Examining Electromagnetic Emission from Ethernet Cables for Data Exfiltration from the Far-field

W.R.A.S. Kavishka

Index No: 20000962

Supervisor: Dr. Asanka P. Sayakkara

May 2025

Submitted in partial fulfillment of the requirements of the
B.Sc. (Honours) in Computer Science Final Year Project

UCSC

Declaration

I certify that this dissertation does not incorporate, without acknowledgement, any material previously submitted for a degree or diploma in any university, and to the best of my knowledge and belief, it does not contain any material previously published or written by another person or myself except where due reference is made in the text. I also hereby give consent for my dissertation, if accepted, be made available for photocopying and for interlibrary loans, and for the title and abstract to be made available to outside organizations.

Candidate Name: : W.R.A.S. Kavishka

Sandali

Signature of Candidate

Date: June 28, 2025

This is to certify that this dissertation is based on the work of Ms. W.R.A.S. Kavishka under my supervision. The thesis has been prepared according to the format stipulated and is of acceptable standard.

Supervisor's Name : A.P. Sayakkara

Signature of Supervisor

Date: June 28, 2025

Abstract

Electromagnetic (EM) side-channel attacks present a critical threat to the security of air-gapped and physically isolated systems. While previous research has demonstrated the feasibility of near-field electromagnetic exfiltration from Ethernet cables using software-defined radios (SDRs), there remains limited exploration into the performance and practicality of such attacks from the far-field. This study investigates the technical feasibility, limitations, and performance of covert EM-based data exfiltration from Cat6 Ethernet cables using far-field antennas under controlled laboratory conditions.

The research aims to evaluate how effectively binary data can be transmitted through intentional EM emissions created via modulated network traffic patterns. Experiments were conducted to determine the maximum achievable data rate and to analyze how increasing the distance between the transmission medium and the receiver affects signal quality, Bit Error Rate (BER), and Signal-to-Noise Ratio (SNR). The methodology involved the use of SDRs, far-field antennas, and a custom-built encoder/decoder system incorporating Manchester encoding, Cyclic Redundancy Check (CRC), and Hamming error correction techniques to ensure reliable transmission and recovery of data.

Results demonstrate that reliable data exfiltration is achievable at a maximum rate of 10 bits per second at short distances (e.g., 2 cm), but signal quality and decoding reliability significantly degrade as distance increases, due to attenuation and environmental noise. The application of noise filtering through a Least Mean Squares (LMS) adaptive filter showed moderate improvement in SNR but was insufficient to overcome the limitations imposed by hardware sensitivity and signal strength. Furthermore, the study highlights the inadequacy of conventional shielding techniques in mitigating low-frequency emissions and emphasizes the need for proactive security measures.

This research contributes to the growing field of electromagnetic side-channel analysis by extending the attack surface beyond near-field threats. It offers insights into covert channel design, signal processing challenges, and mitigation strategies, raising awareness about the overlooked vulnerabilities of wired networks, even in physically isolated environments.

Contents

List of Figures

Acronyms

BER Bit Error Rate. iv, v, 3, 4, 10–12, 17, 26, 36, 40

CRC Cyclic Redundancy Check. 22, 23, 25, 32

EM Electromagnetic. 1–3, 7–12, 14, 16–19, 24–26, 28–31, 33, 34, 37, 39, 40, 45, 46, 48

EMR Electromagnetic Radiation. iv, 12, 14, 20, 21, 33

EMSC Electromagnetic Side-Channel. 7

FFT Fast Fourier Transform. 19

ICS Industrial Control System. 1, 2, 8

IOT Internet of Things. 1, 7

IPS Intrusion Prevention System. 1

LAN Local Area Network. 2

LMS Least Mean Squares. v, 26, 28, 39, 42–46

NIC Network Interface Card. 2

PLC Power Line Communication. 6

PSD Power Spectral Density. v, 31, 35, 38, 39, 42

RF Radio Frequency. 7

SDR Software Define Radio. 2, 8, 9, 12–14, 21, 34

SNR Signal-to-Noise Ratio. iv, v, 26, 28, 36–39, 41, 42

1 Introduction

1.1 Research Background

Data security is protecting data from unauthorized people stealing or accessing it. In today's world, it has become more critical because there are so many threats to attack sensitive data, such as data breaches, cyberattacks. Without solid data protection, people face significant risks, such as financial loss, privacy violations, reputational damage, and identity theft. Encryption, firewalls, anti-virus, anti-malware, passwords, authentication, and access control are some techniques that are commonly used to enhance data security. Nowadays, researchers have introduced more advanced, innovative solutions for data security in their investigations.

Air gapping is one such solution used for highly sensitive data and critical systems. In this case, it creates an isolated and secure environment by keeping a computer, network, or system separate from any other network without any external or public network connection. It provides security measures such as anti-virus software, network firewalls, Intrusion Prevention System(IPS), blocking USB and unwanted network ports to enhance protection. Air gapping techniques are utilized in many fields such as Military and Defence, Financial Institutions, Industrial Control System(ICS), Energy Sector, Government Sector, and so on (Park et al. (2023)).

Nevertheless, researchers have proven that these closed networks are no longer able to provide the best protection environment by investigating air-gap attacks. Air-gap attacks are a covert method of transmitting internal data from a closed network to external networks by utilizing signals that are generated by computer components, peripheral devices, or Internet of Things(IOT) devices within a closed network as covert channels. These attacks are also known as side-channel attacks as they utilize target system signals (Park et al. (2023)).

These attacks are performed to extract data based on examining signals on non-conventional communication channels within a system. Non-conventional communication channels can be built by exploiting the side effects of hardware or software behaviors such as EM radiation emitted by hardware, sounds generated by hardware components, response time for various inputs in a program, shared memory and cache spaces between different software applications, and electricity consumption of a computer (Sayakkara et al. (2019)).

The main focus of this research is on data exfiltration by using Electromagnetic(EM) radiation from wired Ethernet connections. Wired Ethernet is a technique that uses physical cables to connect all devices in a Local Area Network(LAN) for data transmission. The cables are connected to network devices such as routers, switches, or Network Interface Cards(NICs). Today, this technology is widely used because it provides high security due to the utilization of physical wired medium for data transmission. However, investigations have proven that tools like Software Define Radio(SDR) can exfiltrate data successfully without physically accessing Ethernet cables.

Sachintha et al. (2023) has investigated data exfiltration from ICS using EM radiation of Ethernet cables as a covert channel. They also investigated covert channel protocols and error correction codes to make EM-based covert channels more reliable. They introduced a method to detect automatically EM frequencies that can be leaked information via Ethernet cables.

All of these researches mainly install malicious software into target devices and read and modulate data into EM signals, then use a near-field antenna and SDR device to detect and decode signals that emit from Ethernet cables. The far-field antenna can also be used to detect signals from Ethernet cables from a distance.

Due to covert channels being more reliable at a low data rate, attackers need more time to extract information and need to know about the target hardware. However, this is not possible in most real-world scenarios.

1.2 Research Problem and Questions

Based on various investigations, data exfiltration methods by analyzing EM radiations from Ethernet cables, cause security threats to wired networks. Unfortunately, there is a poor understanding of the efficiency and effectiveness of these techniques. This study hopes to examine the maximum achievable data rate for such exfiltration and how the distance between the Ethernet network and the far-field antenna affects this process using Cat6 cables.

Question 1: What is the maximum data rate achievable for exfiltration using far-field antennas and Cat6 cables?
Hypothesis: The maximum data rate for data exfiltration utilizing far-field antennas will be significantly lower than the data rate achievable within the wired Ethernet network itself due to environmental interference and signal degradation.

Question 2: How does distance from the Ethernet network to the far-field antenna impact the effectiveness of data exfiltration using Cat6 cables?
Hypothesis: Because of signal attenuation and increased noise, the Bit Error Rate (BER) will be increased, so the effectiveness of data exfiltration will decrease exponentially with increasing distance between the Ethernet network and the far-field antenna.

1.3 Research Aim and Objectives

Research Aim

This research aims to assess the technical feasibility and limitations of data exfiltration from wired Ethernet networks using far-field antennas. It focuses on the minimum BER with achievable data rates and the impact of distance on exfiltration effectiveness using Cat6 cables.

Research Objectives

- To identify the minimum BER that may be achieved, under optimal conditions, for data exfiltration with far-field antennas and Cat6 cables.
- To examine the relationship between data exfiltration efficiency and the distance between the far-field antenna and the Ethernet network.
- Improve the performance of the covert channel.

1.4 Scope and Delimitations

In Scope

This research project intends to cover the following areas:

- Focus on the technical aspects of data exfiltration from wired Ethernet networks using far-field antennas and Cat6 cables.
- Limit experiments to controlled environments to ensure accurate measurement and repeatability (remove unnecessary wires and other sources).
- Analyze data exfiltration under optimal and varying distance conditions.

Out of Scope

The following lists the areas that do not come under the scope of the research:

- Will not cover data exfiltration methods involving wireless networks or other physical mediums.

- Only commercially available far-field antennas and standard Ethernet cables (Cat6) will be used.

- Environmental factors, such as extreme weather conditions, will not be considered

2 Literature Review

This section provides prior studies related to this research. These studies have examined different aspects of the research area and contributed important insights and advancements.

2.1 Covert Channel for Data Leakage

As discussed in the previous section, various non-conventional channels are available for data exfiltration as covert channels.

Sometimes, Digital devices emit optical signals that can act as a covert channel. Loughry & Umphress (2002) study has explained how can extract data by using optical radiation from LED status in digital devices and the reasons for increasing such attacks. Also, it has been shown that attackers can exfiltrate data successfully using LED status from 10m to 30m distance between the located device and the detector.

2.2 Air-gapped Covert Channels

Guri et al. (2018) has explained the data exfiltration method from two or more air-gapped computers within the same room by examining ultrasonic waves that can be considered as a covert channel. They found that speakers, headphones, and earphones responded properly under the 18kHz - 24kHz near-ultrasonic range. Their work also, showed that data transmission between two air-gapped computers can be done properly by using speaker-to-speaker communication under a 9m distance between computers and by using headphone-to-headphone communication under a 3m distance between computers. The data exfiltration process can be executed by examining audio signals that air-gapped computer speakers generate.

Nowadays, Power Line Communication (PLC) is a commonly used data transmission method that uses power lines as a physical medium. Guri et al. (2020) has introduced a new attack called PowerHammer to exfiltrate data from air-gapped computers by controlling and managing the system's power utilization. Data can be modulated by changing power usage and transmitted from the power supply via power lines. An attacker can capture emissions on power lines and extract data. In this paper, they discussed this covert channel method for desktop

PCs, servers, and low-power IOT devices and introduced two power hamming methods. *Line level Power-Hamming*: exfiltrate data by analyzing direct power lines of computers, *Phase level Power-Hamming*: analyzing primary electricity service panel of the building.

2.3 Electromagnetic Side-Channels

Electronic devices emit unintentional EM radiation, which is known as Electromagnetic Side-Channels (EMSCs). While EMSCs can act as covert channels, attackers can exfiltrate data without direct access by analyzing this EM radiation.

MAGNETO is a technique for an air-gapped computer covert channel that has been introduced by Guri (2021*b*).In this method, a smartphone can receive data from nearby air-gapped computers even if the smartphone is located in no wireless communication. By examining CPU cores, they have introduced malware that has the ability to control and manage magnetic fields in air-gapped computers. While sensitive data can be transmitted by using this magnetic field, magnetic sensors in smartphones can capture these signals. Also, this study has proven that this covert channel shows the best performance under short distances, low bitrates, and wireless communication blocked environments (e.g. turn on airplane mode on the smartphone or keep it on a Faraday bag).

USBee is a software that can convert an unmodified USB into a short-range Radio Frequency (RF) (RF) transmitter without firmware or USB hardware modification. It can manage EM radiation that emits from the data bus in a USB connector and transmit controlled radio signals to nearby receivers. It generates manageable EM radiation when data is transmitted from the computer to the USB, then a nearby Radio RF can capture EM radiation and extract data (Guri et al. (2016)).

Guri (2023) has introduced an air-gapped EM covert channel method. In this method, injecting malware into air-gapped computers that can generate radio waves on the device, then malware can control the internal processes of a computer and emit EM radiation with low frequency, then nearby receivers can detect these signals and extract data.

Besides above mentioned studies, many investigations have discussed EM covert channels in air-gapped environments.

2.4 Electromagnetic Covert Channels on Wired Networks

This section discusses the behavior of EM covert channels on wired networks.

