char *p;
    if (argc < 2) {
        printf("(usage) %s hostname¥n",
            argv[0]);
        exit(EXIT_FAILURE);
    }
```

```
    pe = gethostbyname(argv[1]);
    if (pe == NULL) {
        printf("host not found¥n");
        exit(EXIT_SUCCESS);
    }
    printf("name:%s¥n", pe->h_name);
    for (i=0;
        pe->h_addr_list[i] != NULL;
        i++) {
        p = pe->h_addr_list[i];
        memcpy(&addr.s_addr, p,
            pe->h_length);
        printf("ip(%d): %s¥n",
            i, inet_ntoa(addr));
    }
    return 0;
}
```

getip を使ってみる

- コンパイル `gcc getip.c -o getip`
- 実行 `./getip ホスト名`
実行例) `./getip www.nagaoka-ct.ac.jp`
`./getip www.yahoo.co.jp`

課題

1. client.c に手を加え、
 - 引数に、ホスト名を指定できるようにする
 - 二つ目の引数として、アクセスするポート番号を指定できるようにする(省略時は80番にする)を実現せよ。
2. server.c に手を加え、
 - アクセスしてきたクライアントのIPアドレスを画面に表示するようにしてみよ。
 - 引数に、待ち受けポート番号を指定できるようにしてみよ(省略時は20000 + 名簿番号とする)。

Teamsで課題を割り当てるので、そちらに
上記 2つのプログラムを提出してください。

課題の補足事項

- サーバもクライアントもyahikoで動作させているため、クライアントのIPアドレスを表示した場合、yahikoのIPアドレスが出る。
- IEなどのブラウザからサーバプログラムにアクセスした場合もプロキシサーバ(代理サーバ)を経由した場合には、プロキシサーバのIPアドレスが出てきます。

補足続き

- `gethostbyname()` の結果が、IPアドレス1つだけでない場合
 - `connect()`に成功するまで、すべてのIPアドレスを順に試すのが通常。
 - 課題では、アドレステーブルの最初のものだけを使うことにしてもよい。
- `inet_ntoa()` で得られる文字列へのポインタは、予め用意されているメモリのアドレス。
 - 結果を保存しておきたいならば、`strncpy()`などで文字列を別の配列にコピーする必要がある。