Министерство образования и науки Российской Федерации

Санкт-Петербургский Политехнический Университет Петра Великого

—

Институт компьютерных наук и кибербезопасности

ЛАБОРАТОРНАЯ РАБОТА № 3

«РАСПРЕДЕЛЕННАЯ СИСТЕМА СБОРА ИНФОРМАЦИИ»

1. по дисциплине «Безопасность современных информационных технологий»
2. Выполнил
3. студент гр. 5151004/10101 Веремейчук Я.Ю.

<подпись>

1. Преподаватель
2. асс. преподавателя Соловей Р.С.

<подпись>

1. Санкт-Петербург
2. 2023
3. Цель работы

Получить навыки организации параллельной обработки большого количества клиентских соединений при помощи механизма портов завершения Win32. Получить навыки реализации шифрования передаваемых данных при помощи средств CryptoAPI. Получить навыки извлечения информации о системе.

1. Задача

Разработать распределенную систему сбора информации о компьютере, состоящую из сервера и клиента, взаимодействующих через сокеты.

1. Ход работы

Для начала был изучен методический материал.

Код сервера из методического материала был протестирован на хосте.

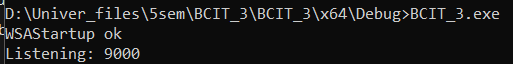


Рисунок 1 – Запуск сервера

Чтобы протестировать взаимодействие с клиентом на хосте, необходимо воспользоваться командой telnet 127.0.0.1 9000. Была получена ошибка: "telnet" не является внутренней или внешней командой, исполняемой программой или пакетным файлом. Чтобы исправить это, необходимо убедиться, что Telnet-клиент установлен. В стандартных настройках Windows 10 он может быть отключен. Чтобы включить, необходимо открыть «Панель управления» -> «Программы и компоненты» -> «Включение или отключение компонентов Windows» -> поставить галочку на «Клиент Telnet».

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рисунок 2 – Включение Telnet-клиента

После поиска необходимых компонентов настройки были применены.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 3 – Применение изменений

После удачного запуска клиента (на хосте) можно увидеть соответствующее окно. После ввода некоторых символов можно получить сообщение о длине строки.

Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Автоматически созданное описание

Рисунок 4 – Запуск клиента

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 5 – Работа клиента

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

Рисунок 6 – Работа сервера

Далее был написан код для клиента и сервера в соответствии с заданием.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 7 – Вывод возможных команд

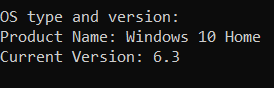


Рисунок 8 – Вывод информации о типе и версии ОС

Изображение выглядит как текст, Шрифт, снимок экрана, Графика

Автоматически созданное описание

Рисунок 9 – Текущее время

Изображение выглядит как текст, Шрифт, снимок экрана, типография

Автоматически созданное описание

Рисунок 10 – Время с момента запуска ОС

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 11 – Информация об используемой памяти

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

Рисунок 12 – Типы подключенных дисков

Также была протестирована ситуация, когда используется флеш-накопитель. Он также отображается в списке дисков.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Автоматически созданное описание

Рисунок 13 – Типы подключенных дисков с флешкой

Изображение выглядит как текст, Шрифт, снимок экрана, Графика

Автоматически созданное описание

Рисунок 14 – Свободное место на локальных дисках

Изображение выглядит как снимок экрана, текст

Автоматически созданное описание

Рисунок 15 – Права доступа

Изображение выглядит как текст, Шрифт, Графика, типография

Автоматически созданное описание

Рисунок 16 – Владелец

Было протестировано взаимодействие клиента на хостовой системе и сервера на виртуальной машине. Сначала это не получалось сделать. На обеих системах был отключен брандмауэр.

Изображение выглядит как текст, программное обеспечение, Шрифт, веб-страница

Автоматически созданное описание

Рисунок 17 – Отключение брандмауэра на хосте

Изображение выглядит как текст, снимок экрана, программное обеспечение, веб-страница

Автоматически созданное описание

Рисунок 18 – Отключение брандмауэра на виртуальной машине

Но это не помогало. Тогда была попытка спинговать две системы. Но пакеты не передавались.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 19 – Попытка спинговаться с виртуальной машиной

Тогда на виртуальной машине были изменены настройки сети.

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 20 – Настройка сети

Настройки сети на виртуальной машине изменились.