LENTENNA is the most recently found EM attack that can exfiltrate data from air-gapped networks. After injecting malicious codes into air-gapped computers, exfiltrate data by examining EM radiation on Ethernet cables by using SDR, and Ethernet cables act as antennas. They have introduced two methods of signal generation called network speed toggling and UDP packet transmission. Also, they discussed exfiltration methods and investigated covert channel behaviour by Guri (2021*a*).

Schulz et al. (2016) is another study that described EM covert channel. They discussed the ability of eavesdropping against wired Ethernet environment by off-the-shelf SDR platforms. They show that Twisted-pair cables emit enough EM signals for successful data exfiltration and eavesdropping is possible without any trace on the cables but the attack success rate depends on cable shielding. They have used near-field or H-probs to capture EM signals. To avoid such attacks, shielded cables should be used.

Sachintha et al. (2023) has investigated a data exfiltration method in ICS under wired Ethernet environment. They used Ethernet cables as covert channels and exfiltrated data by analyzing EM radiation from Ethernet cables. In this method, inject malware into the target ICS device, and that malware steals sensitive information from the target device and converts it into Ethernet cable EM radiation. By using HackRF One SDR, detect EM signals then an attacker can extract data. They also discussed a covert channel protocol and error correction codes that increase the reliability of the covert channel. And also, introduced EM that information leaking frequencies detection methods from Ethernet cables.

2.5 Summary

In conclusion, while past research has explored various methods of data exfiltration from air-gapped environments using channels like audio signals, power lines, LED status, and Electromagnetic (EM) emissions, limitations remain. Many studies on EM side-channel attacks in wired networks have used methods like malware injection and eavesdropping to generate EM signals on Ethernet cables, often relying on SDR devices and near-field antennas for signal detection. While previous studies have extensively examined the threats to air-gapped systems and the mechanisms of covert channels, they often fail to achieve high data rates necessary for efficient and practical data exfiltration via EM emissions. Moreover, most existing work focuses on near-field scenarios, limiting the applicability of these techniques in real-world attacks. This research aims to bridge this gap by exploring methods to enhance data exfiltration performance over longer distances using far-field antennas. The focus is on improving the speed and reliability of EM-based covert channels, pushing the boundaries of current capabilities in air-gapped data leakage.

3 Research Methodology and Design

3.1 Research Methodology

The research will follow an experimental procedure. The following Figure 3.1 outlines the major steps that will be achieved during the research.

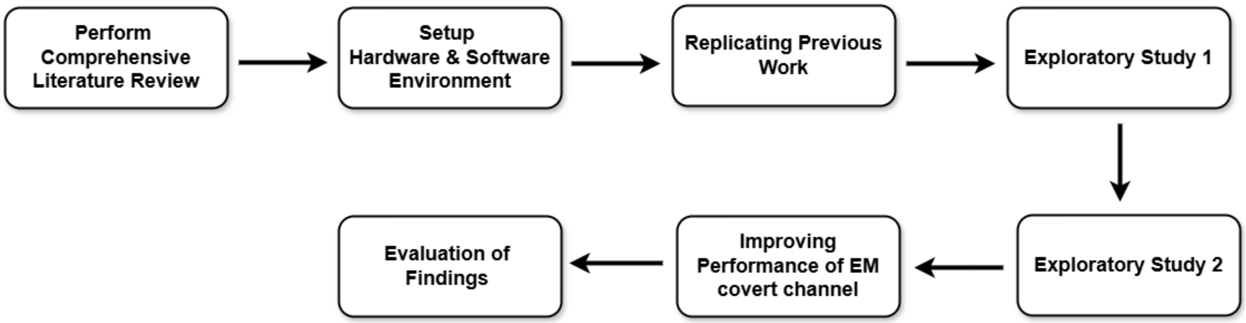
Perform Comprehensive Literature Review
Setup Hardware & Software Environment
Replicating Previous Work
Exploratory Study 1
Exploratory Study 2
Improving Performance of EM covert channel
Evaluation of Findings

Figure 3.1: Major steps of research process

The above-mentioned major steps in the research process are briefly discussed below.

- **Literature Review:** Examine the body of research on data exfiltration strategies, far-field antenna technologies, and vulnerabilities in Ethernet networks.
- **Hardware/Software environment Setup:** To simulate data exfiltration, design and implement a controlled hardware/software environment using far-field antennas and wired Ethernet networks.
- **Replicating Previous Work:** Involves recreating previously discussed EM covert channel setup with near-field antenna and Cat cables. This is intended to provide a better understanding of previous work and the shortcomings of them.
- **Exploratory Study 1:** This study focuses on examining the BER for a fixed hardware setup and a constant antenna distance. By keeping the antenna distance unchanging, the experiment aims to assess how well the far-field antenna can detect data exfiltration signals in a stable environment. This provides a baseline measurement of BER performance without the variable of changing antenna distance.

- **Exploratory Study 2:** This study builds on the findings of Study 1 but introduces varying antenna distances while keeping the hardware setup constant. The goal is to determine how the distance between the antenna and the target affects the minimum achievable BER. By analyzing BER at different distances, this study aims to identify the optimal range for effective data detection while evaluating the impact of distance on signal quality.

- **Improving Performance of EM covert channel:** Designing and applying methods to filter noise, provide better data encoding. to improve the performance of the EM covert channel

- **Evaluation of Findings:** Experimentally evaluating the results and making decisions

3.2 Research Design

3.2.1 Research Approach

In traditional Ethernet communication, data is transmitted by encoding and modulating electrical signals through Ethernet cables, with demodulation at the receiver end to decode the information.

In contrast, the EM covert channel leverages EM radiation emitted during Ethernet data transmission, rather than directly using electrical signals. By creating distinct EM patterns from specific traffic patterns, binary data can be encoded, requiring at least two unique EM symbols to represent binary 1 and 0. The research experiments are divided into two main categories:

Category 1: Focuses on EM signal capturing using a far-field antenna and establishing the relationship between EMR patterns and traffic patterns. (Data Collection and Analyzing)

Category 2: Focuses on developing the covert channel with minimum BER using special techniques such as noise cancellations and better error correction methods. (Attack Model Development)

3.2.2 Identify Emitting Frequency and Traffic Patterns

This phase focuses on identifying Cat6 cable emitting frequency and selecting the most suitable traffic patterns that are produced clearly distinguishable EMR patterns.

EM radiation variations can be detected with traditional signal analysis tools like oscilloscopes. Alternatively, Software Define Radio(SDR) provides greater flexibility and ease of use. SDR digitizes EM signals through high-speed Analog-to-Digital Converters (ADCs), allowing real-time visualization and processing through software.

Hardware Setup

The hardware setup designed for this research, as illustrated in Figure 3.2 below, includes these components:

- Ethernet Cable (Cat6)
- Transmitting Computer (Computer 1)
- Receiving Device
- Software Define Radio (SDR) Device
- Far-Field Antenna
- Capturing Computer (Computer 2)

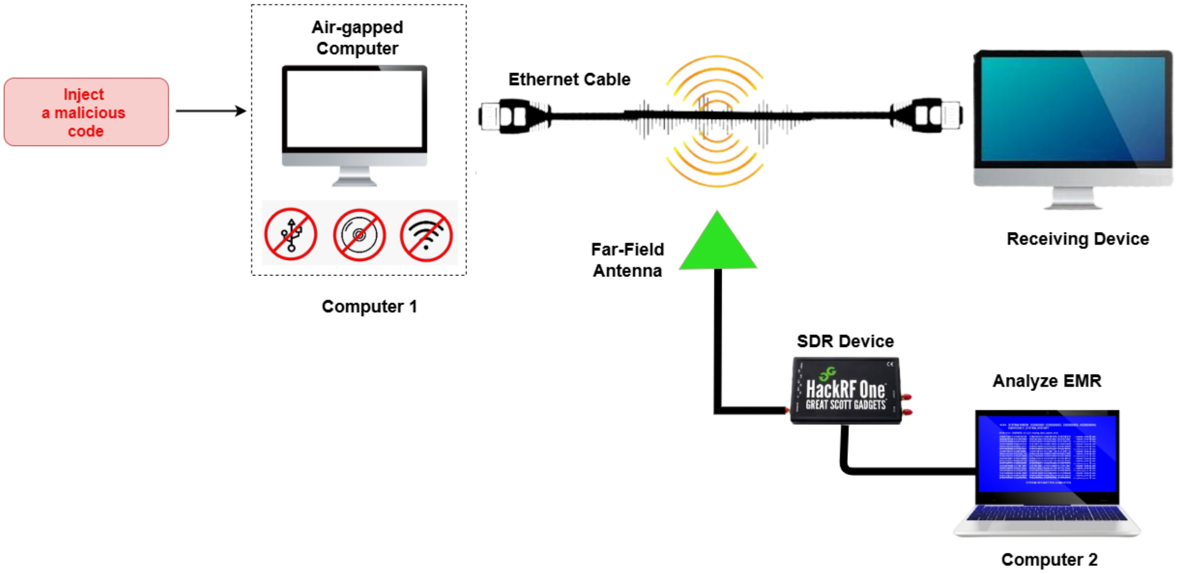

Figure 3.2: High-level diagram of the research design

In the experimental setup, two primary devices a transmitter (Computer 1) and a receiver are connected via a Ethernet cable (Cat6) . The transmitter generates network traffic, which is transmitted through the cable to the receiver.

To analyze the EMR emissions from the Ethernet cable, a SDR device (HackRF) equipped with a far-field antenna is positioned to capture these emissions. The captured signal is then processed and analyzed by a capturing computer (Computer 2) to inspect and evaluate the leaked electromagnetic signals.

Data Collection and Analysis

The goal of the data collection and analysis phase in this research is to identify distinct EM patterns and their associated emitting frequencies to support the implementation of the covert channel. Multiple frequencies may produce intended EMR (Guri (2021*a*)). This detailed analysis aims to pinpoint the frequency with the largest amplitude variation corresponding to traffic changes.

Another key aspect is identifying traffic patterns that produce the most distinct EM traces, enabling their use as symbols for modulating binary ones and zeros. As data packets with known payloads are transmitted from the sender to the receiver over the Ethernet cable, EMR is captured at various frequencies. The EM traces of two patterns are then compared using cross-correlation to measure their dissimilarity. This process is repeated for all patterns across each frequency, and the pattern pair with the greatest dissimilarity at a specific frequency is selected.

3.2.3 Designing the Covert Channel

The covert channel consists of two main phases: Encoding and Decoding.

- **Encoding Phase**: Process and convert the original message into a specific traffic pattern and transmits it through the Ethernet cable. This manipulation of network traffic creates distinguishable EMR signals that can be captured.

- **Decoding Phase**: Perform at attacker's computer, extracts the embedded bit patterns from captured EMR signals. It then processes the extracted data to reconstruct the original message.

This covert communication method leverages intentional EM emissions from Ethernet cables, demonstrating the feasibility of data exfiltration without direct network access.

Encoding and Modulation

The Encoder consists of two key processes: Encoding and Modulation. During the encoding process, original message is encoded into a predefined frame structure consisting of three main components:

- **Preamble**: Used to identify valid frames associated with the covert channel, ensuring proper synchronization and detection.
- **Payload**: Contains the encoded data to be transmitted.
- **Error Correction Mechanism**: Employed to mitigate data loss, as retransmission is not possible in this simplex communication channel.

In this frame structure uses error correction and error detection techniques as well. Unlike traditional bidirectional communication systems, this covert channel operates in a simplex mode, meaning corrupted packets cannot be retransmitted. To enhance reliability, a validation checksum is incorporated within the predefined frame structure, ensuring error detection and data integrity.

This structured encoding and modulation approach optimizes the covert channel for robust and accurate data exfiltration via electromagnetic emissions.

A straightforward technique such as On-Off Keying (OOK) can be used to modulate data through the Ethernet cable. However, this approach has significant drawbacks. As transmission power attenuates along the length of the cable, the amplitude of the received signal at the attacker's setup may vary depending on their proximity to the cable (Guri (2021*a*)). Additionally, Variations in the data rate of the communication medium can occur, making it difficult for the attacker to determine the precise clock intervals for each transmitted bit.

To overcome these limitations, Manchester encoding offers a more reliable alternative (Smith (2022)). Unlike OOK, where information is conveyed through amplitude variations, Manchester encoding represents each bit as a transition in the signal. This ensures that variations in signal strength have minimal impact on decoding. Furthermore, the encoding scheme inherently provides clock synchronization, making it easier to accurately distinguish individual bits without requiring external timing references.

By adopting Manchester encoding, the covert channel becomes more robust and resistant to signal degradation, ensuring reliable data transmission through EM emissions.

Demodulation and Decoding

The Decoder consists of two key processes: Demodulation and Decoding.

