

Proiect Retele MySSH(A)

Aioane Dragos Costin

16.01.2019

1 Introduction

1.1 Enuntul problemei

In acest proiect unul sau mai multi clienti se conecteaza la server si executa comenzi bash remote.

2 Tehnologii utilizate

TCP este un protocol prin care se trimite informatii sub forma de byets. In alte cuvinte, de fiecare data cand se trimite un packet, acesta este trimis pe o adresa IP. Concurenta este asigurata de functia **poll** asemanatoare functiei **select**.

3 Arhitectura aplicatiei

3.1 Conectarea

Am hotarat sa fac conectarea clientilor la server cu ajutorul unui socket. Acesta trimite informatiile clientului pe care in viitor serverul trebuie sa le foloseasca. Urmatoarea structura arata ce informatii are nevoie clientul pentru a se putea conecta la server: „server.sin_family = AF_INET;” prin care primeste familia de adrese, „server.sin_addr.s_addr = inet_addr(argv[1]);” prin care acesta primeste IP-ul si „server.sin_port = htons(atoi (argv[2]));” cu ajutorul caruia clientul stie la ce port trebuie sa se conecteze.

3.2 Trimiterea informatiilor

Clientul citeste de la tastatura un nume si o parola cu ajutorul carora se conecteaza la server. Daca acestea nu se afla in fisier atunci clientul nu se poate conecta. Acestea sunt citite inainte de conectare pentru a ne asigura ca serverul nu este blocat de primitiva „recv”. Daca conectarea a reusit, clientului i se cere o comanda pe care serverul sa o execute, dupa care sa trimita raspunsul inapoi la client.

3.3 Funcționalitatea serverului

Serverul după ce a primit un nume și o parolă de la client începe să facă validarea acestuia. Aceasta validare se face în cadrul unui fișier de tip „.xml”. După ce primește comanda criptată de la client, acesta o decriptează și începe să caute operatorii speciali pentru a realiza o parsare a comenzii. Aceasta ajută Serverul să găsească mai ușor instrucțiunile pe care le are de executat și să aplice funcționalitatea operatorului găsit în comandă.

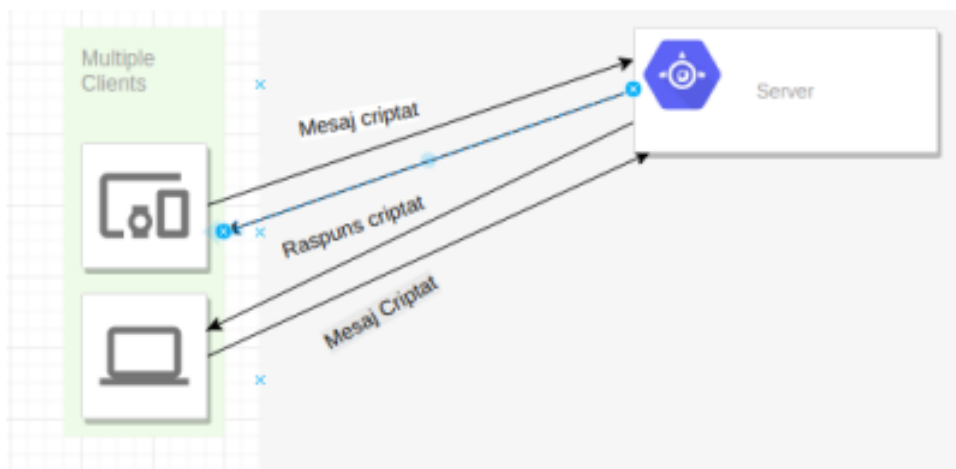
3.4 Detalii input

Un input valid pentru ca programul în cauză arată în felul următor. `grep socket < cliTcp.cpp`
`< cliTip.cpp ls & pwd ls udhdj || pwd`

- `grep socket < cliTcp.cpp`
- `ls`
- `ls & pwd`
- `ls udhdj || pwd`

Între fiecare comandă și operatorii speciali poate exista un singur spațiu.

3.5



4 Detalii de Implementare

Funcția main folosită de clienți.

```

struct sockaddr_in server; // structura folosita pentru conectare
bzero(&server, sizeof(server));
server.sin_family = AF_INET;
server.sin_addr.s_addr = inet_addr(argv[1]);
server.sin_port = htons(atoi(argv[2]));

int sd; // descriptorul de socket
while(1) {
    char nume[100]; // citesc numele
    bzero(nume, 100);
    printf("[client]Introduceti un nume: ");
    fflush(stdout);
    int readBytes = read(0, nume, 100);
    nume[readBytes-1] = 0;

    char parola[100]; // citesc parola
    bzero(parola, 100);
    printf("[client]Introduceti un parola: ");
    fflush(stdout);
    readBytes = read(0, parola, 100);
    parola[readBytes-1] = 0;
    if ((sd = socket(AF_INET, SOCK_STREAM, 0)) == -1) {
        perror("[client] Eroare la socket().\n");
        return errno;
    }

    /* ne conectam la server */
    if (connect(sd, (struct sockaddr *) &server, sizeof(struct sockaddr)) == -1) {
        perror("[client]Eroare la connect().\n");
        return errno;
    }

    trimiteSir(sd, nume);
    trimiteSir(sd, parola);
}

```

Functia folosita de server la validare de useri

```

tinyxml2::XMLDocument doc;
doc.LoadFile("SavedData.xml");
int valid = 0;
for( const tinyxml2::XMLElement* child = doc.FirstChildElement("Utilizator")->FirstChildElement("data");
    child;
    child = child->NextSiblingElement())
{
    if(child->Attribute("Username") && child->Attribute("Parola")){
        const char* nume_xml;
        nume_xml = child->Attribute("Username");
        const char* parola_xml;
        parola_xml = child->Attribute("Parola");
        std::cout << "Username = " << nume_xml << ' ' << strlen(nume_xml) << "Nume: " << nume << std::endl;
        if(strcmp(nume_xml, nume) == 0 && strcmp(parola_xml, parola) == 0)
            valid = 1;
    }
}

```

5 Îmbunătățiri care se pot aduce

- Îmbunătățirea funcției de criptare

- O conexiune mai rapida
- Transmiterea datelor într-un mod mai eficient