Изображение выглядит как текст, снимок экрана, Веб-сайт, веб-страница

Автоматически созданное описание

Рисунок 21 – Настройки сети виртуальной машины

Теперь системы пингуются. Можно запускать написанные программы.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 22 – Успешный пинг

Далее была произведена проверка установки сеансового ключа. Была установлена утилита Wireshark и проанализирован трафик.

Изображение выглядит как снимок экрана, текст, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание

Рисунок 23 – Установка сеансового ключа

1. Клиент отправляет пакет с флагом SYN, указывая начало установки соединения.
2. Сервер получает пакет от клиента и отвечает пакетом с флагами SYN и ACK. Этот пакет подтверждает получение пакета от клиента и начинает процесс установки соединения.
3. Клиент подтверждает получение ответа сервера путем отправки пакета с флагом ACK.
4. Клиент отправляет данные (флаг PSH) вместе с ACK. Длина данных составляет 149 байт.
5. Сервер также отправляет данные (флаг PSH) вместе с ACK. Длина данных составляет 141 байт.
6. Клиент подтверждает получение данных от сервера, отправляя пакет с флагом ACK.
7. Контрольные вопросы
8. Какова структура списков контроля доступа в ОС Windows?

В операционной системе Windows списки контроля доступа (ACL) содержат записи о пользователях или группах пользователей, а также о том, какие разрешения предоставлены или запрещены для определенных объектов (например, файлов или папок). Структура ACL включает в себя следующие элементы:

* ACE (Access Control Entry): Запись, определяющая разрешения или запреты для конкретного пользователя или группы. Каждая запись содержит идентификатор безопасности (SID) пользователя или группы, тип доступа (разрешение или запрет) и соответствующие атрибуты.
* SID (Security Identifier): Уникальный идентификатор безопасности, присвоенный каждому объекту (пользователю, группе, процессу и т. д.) в системе. Он уникален в пределах домена или локальной системы.
* Разрешения (Permissions): Определяют, какие действия разрешены или запрещены для пользователя или группы. Примеры разрешений включают чтение, запись, выполнение и т. д.
* ACL (Access Control List): Список, содержащий все ACE для конкретного объекта.

1. Что такое наследование прав доступа?

Наследование прав доступа в ОС Windows позволяет объектам наследовать разрешения от их родительских объектов (например, папок). Это упрощает управление правами доступа, поскольку разрешения, установленные на уровне родительского объекта, автоматически применяются к его дочерним объектам. Если у объекта нет явно установленных разрешений, он наследует их от своего родительского объекта.

1. Для чего используются well-known SID?

Well-known SID (идентификатор безопасности) – это уникальный идентификатор, который ассоциирован с предопределенной группой или пользователем в операционной системе Windows. Они используются для облегчения работы с системными объектами и обеспечивают стандартные уровни доступа. Например, well-known SID может представлять группу "Администраторы" или "Пользователи". Использование well-known SID обеспечивает унификацию идентификации для таких системных сущностей.

1. Как выглядит схема шифрования с использованием сеансового ключа?

В схеме шифрования с использованием сеансового ключа данные шифруются с использованием временного ключа, который генерируется специально для данной сессии. Этот ключ обычно не сохраняется и используется только в течение текущей сессии обмена информацией. Процесс может выглядеть следующим образом:

* Стороны обмениваются информацией для установки безопасного соединения.
* Генерируется сеансовый ключ, который используется для шифрования данных в рамках текущей сессии.
* Сеансовый ключ может быть обновлен при необходимости.

1. В чем преимущества использования сеансового ключа?

* Ограниченный срок действия: сеансовые ключи обычно имеют ограниченный срок действия, что означает, что даже если они были скомпрометированы, у злоумышленника будет ограниченное время для их использования.
* Уменьшенный риск повторного использования: поскольку сеансовый ключ используется только в течение одной сессии, риск повторного использования ключа снижается.
* Легкость обновления: сеансовые ключи могут быть обновлены в процессе обмена данными, что усиливает безопасность сеанса.
* Большая сложность для атаки: если ключи изменяются для каждой сессии, это делает задачу злоумышленника более сложной, так как даже при компрометации одного ключа, остальные ключи остаются неповрежденными.

1. Выводы

В ходе работы были получены навыки организации параллельной обработки большого количества клиентских соединений при помощи механизма портов завершения Win32, получены навыки реализации шифрования передаваемых данных при помощи средств CryptoAPI. Получить навыки извлечения информации о системе.