At the attacker's end, demodulation and decoding are used to extract the leaked data. While speed is not a major concern, collecting clear EM traces is crucial for accurate data recovery. The captured EM signals must go through demodulation to reconstruct the transmitted frames. The preamble, which has a unique EM pattern, helps identify where each frame starts. Once the preamble is detected, the rest of the frame can be extracted.

After all frames are extracted, decoding is performed. This step includes error correction to fix possible mistakes and checksum verification to ensure the data is accurate. These processes help recover the transmitted information reliably, even if there are signal distortions or interference.

4 Implementation

4.1 Experiments

4.1.1 Replication of Previous Work

In this section, the details of the replicated attack of covert channel implementation as demonstrated in the previous work, where a near-field antenna was used to capture EM signals is presented. The previous approach involved transmitting data covertly by leveraging EM emissions from Ethernet cables, with the near-field antenna positioned close to the target device to capture the signal effectively. This setup allowed for high signal accuracy with minimal noise, essential for precise data extraction.

To replicate their experiment, a similar procedure was followed by positioning the near-field antenna in close proximity to the emitting source. This configuration allows for capturing EM signals with minimal interference, providing a stable channel for data decoding and analysis. The effectiveness of the near-field antenna setup is particularly evident in its lower BER, attributed to the reduced impact of external noise and interference. This replication provides a baseline for comparing near-field and far-field antenna performance in covert channel transmission.

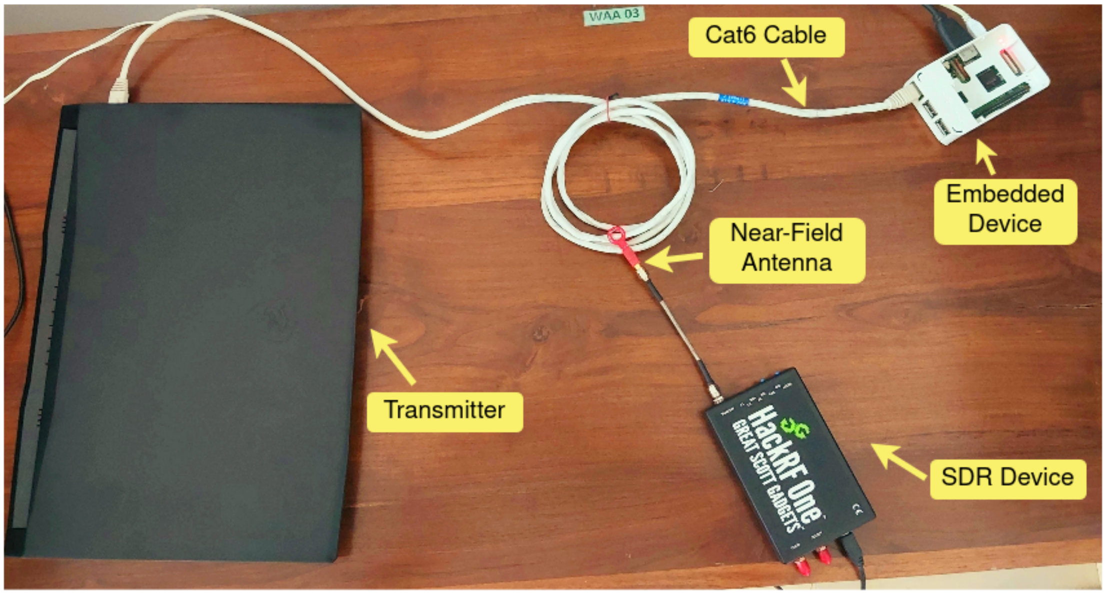

Figure 4.1: Hardware Setup for previous work

Device	HackRF One
Sampling Rate	20 MHz
RF Gain	10 dB
IF Gain	20 dB
BB Gain	20 dB
DC Correction	Disabled

Table 4.1: SDR device configuration details

4.1.2 Identifying Emitting Frequency

In this section, the emitting frequency of the Cat6 cable is identified.

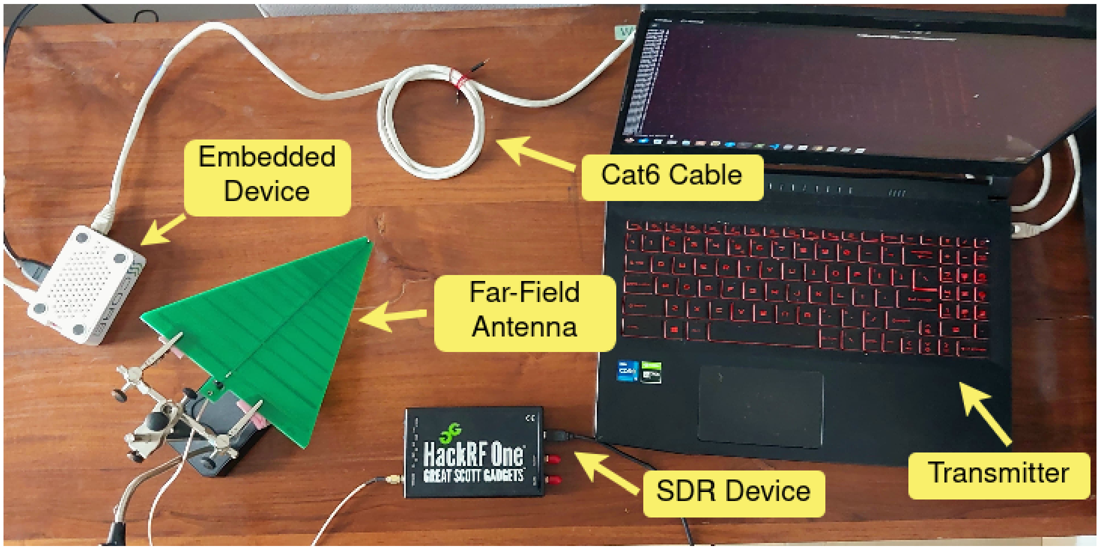

Figure 4.2: Hardware Setup with Far-field antenna

In this experimental Figure: 4.2, a measurement system is configured to detect EM emissions from a Cat6 Ethernet cable, with the objective of identifying both the emitting frequency and correlating it to network traffic patterns

Manual Inspection

In this experiment, GQRX is used to visualize real-time EM radiation captured from an Ethernet cable. The tool provides both Fast Fourier Transform (FFT) plots and spectrograms to detect variations in radiation patterns caused by packet transmissions. Due to background EM interference amplified by the Ethernet cable, careful manual inspection across the frequency range is necessary.

Figure: 4.3 and 4.4 illustrate the GQRX interfaces displaying frequency spectra for an Ethernet cable when the cable is idle and the cable is actively transmitting data. The observed frequency is centered around **125 MHz**.

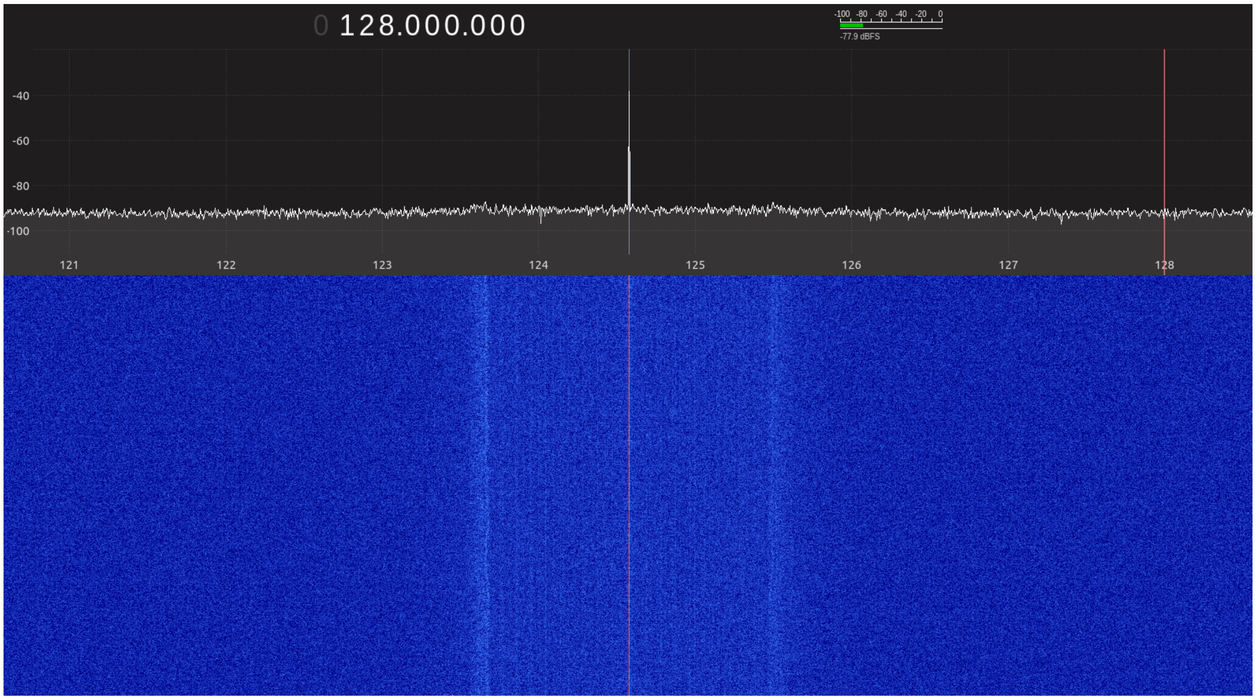

Figure 4.3: GQRX interface when Ethernet cable is idle

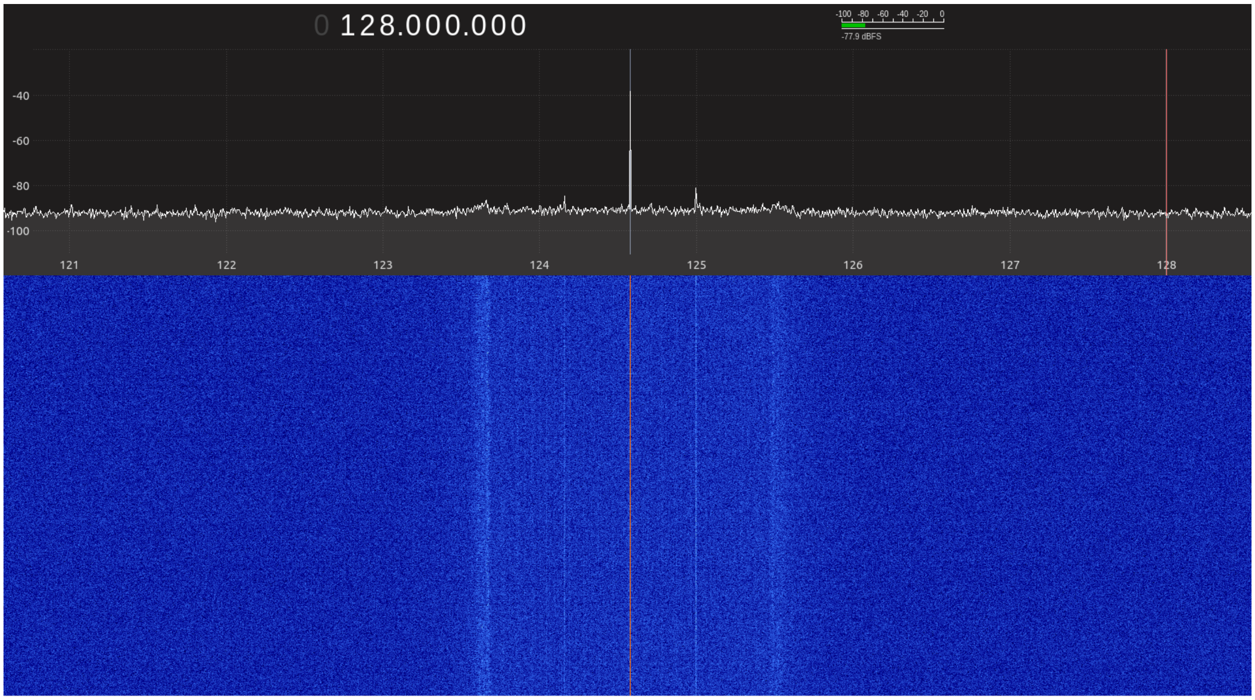

Figure 4.4: GQRX interface when Ethernet cable transmits the traffic

4.1.3 Identifying Traffic Patterns

In this section, identify the most suitable traffic patterns that are produced clearly distinguishable EMR patterns for representing 0 and 1s.

In line with previous work, Specific traffic patterns are selected, and examined their cross-correlations to determine their suitability for far-field antenna detection. By analyzing these patterns, aimed to assess whether the previously chosen traffic patterns maintain their effectiveness in a far-field setup, ensuring reliable detection and characterization of electromagnetic emissions under varying transmission conditions.

