# Packet Sniffer Project Report

## 1. Introduction

This project is focused on the design and implementation of a packet sniffer. Packet sniffing is a technique that is used to monitor and analyze network traffic. The goal was to develop a Python-based packet sniffer with the use of Scapy to capture and analyze live network packets. This helps in understanding network communications and the identification of potential security risks.

## 2. Technical Implementation

The Packet sniffer was successfully implemented using Python and the Scapy library, with additional analysis performed using Wireshark.

Tools Used:

* Python 3
* Scapy (Packet manipulation library)
* Wireshark (Packet analysis tool)

The sniffer captured network packets, filtered them based on various protocols, and saved the data for further analysis.

## 3. Findings & Analysis

The sniffer successfully captured various types of network packets, including:

* -NetBIOS and SMB traffic (used for Windows file sharing)
* SSDP (UPnP) traffic from IoT and network discovery services
* mDNS packets used for device discovery on local networks
* Unexpected UDP traffic on ports 4001 and 10001
* TCP traffic communicating with an external server (HTTP/HTTP\_ALT)

My analysis of the captured packets revealed a normal network activity, but some unexpected broadcasts suggested potential security risks.

## 4. Challenges & Solutions

During implementation, the following challenges were encountered:

* first time setting up the environment was challenging.
* Permission errors when running the sniffer
* Unexpected network broadcasts on UDP port 4001
* identifying unknown traffic sources

These challenges were resolved by running the script with root privileges, using netstat and Wireshark to investigate unknown traffic, and filtering unnecessary packets.

## 5. Conclusion & Recommendations

This project taught the importance of packet sniffing in network security and traffic analysis. By the monitoring network packets, potential security risks can be identified and mitigated.

Recommendations:

* Regularly monitor network traffic for anomalies
* Disable unnecessary network services (e.g., UPnP) to reduce attack surface
* Implement firewall rules to restrict unauthorized broadcasts