Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и кибербезопасности

Высшая школа компьютерных технологий и информационных систем

**Отчёт по лабораторным работам**

Дисциплина: Технологии компьютерных сетей.

Выполнил студент гр. 5130901/10101 \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Д.Л. Симоновский (подпись)

Руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Н.В. Богач (подпись)

“10” сентября 2024 г.

Санкт-Петербург

2024

Оглавление

[1. Лабораторная работа 1. Wireshark: Введение. 2](#_Toc176889014)

[1.1. Цель работы: 2](#_Toc176889015)

[1.2. Ход работы: 2](#_Toc176889016)

[1.3. Вывод: 5](#_Toc176889017)

[2. Приложение: 6](#_Toc176889018)

# Лабораторная работа 1. Wireshark: Введение.

## Цель работы:

В этой лабораторной работе мы познакомимся с программой Wireshark, которая используется для анализа сетевого трафика путем перехвата пакетов данных и изучения их структуры. Wireshark является мощным инструментом для мониторинга сетевой активности, позволяющим наблюдать за обменом сообщениями между протоколами, такими как HTTP, FTP, TCP, UDP, DNS, и IP, на различных уровнях сетевой архитектуры.

Цель данной работы — углубить понимание работы сетевых протоколов, увидеть их в действии, анализируя последовательности пакетов, передаваемых между устройствами в сети. Мы будем наблюдать, как протоколы, используемые нашим компьютером, обмениваются данными с удаленными узлами сети Интернет, что позволит лучше понять работу сетевых приложений.

В ходе лабораторной работы мы научимся использовать Wireshark для захвата сетевых пакетов, интерпретации их структуры и анализа различных полей протокольных сообщений. Особое внимание будет уделено тому, как сообщения верхних уровней, такие как HTTP-запросы, инкапсулируются в кадры канального уровня и передаются по сети. Это позволит понять, как данные путешествуют через сетевые уровни от приложения до физического канала связи.

Таким образом, мы увидим на практике, как работают сетевые протоколы, и получим ценные навыки работы с инструментами анализа сетевого трафика, что является важной частью изучения современных компьютерных сетей.

## Ход работы:

Перейдем непосредственно в Wireshark и посмотрим на главный экран:

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

Автоматически созданное описание

Рис. 1.1. Главный экран Wireshark.

Здесь мы видим все подключенные интернет адаптеры к компьютеру, в том числе различные VPN подключения. Ethernet 2 является основным адаптером, поэтому выберем именно его. Тогда окно изменится следующим образом:

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

Автоматически созданное описание

Рис. 1.2. Wireshark. Захват из Ethernet 2.

На рис. 1.2. мы видим следующие поля в интерфейсе:

* Командные меню — меню для сохранения, открытия, захвата данных и др.
* Поле фильтра — фильтрация пакетов по протоколам и критериям.
* Окно списка пакетов — список перехваченных пакетов с фильтрацией.
* Окно деталей заголовка — детальная информация по выбранному пакету.
* Окно содержимого пакета — данные пакета в шестнадцатеричном и ASCII формате.

Как мы видим по рисунку 1.2. наш компьютер непрерывно обменивается множеством различных пакетов с ресурсами, которые мы даже не запускали. Попробуем отследить конкретно какой-то пакет. Например, по протоколу http, для этого в поле фильтров напишем http. Теперь нам будут отображаться только http пакеты. Попробуем посмотреть на них, для этого перейдем в браузер и зайдем на сайт <https://gaia.cs.umass.edu/wireshark-labs/INTRO-wireshark-file1.html>. В браузере отобразится следующее окно:

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рис. 1.3. Результат перехода на сайт.

Однако для нас представляет интерес, что мы увидим в wireshark:

Изображение выглядит как текст, программное обеспечение, Мультимедийное программное обеспечение, Значок на компьютере

Автоматически созданное описание

Рис. 1.4. Окно Wireshark после открытия страницы.

Можно увидеть 2 GET запроса и 2 ответа. Первый – запрос непосредственно веб-страницы, по протоколу HTTP, а второй запрос какого-то файла иконки, который не был успешно получен. Рассмотрим первый запрос подробнее:

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

Автоматически созданное описание

Рис. 1.5. Пакет запроса в Wireshark.

Как мы видим, здесь есть вся информация о нашем запросе, в том числе метод (GET) и другая служебная информация, такая как User-Agent и др.

Далее рассмотрим ответ, который, как видно по рисунку, идет следующим:

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

Автоматически созданное описание

Рис. 1.6. Пакет ответа в Wireshark.

Как видим, тут есть как тело ответа с веб-страницей, так и описание самого ответа, в том числе его код и другая служебная информация.

Также используя Wireshark можно обнаружить ip адрес своего компьютера, он находится в столбце Source (Отправитель) у отправляемого пакета и наоборот в столбце назначение (Destination) у принимаемого, в моем случае это 192.168.1.120, это ip адрес внутри моей локальной подсети. Также можно посмотреть на ip адрес сервера, куда отправляется запрос, в столбце назначение (Destination) у отправляемого пакета и наоборот в столбце назначение Source (Отправитель) у принимаемого, в данном случае сервер имеет ip 128.119.245.12.

Еще можно посмотреть время между принятием и отправкой пакета, для этого обратимся к столбцу time, по которому видно, что между запросом и ответом прошло коло 120 ms (0.12 сек), что достаточно быстро т. к. принимаемый пакет достаточно маленький.

Сохраним пакеты, их можно будет найти в репозитории лабораторной или по следующей ссылке: [github.com/DafterT/TKS\_Labs/tree/main/Лабораторная 1](https://github.com/DafterT/TKS_Labs/tree/main/%D0%9B%D0%B0%D0%B1%D0%B0%201)

Wireshark позволяет анализировать не только http трафик, но и множество других, что видно на следующем рисунке:

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

Автоматически созданное описание

Рис. 1.7. Окно Wireshark с различными пакетами.

Здесь можно заметить такие протоколы, как UPD, ARP, TCP и некоторые другие, этот список далеко не полный и Wireshark позволяет работать со множеством других интерфейсов, что будет рассмотрено в последующих лабораторных.

## Вывод:

В ходе лабораторной работы была успешно изучена программа Wireshark, которая используется для анализа сетевого трафика. Мы научились захватывать пакеты данных, анализировать их структуру и извлекать полезную информацию о работе сетевых протоколов. На практическом примере с HTTP-запросом и ответом мы рассмотрели, как сетевые протоколы передают данные между устройствами. Было продемонстрировано, как данные инкапсулируются и передаются через различные уровни сетевой архитектуры, от прикладного до канального.

Мы также научились использовать фильтры Wireshark для отслеживания конкретных протоколов, что упрощает анализ большого объема трафика. Определение IP-адресов отправителей и получателей, а также анализ времени между запросами и ответами помогли глубже понять процессы взаимодействия в сети.

Таким образом, данная работа позволила на практике увидеть, как функционируют сетевые протоколы и как передаются данные в сети. Мы освоили базовые навыки работы с инструментами анализа сетевого трафика, что является важным шагом в изучении принципов работы современных компьютерных сетей. Полученные знания помогут лучше понимать сетевые процессы и взаимодействие между устройствами в сети.

# Приложение:

Ссылка на репозиторий с исходными кодами: <https://github.com/DafterT/TKS_Labs>