#	Pattern ID	Protocol	Payload
1	P0 (Idle)	None	None
2	P1	UDP	all 1
3	P2	TCP	all 1

Table 4.2: Traffic Patterns with Protocols and Payloads

Algorithm 4.1 Packet Pattern Simulation

```
Input: payloads[], protocols[], duration
Output: None
for protocol in protocols[] do
    for payload in payloads[] do
        endTime ← currentTime + duration
        while currentTime < endTime do
            packet ← definePackets(payload, protocol)
            sendPackets(packet)
        end while
    end for
end for
```

4.2 Covert Channel

During the data exfiltration process from an Ethernet cable, the victim device first converts the original data into an encoded format. The encoded data is then modulated using predefined traffic patterns that represent bit 0 and bit 1. Once modulated, the data is transmitted covertly to the receiver device.

A nearby receiver device with a SDR and a far-field antenna detects the intentional EMR emitted from the Ethernet cable. The captured signal is then processed through a demodulation stage, where the extracted bits are compared against the identified traffic patterns corresponding to bit 0 and bit 1. Finally, after decoding, the original message is reconstructed.

This entire process occurs within a covert channel and consists of two main components: Encoder and Decoder

4.2.1 Encoder

In this process, the original message is converted into a predefined frame structure. As discussed in Section 3.2.3, the frame consists of three main components: the preamble, the payload, and the error detection mechanism. Additionally, an error correction mechanism is incorporated to ensure data integrity. The key components of this structure are detailed below:

Preamble: This section is used to validate packets originating from the victim computer and to identify the frame. It consists of a fixed 8-bit sequence, "10101010", which serves as a synchronization pattern for accurate frame recognition.

Payload: The payload contains the data that needs to be leaked from the victim's device. The data is taken as ASCII characters, where each character is represented using 8 bits. The characters are grouped into chunks of three, meaning each chunk forms a 24-bit payload. The number of characters included in the payload can be adjusted as needed.

Error Detection: During transmission, the leaked data can be affected by errors, making error detection necessary. To ensure data integrity, a Cyclic Redundancy Check (CRC) is used to generate a checksum for the payload. This checksum is added at the end of the payload bits. When decoding the data, the payload is verified using CRC, and any corrupted payloads are discarded. CRC works by using polynomial division to create the checksum, which provides better error detection than regular checksums. It can detect multiple bit errors and help locate errors in the data. In this implementation, an 8-bit CRC is used for error detection.

Error Correction: To fix errors that happen during transmission, error correction techniques are needed. Hamming Code is one such technique that helps detect and correct bit errors by adding parity bits to the data. In this implementation, the Hamming(7,4) code is used. This means that for every 4 bits of data, 3 extra parity bits are added, making a 7-bit encoded block. These parity bits help detect and correct single-bit errors, improving data reliability.

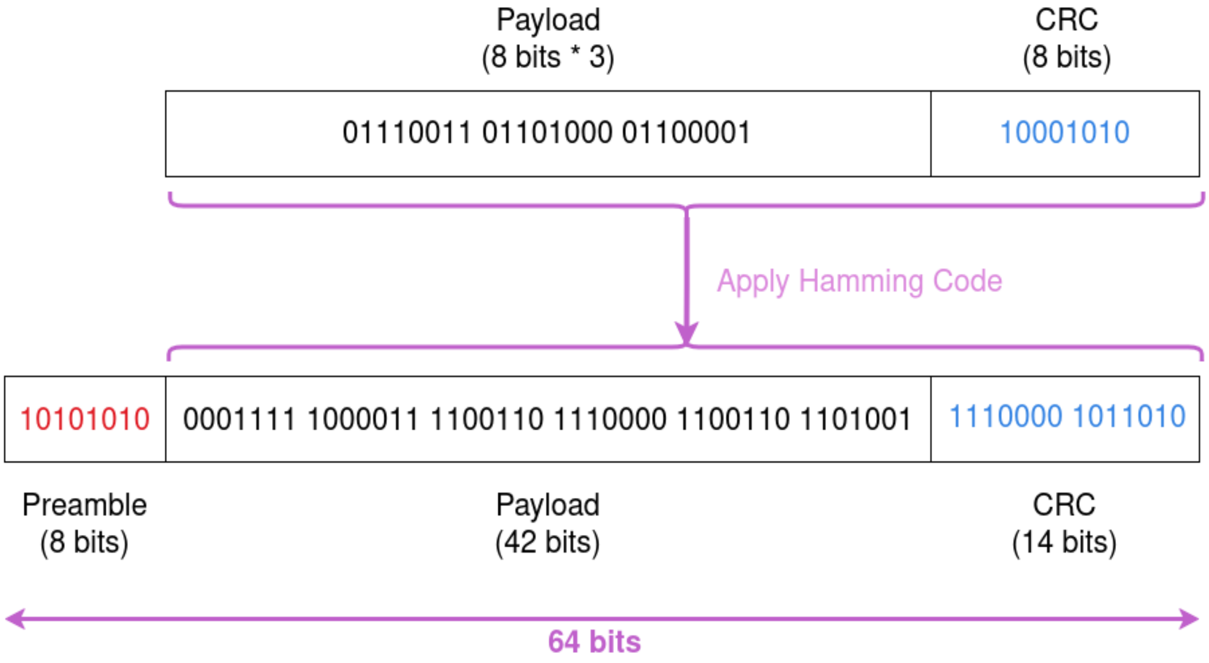

Figure 4.5: The frame structure of encoded data

The process of frame preparation, as outlined in Algorithm 4.2, begins by taking the input string (characters) that needs to be encoded. The characters are first separated to determine how they will be grouped within each frame (line 3). Subsequently, the binary representation of these characters is retrieved to facilitate encoding (lines 5-6). If necessary, padding bits are added to ensure proper alignment of the frame structure (lines 8-10). Once the binary data is prepared, a CRC is computed and appended to the payload to enable error detection. Both the payload and CRC are then encoded using Hamming code, which provides additional error correction capabilities (lines 11-13). Finally, to complete the frame, a preamble is added at the beginning, serving as a synchronization marker to aid in the detection and decoding process. This structured approach ensures the integrity and reliability of data transmission while preparing frames for encoding and transmission.

Algorithm 4.2 Algorithm for preparing the frame by encoding data

```
Input: characters, payloadSize
Output: Array of encoded frames
encodedFrames $\leftarrow$ []
while not characters.isEmpty() do
    charStream $\leftarrow$ characters.getCharsPerFrame()
    bitStream $\leftarrow$ []
    for char in charStream do
        bitStream[] $\leftarrow$ getBinary(char)
    end for
    if len(charStream) $<$ payloadSize then
        padding $\leftarrow$ addPadding(len(charStream) - payloadSize)
    end if
    payload $\leftarrow$ bitStream[] + padding
    crc $\leftarrow$ calcCRC(payload)
    hamCode $\leftarrow$ calcHam(payload + crc)
    frame $\leftarrow$ preamble + hamCode
    encodedFrames[] $\leftarrow$ frame
end while
return encodedFrames
```

Modulating Data

As highlighted in Section 3.2.3, Manchester encoding is utilized to modulate the data across the Ethernet wire, where binary 0 is represented by a falling edge, and binary 1 is represented by a rising edge (Refer to Figure 4.6). This encoding scheme ensures synchronization and reduces the likelihood of long sequences of identical bits, which helps in maintaining signal integrity. Furthermore, two distinct EM patterns, selected through dissimilarity analysis, are used to represent binary 0 and 1 during modulation. This approach leverages the inherent variations in EM emissions to encode data, making it suitable for covert communication or signal analysis applications.

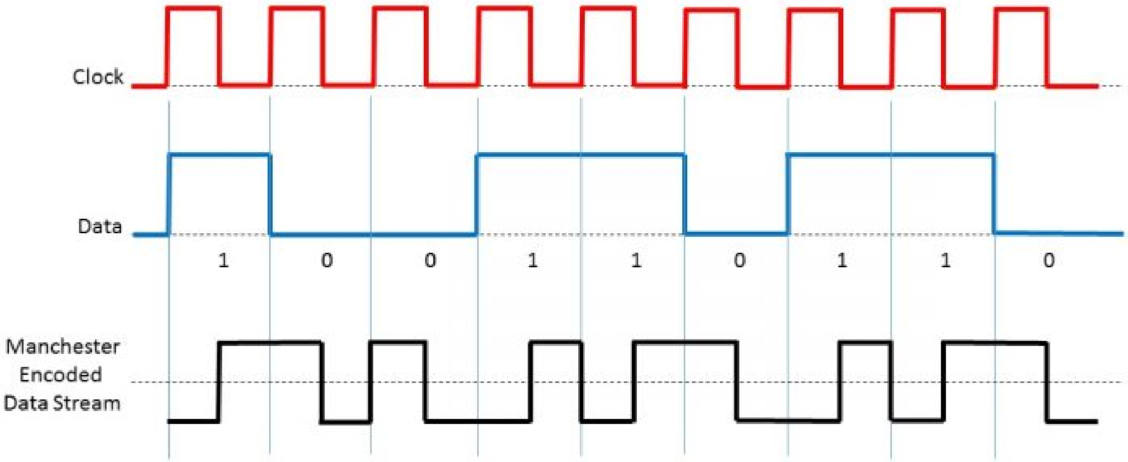

Figure 4.6: Manchester encoding applied to a data frame

4.2.2 Decoder

Data Extraction

The first step in reconstructing leaked data at the attacker's end involves capturing EM emissions at a selected frequency. Once captured, the EM patterns within the trace file are analyzed and demodulated into binary sequences of 0s and 1s. These binary sequences are subsequently decoded into characters, facilitating data extraction.

The main driver code responsible for initiating data extraction processes the captured EM trace data as input. This procedure is formally outlined in Algorithm 4.4. Each data frame begins with a predefined preamble sequence, 10101010, which serves as a synchronization marker. Therefore, the extraction process starts by verifying the presence of this preamble (lines 9-12). Once verified, the payload and CRC bits are demodulated (lines 14-16), followed by decoding to reconstruct the original characters (lines 19-21).

The following sections provide an in-depth discussion on the key steps involved in this process, including preamble detection, payload and CRC demodulation, and character decoding.

Noise Filtration

The first step in accurately demodulating bits from the captured EM data file involves determining whether the EM signal is affected by noise. This assessment is crucial, as noise can distort the transmitted signal, leading to errors in bit extraction and increasing the Bit Error Rate (BER).

To evaluate the impact of noise, the Signal-to-Noise Ratio (SNR) is analyzed. A high SNR indicates that the EM signal is strong relative to background noise, enabling precise demodulation. Conversely, a low SNR suggests significant interference, making it difficult to distinguish transmitted bits from random fluctuations. To ensure accurate bit demodulation, if the SNR is low, a noise filtering mechanism is applied to enhance signal clarity. In this study, utilizes the Least Mean Squares (LMS) Adaptive Filter to mitigate noise and improve demodulation accuracy.

The LMS Adaptive Filter is a widely used technique for noise reduction and signal enhancement. It works by iteratively adjusting its filter coefficients to minimize the error between the desired and actual output signals. This self-adjusting mechanism helps improve signal clarity while maintaining computational efficiency.

Compared to other filtering methods, the LMS filter offers several advantages. It is computationally simple, requires low memory, and is easy to implement. Additionally, it effectively reduces noise while preserving important signal characteristics, making it a preferred choice in many signal processing applications GeekForGeeks (2024).

In this algorithm:

- **Padding the Reference Signal:**

$$\hat{r} \leftarrow \text{pad}(r, M-1)$$

  - This equation pads the reference signal r with $M-1$ extra samples at the beginning.
  - Padding ensures that the sliding window has enough data for the filter to process the signal, even at the start.

Algorithm 4.3 LMS Adaptive Filter

```
Input: Reference signal $r[n]$, Target signal $t[n]$, Step size $\mu$, Filter order $M$
Output: Filtered output signal $y[n]$, Error signal $e[n]$
$N \leftarrow$ length of $t[n]$
weights $w \leftarrow 0$ (vector of size $M$)
output $y[n] \leftarrow 0$, error $e[n] \leftarrow 0$
Pad reference signal: $\hat{r} \leftarrow \text{pad}(r, M-1)$
for $n \leftarrow 0$ to $N-1$ do
    $x_n \leftarrow$ last $M$ samples of $\hat{r}$ in reverse order
    $y[n] \leftarrow w^T x_n$
    $e[n] \leftarrow t[n] - y[n]$
    $w \leftarrow w + \mu e[n] x_n$
end for
return $y[n]$, $e[n]$
```

- **Sliding Window Mechanism:**

$$x_n \leftarrow \text{last } M \text{ samples of } \hat{r} \text{ in reverse order}$$

  - This extracts the last M samples from the padded reference signal $\hat{r}$, and reverses their order.
  - The reversed samples form the input for the filter, simulating the sliding window mechanism by always using the most recent M samples for each filtering step.

