МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ

ВЫСШЕГО ОБРАЗОВАНИЯ

«ИЖЕВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. М.Т. Калашникова»

Кафедра «Защита информации в компьютеризированных системах»

Отчёт по лабораторной работе №6

по дисциплине «Программно-аппаратные средства обеспечения информационной безопасности»

на тему «Создание клиент-серверной программы-шутки»

Выполнил:

студент группы С08-361-1

Максимова А.В.

Проверил:

к.т.н, доцент

Вдовин А. Ю.

Ижевск 2017

Цель работы:

Изучение основных принципов создания клиент-серверной программы-шутки.

Задание:

Создание клиент-серверной программы-шутки. После запуска серверной части программы на компьютере-жертве она работает в фоновом режиме (окно программы отсутствует либо невидимо), ожидая сигналов от клиентской части. Клиентская часть позволяет осуществлять нежелательные для владельца компьютера-жертвы действия (блокировать нажатия некоторых клавиш, изменять поведение курсора мыши, модифицировать содержимое рабочего стола, демонстрировать видеоролики с невозможностью их завершения и пр.).

Листинг программы:

Клиент:

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Net;

using System.Net.Sockets;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace SystemServer

{

public partial class FormClient : Form

{

const int MAX\_CONNECTIONS = 10;

const int ACTIVE\_PORT = 53379;

public FormClient()

{

InitializeComponent();

}

private void SendJoke\_Click(object sender, EventArgs e)

{

try

{

SendMessageFromSocket(ACTIVE\_PORT);

}

catch (Exception ex)

{

}

finally

{

}

}

static void SendMessageFromSocket(int port)

{

// Буфер для входящих данных

byte[] bytes = new byte[1024];

// Соединяемся с удаленным устройством

// Устанавливаем удаленную точку для сокета

IPHostEntry ipHost = Dns.GetHostEntry("localhost");

//IPAddress ipAddr = ipHost.AddressList[0];

IPAddress ipAddr = IPAddress.Parse("127.0.0.1");

IPEndPoint ipEndPoint = new IPEndPoint(ipAddr, port);

Socket sender = new Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp);

// Соединяем сокет с удаленной точкой

sender.Connect(ipEndPoint);

string message = "<ShowMustGoOn>";

byte[] msg = Encoding.UTF8.GetBytes(message);

// Отправляем данные через сокет

int bytesSent = sender.Send(msg);

// Получаем ответ от сервера

int bytesRec = sender.Receive(bytes);

// Освобождаем сокет

sender.Shutdown(SocketShutdown.Both);

sender.Close();

}

}

}

Сервер:

using System;

using System.Collections.Generic;

using System.Diagnostics;

using System.Linq;

using System.Net;

using System.Net.Sockets;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace SystemClient

{

static class Program

{

/// <summary>

/// Главная точка входа для приложения.

/// </summary>

[STAThread]

static void Main()

{

const int MAX\_COUNT = 10;

var JokeArr = new string[MAX\_COUNT, 2];

// Заполним массив

JokeArr[0, 0] = "Купи слона!";

JokeArr[0, 1] = "https://otvet.mail.ru/question/92324700";

JokeArr[1, 0] = "Все нажимают эту кнопку, а ты купи слона!";

JokeArr[1, 1] = "https://otvet.imgsmail.ru/download/695303f47291f3111acc27eeba264a32\_i-437.gif";

JokeArr[2, 0] = "Ну купи, купи уже слона!! Балдахин не должен оставаться один";

JokeArr[2, 1] = "http://btalk.org/class/oc-content/uploads/0/3.jpg";

JokeArr[3, 0] = "Не хочешь слона, возьми тогда вьетнамскую вислобрюхую свинью";

JokeArr[3, 1] = "https://www.avito.ru/izhevsk/drugie\_zhivotnye/vetnamskie\_vislobryuhie\_porosyata\_917017334";

JokeArr[4, 0] = "А ведь слона можно дрессировать!! Вот просыпаешься ты утром, а он тебе уже кофе сварил, тапочки принес и лежит рядышком.. Хвостиком виляет ^\_^";

JokeArr[4, 1] = "http://zooclub.ru/attach/21000/21107.jpg";

JokeArr[5, 0] = "КУПИ Я СКАЗАЛА!!!!!!!!!";

JokeArr[5, 1] = "https://otvet.mail.ru/question/92324700";

JokeArr[6, 0] = "Вот поиграй пока, и подумай над покупкой слона";

JokeArr[6, 1] = "https://www.google.ru/search?q=do+a+tilt&newwindow=1&source=lnms&tbm=isch&sa=X&ved=0ahUKEwiuzYHYzNDSAhVBXSwKHWERC3kQ\_AUIBigB&biw=1366&bih=662#newwindow=1&tbm=isch&q=atari+breakout&\*";

JokeArr[7, 0] = "А в Объединенных Арабских Эммиратах уже просекли фишку и скупают их пачками!!! Успей и ты ;)";

JokeArr[7, 1] = "http://zooclub.ru/attach/21000/21167.jpg";

JokeArr[8, 0] = "Когда кто-то не покупает слона, в мире грустит один слоненок :( Купи слона!!";

JokeArr[8, 1] = "http://ic.pics.livejournal.com/krapan\_5/15106335/423156/423156\_original.jpg";

JokeArr[9, 0] = "Ну и фиг с тобой.. Но знай, что из-за таких как ты, вот этот слоненок потерялся.. Нужно вовремя покупать слонов!!";

JokeArr[9, 1] = "https://www.youtube.com/watch?v=DsHVwwkQPr0";

