**Introduction**

Bluetooth and USB devices have become ubiquitous in our daily lives, used for various purposes such as transferring files, connecting to wireless headphones, and charging our smartphones. To better understand the behaviour of these devices, it is important to investigate the various protocols involved in their communication.

In this project, we will use Wireshark, an open-source software, to read live data from Bluetooth and USB devices and investigate 10 protocols. By reading live data, we can gain insights into how Bluetooth and USB communication works, as well as the various protocols involved in their communication. The project aims to gain a deeper understanding of Bluetooth and USB devices, as well as the protocols involved in their communication.

The report will describe the methodology used to capture and analyze Bluetooth and USB traffic, as well as the protocols investigated using Wireshark.

**Objective**

The objective of this project is to use Wireshark to analyze a computer network at the microscopic level and investigate at least 10 protocols. In addition to analyzing network traffic, the project will also involve reading live data from Bluetooth and USB devices. The project aims to gain insights into how network protocols work, identify potential security risks, and improve the performance of the network. The report will describe the methodology used to capture and analyze network traffic, the tools and equipment used, and the results obtained from the analysis. Additionally, the report will discuss the limitations of the project and future work that could be done to improve or expand upon the results. By gaining a deeper understanding of network protocols and USB/Bluetooth communication, we can better secure and optimize computer networks for a variety of applications.

**Description**

The project aims to gain a better understanding of Bluetooth and USB devices, as well as the various protocols involved in their communication. By using Wireshark, an open-source software, we will read live data from Bluetooth and USB devices and investigate 10 different communication protocols.

The project will start with an overview of Bluetooth and USB devices, their common use cases, and how they work. We will then describe the methodology used to capture and read live data from these devices, including the equipment and tools used.

Using Wireshark, we will investigate 10 different protocols involved in Bluetooth and USB communication, such as Bluetooth Low Energy (BLE), USB 3.0, and Wireless USB.

**Scope**

The scope of this project is to investigate the protocols involved in Bluetooth and USB communication using Wireshark. We will focus on reading live data from Bluetooth and USB devices and analysing 10 different communication protocols.

**System Description**

The system used for this project includes a computer running the Wireshark software and a network interface card (NIC) connected to the network being analysed. Wireshark is an open-source software tool that allows for the capturing and analysis of network traffic. The NIC used for capturing traffic should be capable of promiscuous mode, which allows it to capture all traffic on the network, even if it's not intended for the host computer.

Additionally, for investigating Bluetooth traffic, a Bluetooth adapter is required to capture Bluetooth packets. The adapter should be compatible with the Bluetooth version being used and should support the Bluetooth protocol analyser in Wireshark.

For investigating USB traffic, a USB protocol analyser should be used. The protocol analyser can be a hardware device or a software tool that captures USB packets and sends them to Wireshark for analysis. The USB device being analysed should be connected to the analyser using a USB cable. Thus, the tool that I used is USBPcap which worked as an extension to the Wireshark.

The system requirements for running Wireshark are minimal, with the software being compatible with various operating systems, including Windows, which is the default OS my laptop have.