- **Filter Output Calculation:**

$$y[n] \leftarrow w^T x_n$$

  - The dot product of the input vector x_n and filter weights w generates the estimated noise component $y[n]$.
  - This output represents the predicted signal based on the current filter coefficients.

- **Error Computation:**

$$e[n] \leftarrow t[n] - y[n]$$

  - The error signal $e[n]$ is calculated by taking the difference between the target signal $t[n]$ and the estimated output $y[n]$.
  - This error reflects how much the filter output deviates from the desired target.

- **Weight Update:**

$$w \leftarrow w + \mu e[n] x_n$$

  - The filter weights w are updated using the LMS update rule, which adjusts them based on the error signal $e[n]$ and the input vector x_n.
  - The step size μ controls the rate of weight adjustment to minimize the error.

- **Iteration:**
  - The process continues for all samples in the signal, progressively refining the filter coefficients to improve noise suppression.

In this study, the reference signal is the EM signal generated during data traffic in the Ethernet cable, while the noise signal represents the EM emission when the cable is idle. To optimize the performance of the LMS filter, we need to evaluate different values of the step size μ and filter order **M**. The step size controls the convergence speed of the filter, while the filter order determines how many past samples are used to compute the filter output. By experimenting with different combinations of these parameters, we aim to find the optimal values that effectively suppress noise while maintaining the integrity of the reference signal. This will be assessed by comparing the filtered signal with the target signal and analyzing the resulting error and SNR.

Detecting Preamble

Since the attacker lacks prior knowledge of when the transmitter begins leaking data, the start times of data leakage and EM trace collection are not synchronized. Consequently, detecting the preamble is essential to ensure proper alignment for demodulating the subsequent data bits.

To achieve this, a sliding window approach is employed within the detectPreamble() function. For each window, the corresponding EM sample is processed using Algorithm 4.5. Once a demodulated window matches the predefined bit pattern 10101010, the variable preambleVerified is set to 1, allowing the demodulation of the remaining frame to proceed.

The following sections further elaborate on the sliding window implementation, its role in synchronizing the demodulation process, and its effectiveness in mitigating timing uncertainties in EM-based data exfiltration.

Demodulating Bits

Both the driver code in Algorithm 4.4 and the detectPreamble() function invoke the demodulateBit() function, as detailed in Algorithm 4.5. This function processes an EM sample corresponding to a single bit duration and determines its binary value. Since Manchester coding is used for data modulation, each bit consists of two equal halves: one with high amplitude and the other with low amplitude. A rising edge (low-to-high transition) represents binary 1, while a falling edge (high-to-low transition) represents binary 0 (refer Figure 4.7).

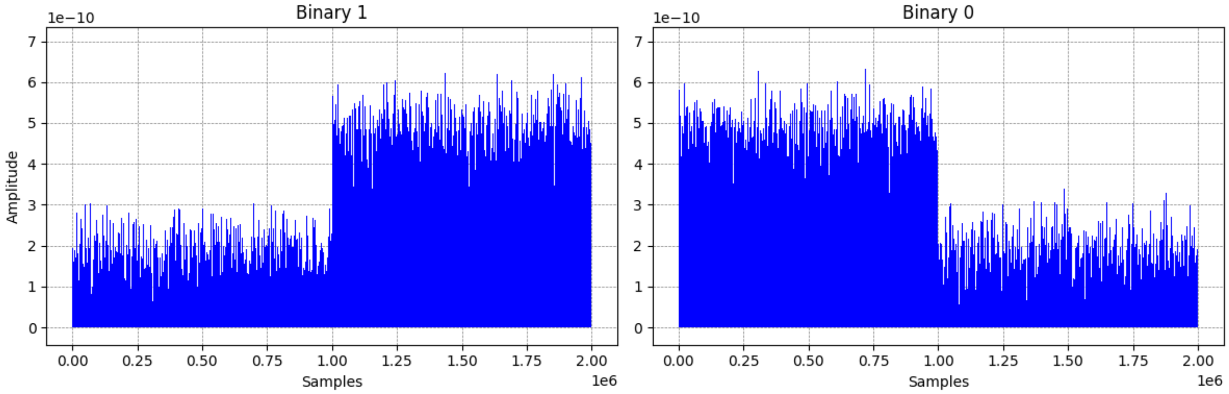

Figure 4.7: Comparison of Manchester encoded binary 0 and binary 1

Algorithm 4.4 Algorithm for extracting data from a captured EM signal

```
Input: sampleRate, emSignalData, biTime, preambleSize, bitCountForExtraction
Output: Array of decoded characters
samplesPerBit ← sampleRate × biTime / 1000
samplesPerPreamble ← preambleSize × samplesPerBit
blockSize ← 0
extractedBitCount ← 0
decodedChars ← []
extractedFrame ← []
preambleVerified ← False
bitStream ← []
while blockSize < len(emSignalData) do
    if preambleVerified = False then
        preamble ← emSignalData[blockSize : blockSize + samplesPerPreamble]
        blockSize ← blockSize + samplesPerPreamble
        preambleVerified ← detectPreamble(preamble)
    else
        bit ← emSignalData[blockSize : blockSize + samplesPerBit]
        blockSize ← blockSize + samplesPerBit
        extractedFrame ← extractedFrame+ demodulateBit(bit)
        extractedBitCount ← extractedBitCount + 1
        if extractedBitCount = bitCountForExtraction then
            correctedFrame ← correctErrors(extractedFrame)
            payload ← extractPayload(correctedFrame)
            extractedChars ← decodeChars(payload)
            decodedChars ← decodedChars + extractedChars
            extractedBitCount ← 0
            extractedFrame ← []
            preambleVerified ← False
        end if
    end if
end while
return decodedChars
```

To identify high- and low-amplitude segments, the Welch method is employed. This method estimates the Power Spectral Density(PSD) for two halves of the signal (lines 2 and 3 in Algorithm 4.5). The estimated PSD values are then compared against predefined thresholds to classify the EM sample into a binary value representing the bit (lines 4–9). This approach ensures a systematic evaluation of amplitude variations, enabling robust bit classification in the presence of noise.

Algorithm 4.5 Algorithm for demodulating bits

```
    Input: bitSample, bitTime, higherThreshold, lowerThreshold
    Output: bit derived from bitSample
 1: middlePoint ← bitTime / 2
 2: firstHalf ← bitSample[0 : middlePoint]
 3: secondHalf ← bitSample[middlePoint : bitTime]
 4: firstPSD ← calcWelch(firstHalf)
 5: secondPSD ← calcWelch(secondHalf)
 6: if firstPSD > higherThreshold AND secondPSD < lowerThreshold then
 7:     return 0
 8: else if firstPSD < lowerThreshold AND secondPSD > higherThreshold then
 9:     return 1
10: else
11:     return None
12: end if
```

Algorithm 4.6 Algorithm for decoding payload

```
Input: payload, paddingChar
Output: Decode characters from payload
binarySegments[] ← getChunksOfChars(payload)
decodedChars ← []
for binarySegments[] do
    if char paddingChar then
        decodedChars[] ← getAsciiChar(char)
    end if
end for
return decodedChars
```

After demodulating the frames, error correction is performed using Hamming code to fix any single-bit errors. The payload is then verified for integrity using CRC. Once verified, the payload is decoded into ASCII characters using Algorithm 4.6, which divides the payload into 8-bit chunks, decodes each chunk into an ASCII character, and returns the resulting characters as the decoded message. This process ensures reliable data recovery and accurate interpretation.

Please refer to the GitHub repository for the full code base of the implementations and the analysis scripts.[1]

[1] GitHub repository: https://github.com/SandaliKavi99/EM-Covert-Channel.git

5 Result and Discussion

This chapter presents the research flow by integrating the results obtained from the experiments detailed in Sections 3 and 4. It provides an in-depth discussion of the findings from each experiment and the corresponding decisions made. Additionally, it offers a comprehensive analysis of EM-based Ethernet covert channels, evaluating performance and examining the impact of probe positioning variations

5.1 Electromagnetic Radiation (EMR) Datasets

Throughout this research, multiple datasets were gathered for various objectives, including performing dissimilarity analysis of packet patterns, performing covert channel and analyzing noise filtration effectiveness. Table 5.1, below represents the Specification of Devices that are used for experiments.

#	Device Name	Hardware Specification	Operating System
1	**Laptop (Transmitter)**	MSI Katana GF66 11SC 11th Gen Intel(R) Core(TM) i7-11800H @ 2.30GHZ,2304Mhz, 8Core(s) 24 GB (8GB + 16GB) DDR4 RAM KINGSTON OM8PCP3512F-A11 512GB SSD NIC: ASIX USB to Gigabit Ethernet Family Adapter	Windows 11, version 23H2
2	**Embedded Device (Receiver)**	Raspberry Pi 3 Model B+ ARM Cortex-A53 CPU @ 1.4GHz 1GB LPDDR2 RAM NIC: Microchip LAN7515	Raspberry OS 2023-02-21

Table 5.1: Summary of Device Specifications

During the dataset collection process, packet patterns were sent from the transmitting computer to the receiver device through an Ethernet cable. A far-field antenna, connected to an SDR(HackRF One), was placed with a distance from the cable to capture EM emissions during transmission. The SDR was linked to a capturing computer, which recorded and stored the collected data in IQ data files.

5.2 Dissimilarity of Packet Patterns

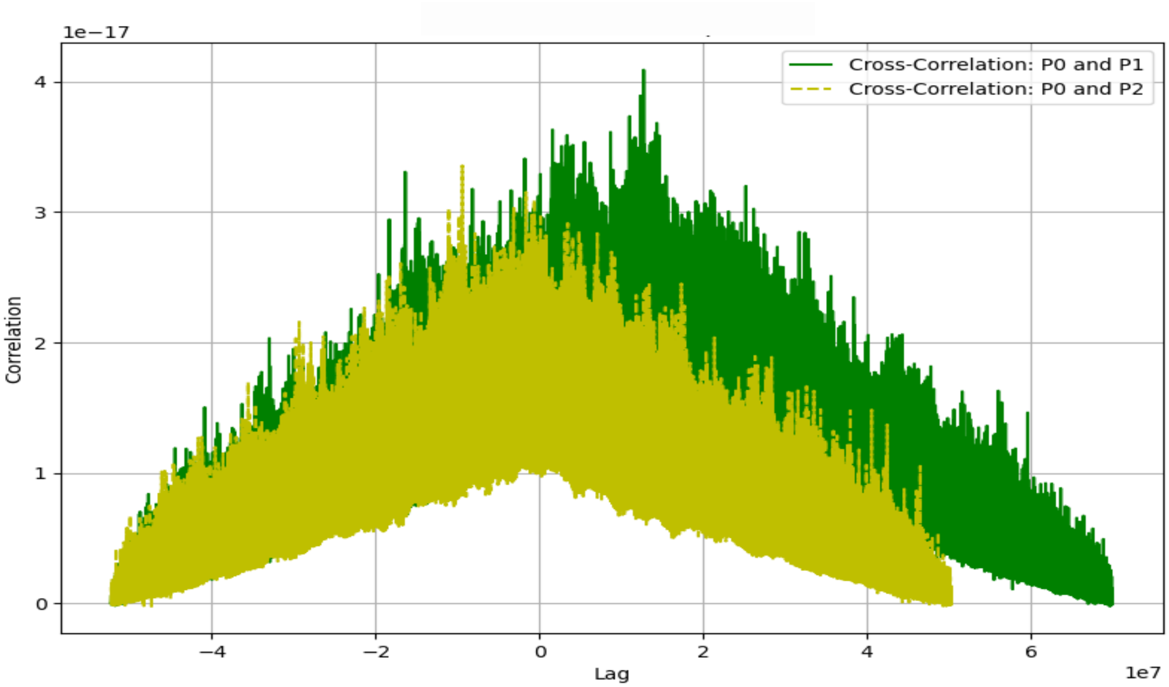

Figure 5.1: Cross-Correlation Comparison of P0 with P1 and P2

In previous studies, various patterns of network activity (such as P0, P1, P2, and others) were evaluated to determine their effectiveness in representing binary values 0 and 1. As discussed in Section 4.1.3, three specific traffic patterns were identified to have higher cross-correlation values compared to others, indicating stronger consistency and reliability. In this work, further examined whether these patterns remain effective under far-field antenna experiments.