// Устанавливаем для сокета локальную конечную точку

IPHostEntry ipHost = Dns.GetHostEntry("localhost");

//IPAddress ipAddr = ipHost.AddressList[0];

IPAddress ipAddr = IPAddress.Parse("127.0.0.1");

IPEndPoint ipEndPoint = new IPEndPoint(ipAddr, 53379);

// Создаем сокет Tcp/Ip

Socket sListener = new Socket(AddressFamily.InterNetwork, SocketType.Stream, ProtocolType.Tcp);

// Назначаем сокет локальной конечной точке и слушаем входящие сокеты

try

{

sListener.Bind(ipEndPoint);

sListener.Listen(10);

// Начинаем слушать соединения

while (true)

{

// Программа приостанавливается, ожидая входящее соединение

Socket handler = sListener.Accept();

string data = null;

// Мы дождались клиента, пытающегося с нами соединиться

byte[] bytes = new byte[1024];

int bytesRec = handler.Receive(bytes);

data += Encoding.UTF8.GetString(bytes, 0, bytesRec);

// Отправляем ответ клиенту\

string reply = "Спасибо за запрос в " + data.Length.ToString()

+ " символов";

byte[] msg = Encoding.UTF8.GetBytes(reply);

handler.Send(msg);

if (data.IndexOf("<TheEnd>") > -1)

{

break;

}

if (data.IndexOf("<ShowMustGoOn>") > -1)

{

for (int i = 0; i < MAX\_COUNT; ++i)

{

var text = JokeArr[i, 0];

var site = JokeArr[i, 1];

Process.Start(site);

MessageBox.Show(text, "Купи слона", MessageBoxButtons.OK, MessageBoxIcon.Error, MessageBoxDefaultButton.Button1, MessageBoxOptions.ServiceNotification);

}

}

handler.Shutdown(SocketShutdown.Both);

handler.Close();

}

}

catch (Exception ex)

{

}

finally

{

}

}

}

}

Скриншоты работы программы:

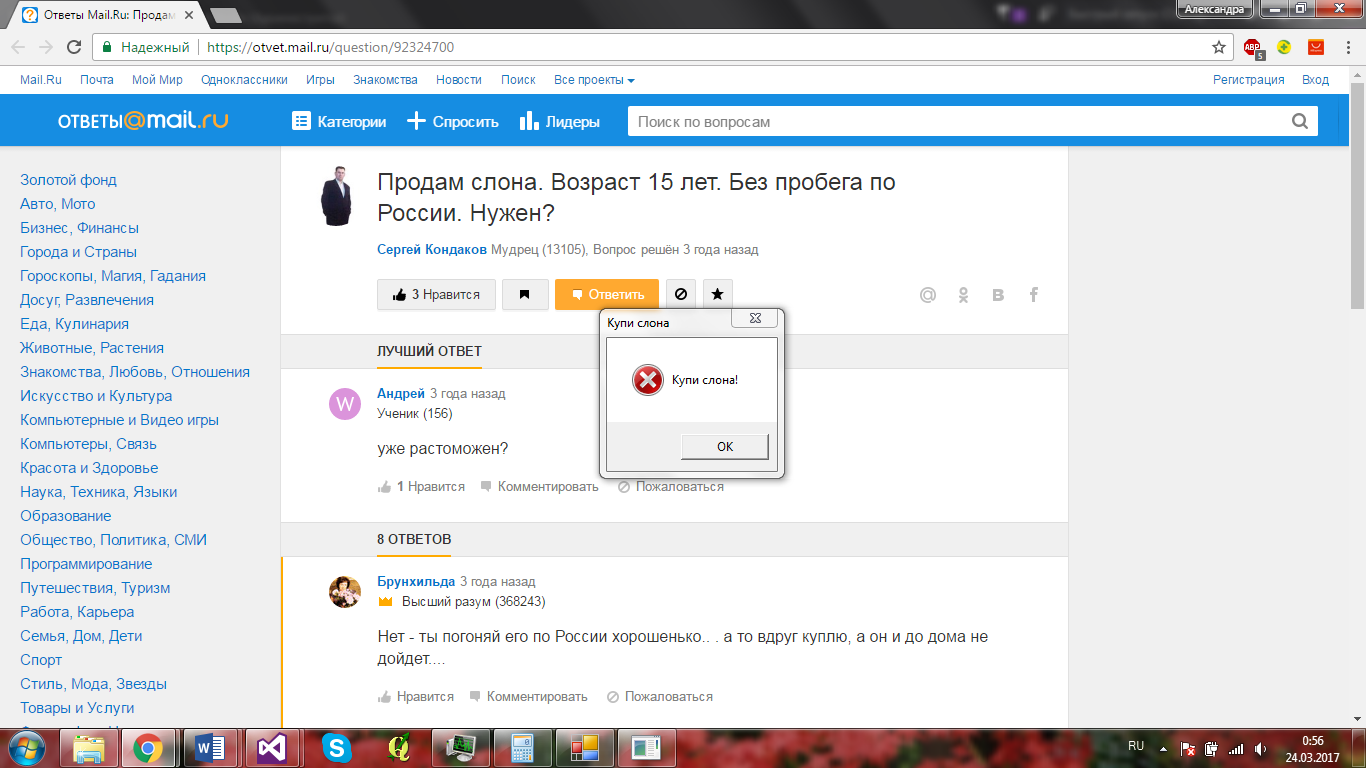


Рис. 1 – Первое окно программы – шутки

Вывод:

В данной лабораторной работе были изучены основные принципы создания клиент – серверного приложения, серверная часть которого работает в фоновом режиме.