Figure 5.1 presents the cross-correlation comparison, a technique used to measure the similarity between two time-based signals. The results confirm that P0 and P1 exhibit the most distinct and stable characteristics, making them well-suited for representing binary data. Therefore, in this study, for modulation process, have used UDP packet transmission to represent a bit 1 and an idle state (i.e., no packet transmission) to represent a bit 0. This approach ensures consistent transmission timing while allowing accurate bit detection based on the electromagnetic emissions generated by each traffic pattern.

5.3 Varying Distance Between Probe and Ethernet Cable

Figure 5.2 illustrates how PSD values change with distance at a fixed data rate of 10 bps. The graph shows PSD curves at various distances ranging from 2 cm to 14 cm. It is clearly observed that as the distance increases, the overall PSD values decrease. For instance, the PSD measured at 2 cm (blue curve) is noticeably higher than that at 14 cm (yellow curve). This pattern confirms the expected behavior where electromagnetic signal strength weakens with increasing distance due to signal attenuation in space. As a result, the energy captured by the antenna reduces, causing lower PSD values at farther distances. This reduction in PSD can make it more challenging to distinguish bit patterns.

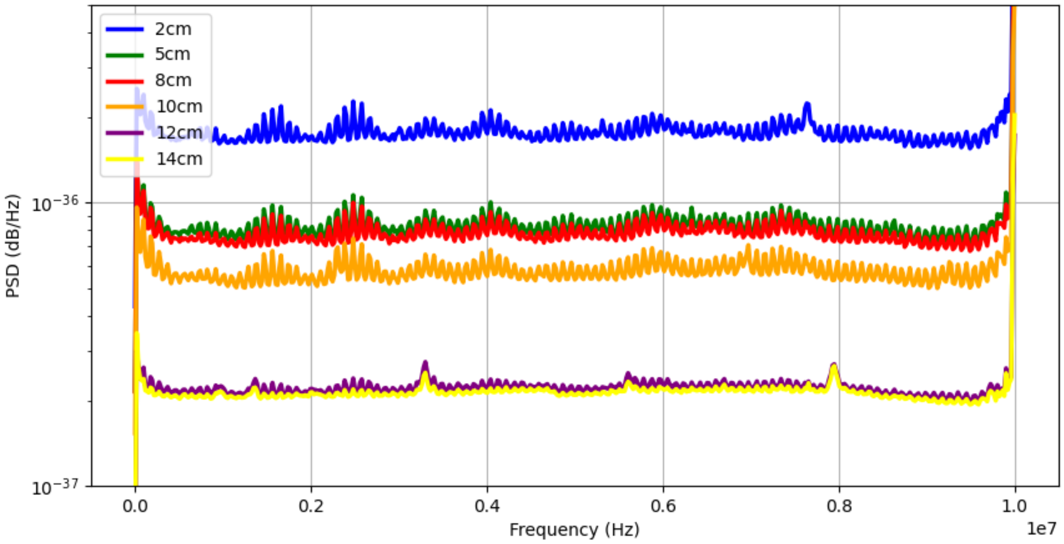

Figure 5.2: PSD values with distance under 10bps data rate

5.4 Performance Evaluation of Covert Channel

5.4.1 Different Metrics Used

Bit Error Rate

Bit Error Rate(BER) represents the percentage of bit flips that occurred during transmission in relation to the total number of bits sent. It serves as an indicator of the quality of a communication channel. Equation (5.1) provides the formula for calculating the channel's BER.

$$BER = \frac{\text{Hamming weight of } (M_{\text{Sent}} \oplus M_{\text{Received}})}{\text{Total number of bits transmitted}} \times 100\% \tag{5.1}$$

In this formula:

- $\oplus$: Represents the XOR operation.
- M_{Sent} represents the bit sequence that was transmitted.
- M_{Received} represents the bit sequence that was received.
- The Hamming weight of $(M_{\text{Sent}} \oplus M_{\text{Received}})$ measures the number of differing bits between the transmitted and received sequences, reflecting the number of bit errors.
- The total number of bits transmitted refers to the overall length of the bit sequence.

Signal-to-Noise Ratio (SNR)

The Signal-to-Noise Ratio (SNR)(SNR) measures the quality of a signal by comparing the power of the desired signal to the power of the background noise. A higher SNR signifies a clearer, higher-quality signal. Equation (5.2) represents the formula for calculating the SNR in decibels.

$$SNR(\text{dB}) = 10\log_{10}\left(\frac{P_{\text{Signal}}}{P_{\text{Noise}}}\right) \tag{5.2}$$

In this Equation:

- P_{signal} represents the power of the desired signal, indicating the strength of the transmitted signal.

- P_{noise} denotes the power of background noise, signifying the level of unwanted interference in the signal.

The EM of the covert channel is calculated by comparing two EM samples: one recorded during the transmission of the desired signal (P_{signal}) and another captured when the medium is idle (P_{noise}), representing the background noise level (NextPBC (2024)).

5.4.2 Data Rate with Varying Probe Distance

In this study, three primary data rates were evaluated: 10 bps, 11 bps, and 12 bps. At 10 bps, the covert channel operated effectively; however, its performance declined as the antenna distance increased from 2cm to 14cm.

For 11 bps and 12 bps, the covert channel failed to function correctly due to the limited sensitivity of the antenna, which was unable to capture EM emissions accurately at these higher data rates. The following list presents the SNR values for 11bps and 12bps:

Data rate	2cm	5cm
11 bps	1.81 dB	1.09 dB
12 bps	0.98 dB	0.47 dB

Table 5.2: SNR Values with data rates

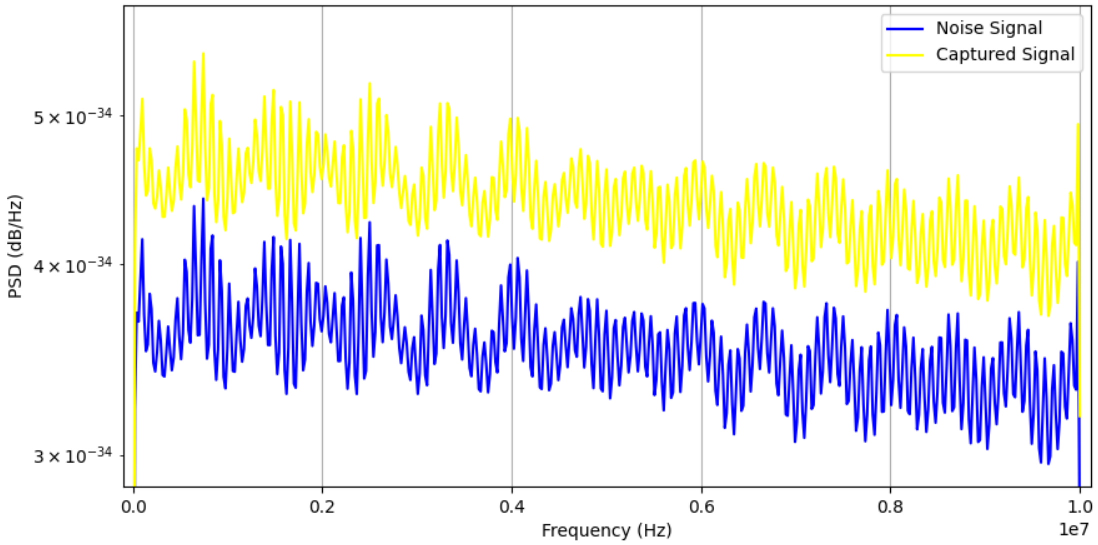

Figure 5.3: PSD with frequency under 11bps data rate (Zoomed In)

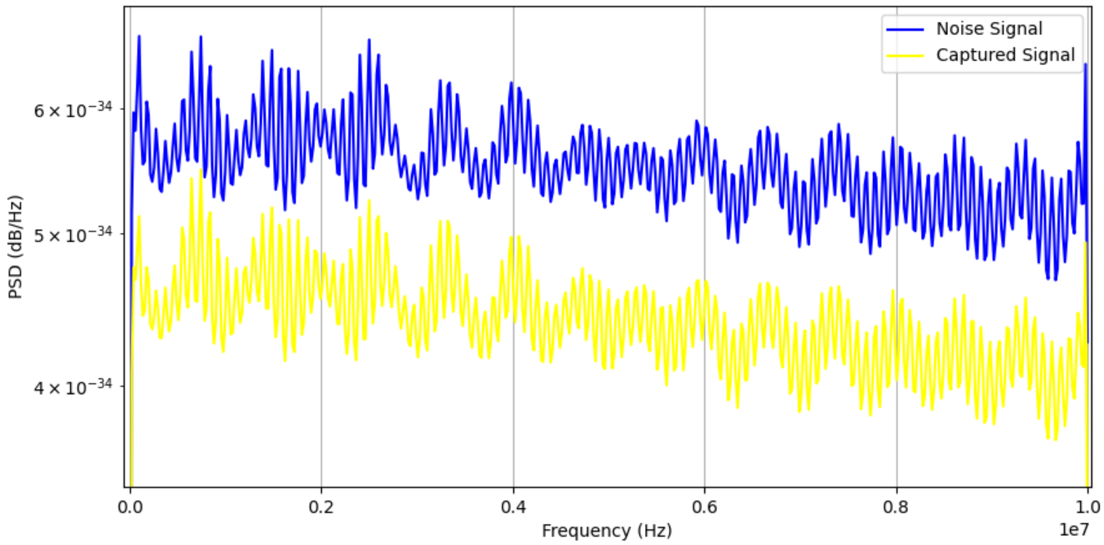

Figure 5.4: PSD with frequency under 12bps data rate (Zoomed In)

Although Figure 5.3 and Figure 5.4 clearly illustrate that the captured signal can be visually distinguished from the background noise, the corresponding SNR values are still relatively low (refer Table: 5.2). This suggests that while the signal is present and can be detected, its strength relative to the noise is not sufficient to ensure robust detection under varying real-world conditions. Importantly, the noise observed in these figures does not appear to directly overlap or distort the signal, which implies that the issue is not due to external interference but rather the weak intensity of the EM emissions themselves. As a result, the low SNR could be attributed to several contributing factors, such as insufficient antenna sensitivity, low transmission power, or limitations in the data acquisition setup. These low SNR conditions indicate that even though signal detection is possible under controlled environments, the reliability of detection might degrade significantly in the presence of environmental disturbances, higher transmission rates, or increased antenna distance.

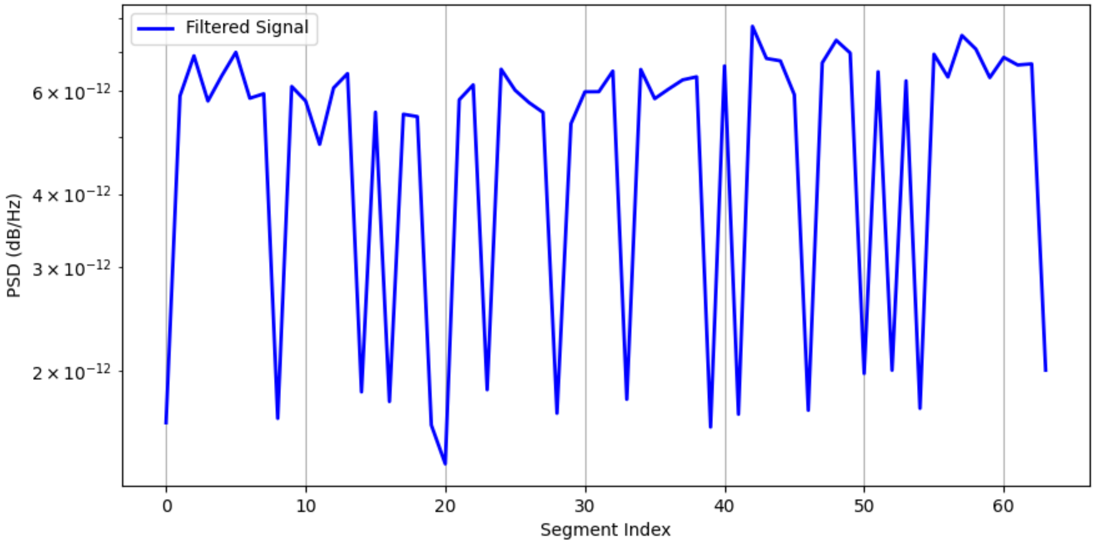

Figure 5.5: PSD with segment-windows under 11bps data rate for filtered signal

Even with the application of noise cancellation techniques (applying LMS adaptive filter), Figure: 5.5 show that the preamble could not be reliably extracted and the correct bit sequence could not be recovered. This indicates that the problem lies not only in the presence of noise but also in the inherent weakness of the emitted EM signals and the limited resolution or sensitivity of the capturing setup. These findings highlight the limitations of using noise filtering alone and suggest the need for more robust techniques such as enhanced signal shaping, improved antenna sensitivity, and other advanced techniques to enable reliable bit demodulation from far-field EM emissions, especially at higher data rates.

5.4.3 Bit Error Rate with Varying Probe Distance

In this section, analyze the BER of the covert channel as a function of antenna distance, ranging from 2cm to 14cm, under a data rate of 10 bps. The BER serves as a key metric for evaluating the reliability of data transmission over an EM side channel.

As previously discussed, distinguishing the EM emissions of an Ethernet cable from ambient EM noise becomes increasingly challenging as the probe moves away from the cable. To analyze this phenomenon, the preamble sequence 10101010 is modulated using Manchester encoding over the transmission medium.

Figure: 5.6 illustrates the amplitude variation of the observed EM emissions over time. Notably, in Figure: 5.6, the amplitude fluctuations are minimal, making it difficult to detect the signal. This reduction in signal strength at greater distances significantly impacts the effectiveness of the covert channel, demonstrating the limitations of EM-based data exfiltration under such conditions.

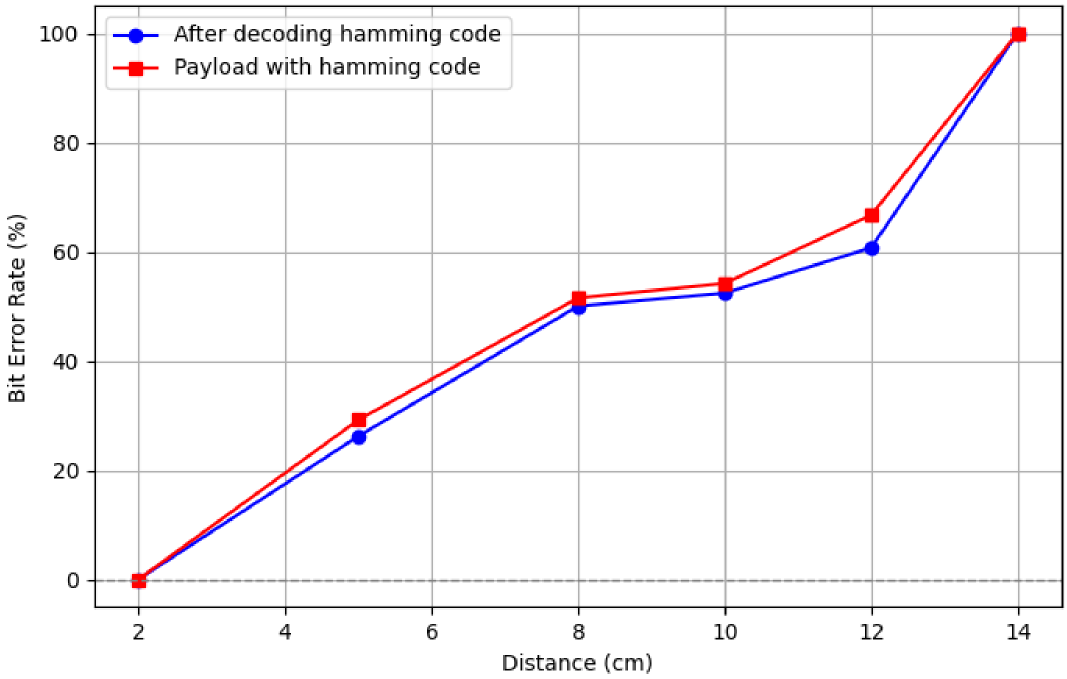

Figure 5.6: BER with distance under 10bps data rate

5.4.4 Signal-to-Noise Ratio with Varying Probe Distance

Figure: 5.7 illustrates the variation in SNR as a function of antenna distance under a low data rate of 10bps. Two signal conditions were analyzed: the original signal and the filtered signal. The plot clearly shows that as the antenna distance increases, the SNR for both the original and filtered signals decreases steadily, indicating a degradation in signal quality with increasing separation between transmitter and receiver.

At shorter antenna distances, specifically at 2cm and 4cm, the captured signals exhibit significantly higher SNR values. This indicates that the signal quality is inherently strong and the level of noise is minimal at these ranges. As a result, the application of additional noise cancellation mechanisms such as adaptive filtering becomes redundant in these scenarios. The high SNR values suggest that the signal can be reliably interpreted and processed without the need for further enhancement or filtering, thereby reducing computational overhead. This observation highlights the importance of antenna placement in ensuring optimal signal integrity, especially in systems where simplicity and efficiency are critical.

This trend continues across all distances, where the filtered signal consistently maintains a higher SNR compared to the original signal. The advantage of filtering is most evident at closer distances, helping to preserve signal integrity. However, beyond 12 cm, the SNR drops below 0 dB for both signals, suggesting increased noise dominance and reduced detectability. This analysis confirms that both antenna proximity and signal processing significantly influence signal clarity, especially under constrained data rates such as 10bps.

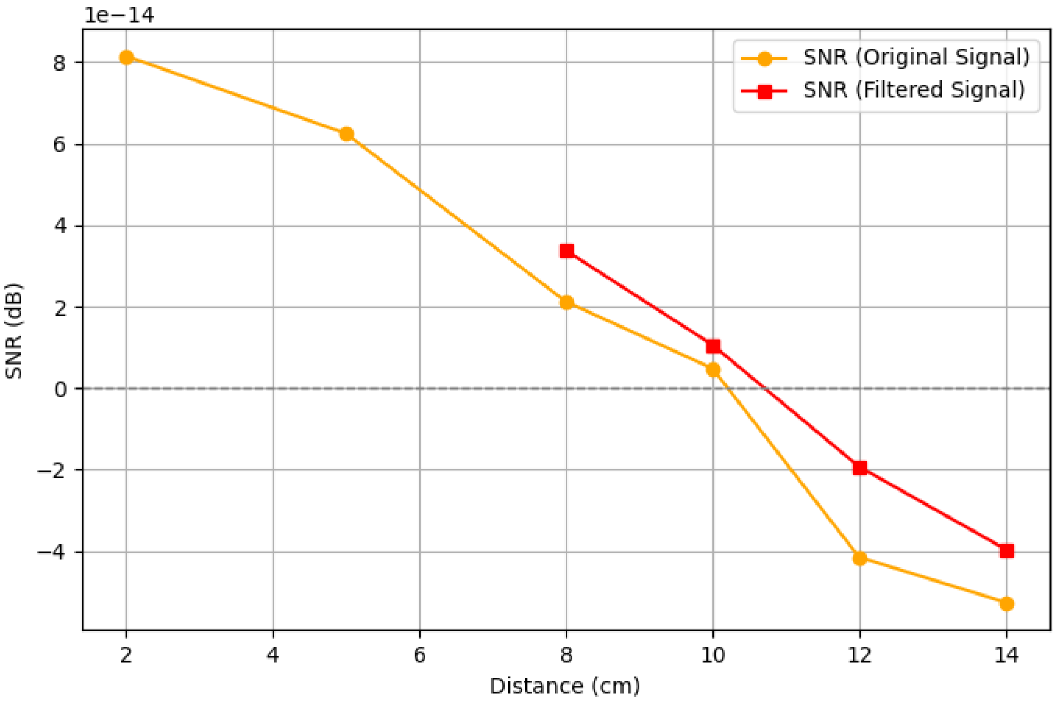

Figure 5.7: SNR with distance under 10bps data rate

5.5 Performance Analysis of LMS Adaptive Filter

5.5.1 Varying Filter Order(M) and Step-Size(μ) Parameter for Noise Cancellation and Signal Prediction

In this section, the performance of the LMS adaptive filter is evaluated by systematically varying the filter order (M) and step-size (μ) parameters. The filter order M determines the number of taps in the filter, thereby influencing the filter's ability to model complex input signals. Conversely, the step-size μ controls the convergence behavior and stability of the LMS algorithm. A series of simulations were conducted using different values of M (64, 128, and 256) and μ (0.01, 0.03, and 0.05), and the results are presented through PSD plots to visually capture the signal characteristics after noise cancellation.

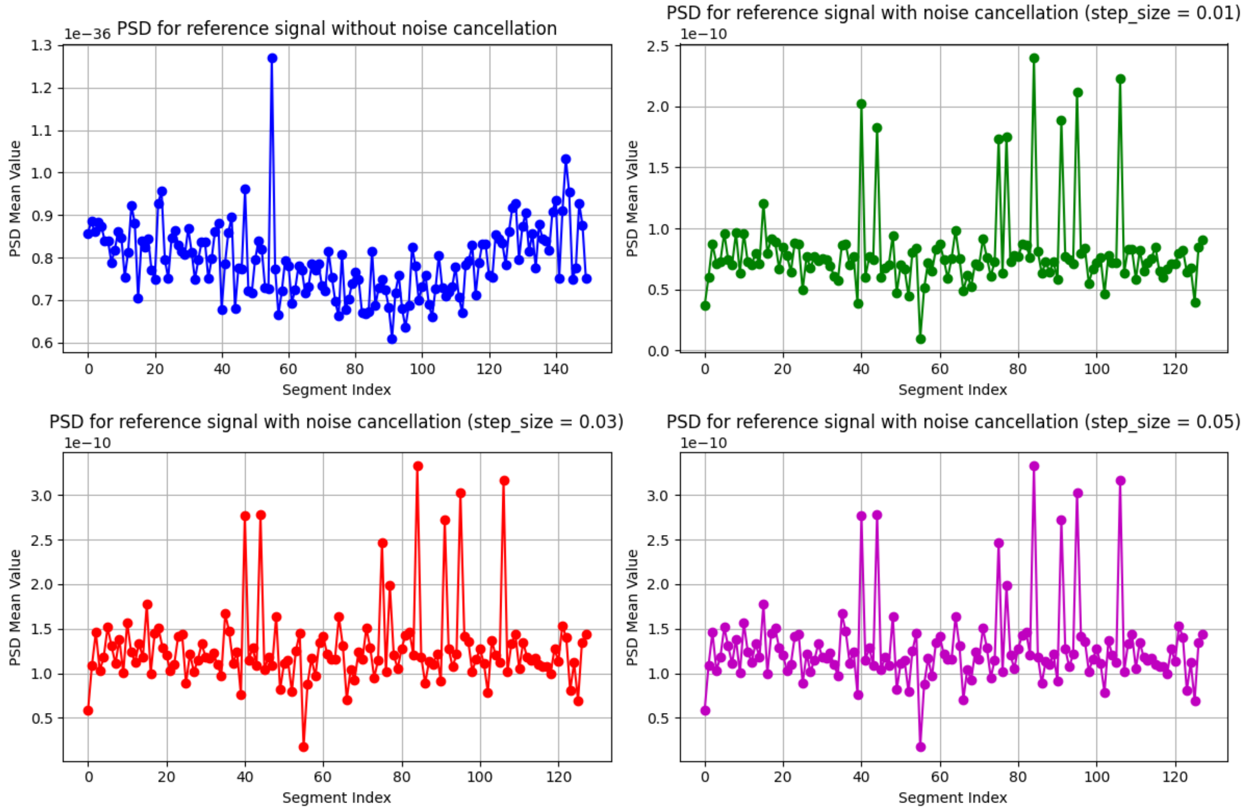

Figure 5.8: LMS Filtering with varying step size (μ)

As the filter order M increases, the LMS filter demonstrates improved capability in adapting to the signal structure and suppressing noise more effectively. With M = 256 (Figure: 5.9), the filter shows a higher resolution in distinguishing signal components from background noise. However, this comes at the cost of increased computational complexity and slower convergence.

Similarly, the step-size μ has a direct impact on the convergence speed and steady-state error. A smaller μ (e.g., 0.01) results in a slower adaptation process but provides a more accurate steady-state solution. In contrast, larger μ values (e.g., 0.05) speed up convergence but may introduce higher residual noise or even instability if the value is too large. Figure: 5.8 show that an intermediate value (μ = 0.03) strikes a balance between these two extremes, providing efficient noise suppression while maintaining algorithm stability.

Overall, the analysis underscores the importance of careful parameter tuning in LMS-based adaptive filtering. The combination of a sufficiently high filter order and an appropriately cho-

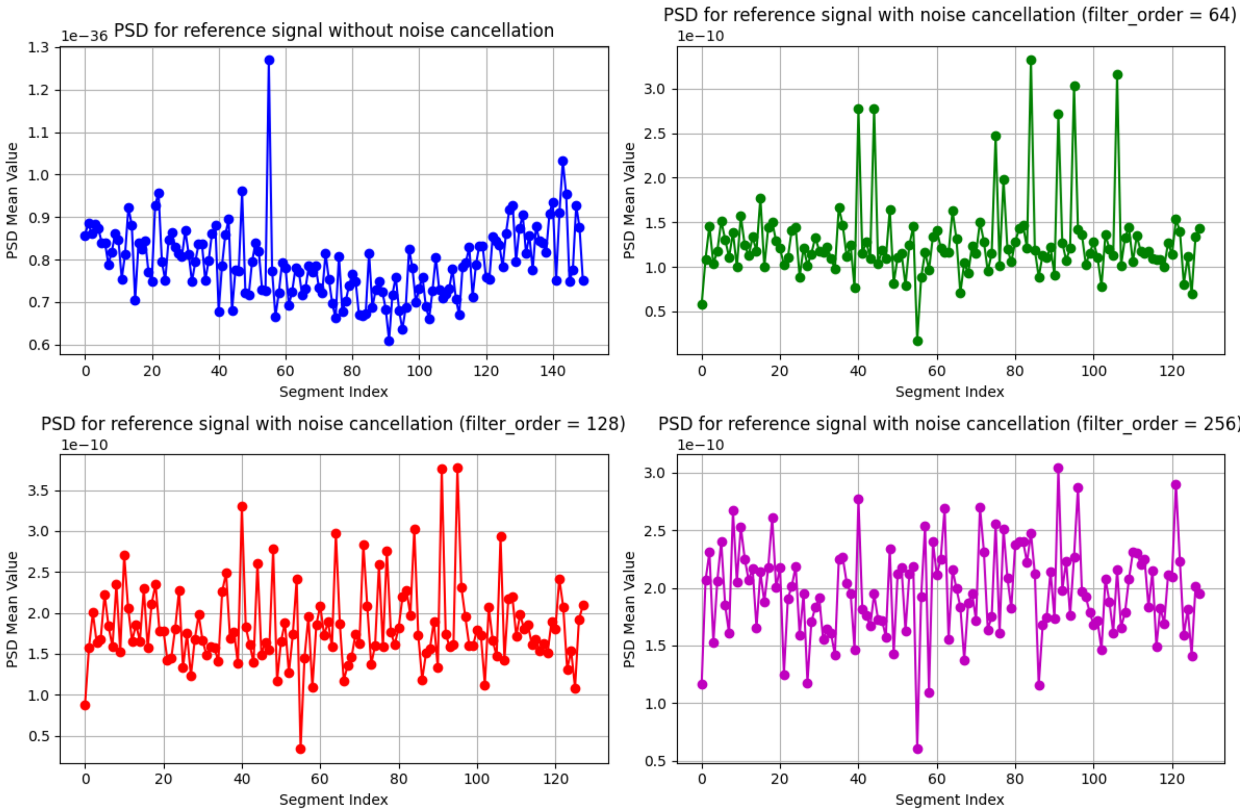

Figure 5.9: LMS Filtering with varying filter order (M)

sen step-size significantly enhances the performance of the filter in both noise cancellation and signal prediction tasks. These insights are vital for real-world applications where adaptive filtering is deployed under noisy environments.

6 Conclusion

6.1 Overview

This research focused on understanding how data can be secretly sent out (exfiltrated) using EM signals from Ethernet cables, especially Cat6 cables, and received by a far-field antenna. Looked into how fast the data can be sent, how distance affects the signal, and how well the attack can work under real conditions. Experiments were done to study the data rate and signal strength at different distances.

6.2 Conclusion of Research Question 1

What is the maximum data rate achievable for exfiltration using far-field antennas and Cat6 cables?

The experiments showed that the maximum data rate depends on the strength of the EM signal and how clearly the far-field antenna can receive and decode it. In this study, a reliable data rate of around 10bps was achieved using Manchester encoding, which ensures clear bit transitions. This data rate was successfully demodulated at distances as close as 2cm from the cable. This proves that even without direct access to the Ethernet system, data can be exfiltrated at low rates under the right conditions.

As the data rate increases, the signal becomes harder to separate from background noise, especially at longer distances (after 2cm). During testing, attempts to transmit data at 11bps and 12bps showed reduced performance and reliability. To improve the signal quality, a LMS adaptive filter was used for noise cancellation. This helped clean the signal, but it was still difficult to maintain higher data rates with consistent accuracy.

6.3 Conclusion of Research Question 2

How does distance from the Ethernet network to the far-field antenna impact the Effectiveness of data exfiltration using Cat6 cables?

As the distance between the Ethernet cable and the far-field antenna increases, the strength of the EM signal decreases significantly. This weakening of the signal makes it harder for the receiver to detect and accurately decode the transmitted data. In this study, decoding was successful up to around 2cm, but beyond that range, the signal quality dropped due to increased noise and lower signal strength. The loss of signal clarity at longer distances highlights how critical physical proximity is to the success of this type of covert channel.

In addition to distance, the low sensitivity of the antenna also limited the system's effectiveness. The far-field antenna used in this research operated within a frequency range of 6.8–11 GHz, which is relatively high and not specifically tuned to the lower-frequency emissions naturally generated by Ethernet activity over Cat6 cables. As a result, the antenna was less effective at capturing the already weak signals from the cable, especially at greater distances. Even with LMS adaptive filtering used to reduce noise, the antenna's frequency range and sensitivity played a major role in restricting the performance of the covert channel. This shows that both the antenna's frequency characteristics and placement are essential for reliable data exfiltration.

6.4 Conclusion about the Research Problem

This research confirms that data can be covertly exfiltrated through EM emissions from Cat6 Ethernet cables. These emissions, although intended, can carry information that is modulated and received by antennas placed some distance away. The study demonstrated that even without physical access to a device or network, an attacker can use software-based modulation techniques and a nearby antenna to leak small but sensitive pieces of data, such as passwords or encryption keys. This presents a serious security risk for air-gapped or otherwise isolated networks, proving that such systems are not immune to side-channel attacks.

The findings emphasize that with the right conditions, such as proper signal encoding, noise filtering, and antenna placement, data can be exfiltrated from secure environments using only electromagnetic leakage. If enhanced with more sensitive antennas and better error correction mechanisms, the covert channel could become even more reliable. These results highlight the importance of physical-layer security and bring attention to a rarely addressed attack vector in cybersecurity.

6.5 Limitations and Future Directions

Despite the successful demonstration of electromagnetic (EM) data exfiltration from Ethernet cables using far-field antennas, this study faces several critical limitations that constrain the practical applicability of the proposed covert channel.

The most prominent limitation is the extremely short effective communication range. Reliable decoding of transmitted data was only feasible within approximately 2 centimeters from the Ethernet cable. Beyond this point, signal strength deteriorated rapidly due to spatial attenuation and ambient electromagnetic noise. As a result, the Signal-to-Noise Ratio (SNR) declined to levels insufficient for accurate bit extraction, even with the application of filtering techniques such as the Least Mean Squares (LMS) adaptive filter.

Another constraint lies in the low data transmission rate, which was limited to around 10 bits per second. While sufficient for exfiltrating small data fragments such as encryption keys or short authentication tokens, this rate is impractical for transmitting large-scale sensitive information. Additionally, the simplistic encoding method and basic modulation patterns used in this work further limit scalability and transmission resilience.

The hardware configuration also introduced limitations. The far-field antenna employed in this study operated within a relatively high-frequency band (6.8–11 GHz), which was not optimally tuned to the lower-frequency emissions typically generated by Ethernet cables. This mismatch reduced the efficiency of EM signal capture, especially over extended distances or in noisy environments.

Future Directions

To overcome these limitations, several avenues for future research are proposed:

Enhanced Antenna Design: Future studies could explore the use of high-gain, frequency-tuned antennas specifically designed to capture low-frequency EM emissions from Ethernet hardware. Directional antennas or array configurations may also help improve range and precision.

Advanced Signal Processing Techniques: Incorporating more sophisticated error correction mechanisms (e.g., Reed-Solomon codes, LDPC) and adaptive thresholding could increase decoding accuracy in noisy conditions. These techniques would strengthen the channel's resilience to environmental variability.

Improved Modulation Schemes: Alternative modulation methods such as frequency shift keying (FSK) or spread spectrum techniques might improve bit distinguishability and increase the feasible data rate while maintaining signal integrity.

Environmental Adaptation: Integrating dynamic filtering based on real-time noise profiling could allow the system to adapt its demodulation strategy depending on ambient interference levels.

Security Countermeasures: On the defensive side, organizations should consider implementing physical-layer protections such as cable shielding, EMI suppression materials, and signal obfuscation. Additionally, intrusion detection systems (IDS) or machine-learning-based anomaly detection mechanisms could be developed to monitor EM leakage patterns and raise alerts when suspicious activity is detected.

This research highlights the potential of far-field EM emissions as a viable covert channel for data exfiltration, but also reveals significant technical barriers. Addressing these challenges through interdisciplinary innovations in hardware, signal processing, and cybersecurity will be essential to fully understand and mitigate EM side-channel threats in modern computing environments.

For future work, several enhancements could be made. Using high-sensitivity antennas tuned to the emission characteristics of Ethernet signals could significantly improve the effective range and signal quality. Also, implementing advanced error correction techniques could make the covert channel more robust against noise and interference. Future research can also investigate more efficient modulation schemes to boost data rates while maintaining reliability.

Finally, defensive measures should be explored, such as physical shielding, anomaly detection systems, or active jamming of unintended emissions. These would be crucial for organizations that handle sensitive data and want to mitigate the risk posed by EM side-channel attacks.

References

GeekForGeeks (2024), ‘Least mean squares filter in signal processing’.
URL: *https://www.geeksforgeeks.org/least-mean-squares-filter-in-signal-processing/*

Guri, M. (2021*a*), Lantenna: Exfiltrating data from air-gapped networks via ethernet cables emission, *in* ‘2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC)’, pp. 745–754.

Guri, M. (2021*b*), ‘Magneto: Covert channel between air-gapped systems and nearby smartphones via cpu-generated magnetic fields’, *Future Generation Computer Systems* **115**, 115–125.
URL: *https://www.sciencedirect.com/science/article/pii/S0167739X2030916X*

Guri, M. (2023), ‘Air-gap electromagnetic covert channel’, *IEEE Transactions on Dependable and Secure Computing* pp. 1–18.

Guri, M., Monitz, M. & Elovici, Y. (2016), Usbee: Air-gap covert-channel via electromagnetic emission from usb, *in* ‘2016 14th Annual Conference on Privacy, Security and Trust (PST)’, pp. 264–268.

Guri, M., Solewicz, Y. & Elovici, Y. (2018), Mosquito: Covert ultrasonic transmissions between two air-gapped computers using speaker-to-speaker communication, *in* ‘2018 IEEE Conference on Dependable and Secure Computing (DSC)’, pp. 1–8.

Guri, M., Zadov, B., Bykhovsky, D. & Elovici, Y. (2020), ‘Powerhammer: Exfiltrating data from air-gapped computers through power lines’, *IEEE Transactions on Information Forensics and Security* **15**, 1879–1890.

Loughry, J. & Umphress, D. A. (2002), ‘Information leakage from optical emanations’, *ACM Trans. Inf. Syst. Secur.* **5**(3), 262–289.
URL: *https://doi.org/10.1145/545186.545189*

NextPBC (2024), ‘What is signal to noise ratio and how to calculate it?’.
URL: *https://www.nextpcb.com/blog/what-is-signal-to-noise-ratio-and-how-to-calculate-it*

Park, J., Yoo, J., Yu, J., Lee, J. & Song, J. (2023), 'A survey on air-gap attacks: Fundamentals, transport means, attack scenarios and challenges', *Sensors* **23**(6).
URL: *https://www.mdpi.com/1424-8220/23/6/3215*

Sachintha, S., Le-Khac, N.-A., Scanlon, M. & Sayakkara, A. P. (2023), 'Data exfiltration through electromagnetic covert channel of wired industrial control systems', *Applied Sciences* **13**(5).
URL: *https://www.mdpi.com/2076-3417/13/5/2928*

Sayakkara, A., Le-Khac, N.-A. & Scanlon, M. (2019), 'A survey of electromagnetic side-channel attacks and discussion on their case-progressing potential for digital forensics', *Digital Investigation* **29**, 43–54.
URL: *https://www.sciencedirect.com/science/article/pii/S1742287618303840*

Schulz, M., Klapper, P., Hollick, M., Tews, E. & Katzenbeisser, S. (2016), Trust the wire, they always told me! on practical non-destructive wire-tap attacks against ethernet, *in* 'Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks', WiSec '16, Association for Computing Machinery, New York, NY, USA, p. 43–48.
URL: *https://doi.org/10.1145/2939918.2940650*

Smith (2022), 'Manchester encoding'.
URL: *https://sierrahardwaredesign.com/basic-networking/glossary-item-manchester-encoding/*