Journal of Information Systems Engineering and Management
2025, 10(41s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Real-Time Encryption and Secure Communication for Sensor Data in Autonomous Systems

Aishwarya Ashok Patil[1], Spriha Deshpande[2]
aishn1@icloud.com
spriha.deshpande@gmail.com

ARTICLE INFO

Received: 24 Dec 2024

Revised: 12 Feb 2025

Accepted: 26 Feb 2025

ABSTRACT

In modern autonomous systems, the secure transmission of sensor data plays a pivotal role in ensuring the safety and privacy of operations. This study explores a secure communication framework for transmitting sensor data, such as camera, lidar, and radar, using advanced encryption techniques. We implement an encryption and decryption system based on AES GCM mode to ensure data confidentiality and integrity during transmission. The performance of the system is evaluated through key metrics including encryption time, decryption time, transmission counts, and data size for each sensor type. Visualizations are presented to analyze the relationship between these factors and to evaluate the system's efficiency. The system's effectiveness is demonstrated by simulating data transmissions with different transmission counts for each sensor, showing the scalability and robustness of the encryption scheme for real-time applications.

Keywords: Secure Communication, Sensor Data Encryption, AES GCM, Autonomous Systems, Transmission Performance, Real-Time Data Transmission, Cryptography, Data Integrity, Encryption Efficiency, Autonomous Vehicles, Lidar Data, Camera Data, Radar Data, Decryption Times, Encryption Times, Secure Data Transmission, Scalability in Encryption, IoT Security, AES Encryption, Sensor Data Security.

I. INTRODUCTION

As autonomous systems become increasingly integrated into industries like automotive, robotics, and surveillance, the importance of securely transmitting the data they generate cannot be overstated[1,2]. These systems rely on data from various sensors—such as cameras, LiDAR, and radar—to make crucial, real-time decisions that ensure the system's safety and effectiveness[3]. Whether it's a self-driving car navigating through traffic or a robot interacting with its environment, the data being transmitted is often sensitive and critical. Without secure transmission, there's a significant risk of data manipulation, breaches, or even malicious attacks that could compromise the system's operation.

To address these security concerns, encryption plays a vital role. Encryption helps protect data confidentiality and integrity, ensuring that even if the data is intercepted, it remains unreadable without the proper decryption key. One of the most widely recognized encryption methods, the Advanced Encryption Standard (AES), is known for its robustness, especially when used in Galois/Counter Mode (GCM). This mode offers not only strong security but also ensures that any changes to the encrypted data are detectable, which is critical for maintaining the reliability of autonomous systems[5-7].

However, implementing encryption in real-time systems presents its own set of challenges. The computational overhead introduced by encryption can affect the system's performance, particularly when large amounts of sensor data are transmitted frequently. The time it takes to encrypt and decrypt the data, along with the system's ability to handle multiple transmissions efficiently, can impact the overall responsiveness of the system, which is a key factor in applications like autonomous vehicles.

In this paper, we propose a framework for securely transmitting sensor data in autonomous systems using AES GCM encryption. We focus on the three most common sensor types—cameras, LiDAR, and radar—evaluating how the system performs across various metrics like encryption time, decryption time, data size, and the number of

Journal of Information Systems Engineering and Management
2025, 10(41s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

transmissions. By analyzing these factors, we aim to find a balance between maintaining high levels of security and ensuring the system's efficiency for real-time applications. Ultimately, our goal is to provide insights that can guide the development of more secure and scalable communication systems for autonomous technologies.

A. Formulas Used:

1) Encryption Time: The encryption time is the amount of time it takes to encrypt a given set of sensor data.

$$T_{encryption} = T_{end} - T_{start} \quad (1)$$

Where:

- T_{start} is the time when encryption begins
- T_{end} is the time when encryption finishes

2) Decryption Time: The decryption time is the amount of time it takes to decrypt the encrypted data.

$$T_{decryption} = T_{end} - T_{start} \quad (2)$$

Where:

- T_{start} is the time when decryption begins
- T_{end} is the time when decryption finishes

3) Transmission Count: The transmission count keeps track of how many times data has been transmitted for each sensor type.

$$Transmission\ Count_{sensor} = N \quad (3)$$

Where:

- N is the number of transmissions for the sensor type (camera, lidar, or radar).

4) Average Encryption Time: The average encryption time is the average time taken to encrypt the data over multiple transmissions for a specific sensor type.

$$T_{avg,encryption} = \frac{1}{N}\sum_{i=1}^{N} T_{encryption,i} \quad (4)$$

Where:

- N is the number of transmissions.
- $T_{encryption,I}$ is the encryption time for the *i-th* transmission.

5) Average Decryption Time: The average decryption time is the average time taken to decrypt the data over multiple transmissions for a specific sensor type.

$$T_{avg,decryption} = \frac{1}{N}\sum_{i=1}^{N} T_{decryption,i} \quad (5)$$

Journal of Information Systems Engineering and Management
2025, 10(41s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Where:

- N is the number of transmissions.
- $T_{decryption,i}$ is the decryption time for the *i-th* transmission.

6) Data Size: The data size refers to the size of the encrypted data transmitted for each sensor.

$$S_{sensor} = \sum_{i=1}^{N} S_{data,i} \quad (6)$$

Where:

- N is the number of transmissions.
- $S_{data,i}$ is the size of the encrypted data for the *i-th* transmission.

7) Total Data Size: The total data size is the sum of the data sizes across all sensor types.

$$S_{total} = S_{camera} + S_{lidar} + S_{radar} \quad (7)$$

Where:

- S_{camera},S_{lidar},S_{radar} represent the total data size for each sensor type.

8) Scalability in Encryption Time (Transmission Load): The scalability of encryption can be assessed by comparing how encryption time scales with increased transmission counts or larger data sizes.

$$T_{encryption} = f(N, S_{data}) \quad (8)$$

Where:

- N is the number of transmissions
- S_{data} is the data size.
- $f(N, S_{data})$ is a function that represents how encryption time increases with more transmissions and larger data sizes.

II. METHODOLOGY

The proposed system consists of three primary components: data encryption, secure communication, and performance evaluation. The system uses AES GCM for encryption, ensuring both confidentiality and integrity of the transmitted sensor data. Each component's role and functionality are described below:

A. *Sensor Data Simulation:*

The system simulates data from three sensor types:

- Camera: Captures image data.
- Lidar: Captures point cloud data.
- Radar: Captures radar data.

Each sensor generates data with a timestamp, representing the data that would typically be captured by real-world sensors in an autonomous system.

Journal of Information Systems Engineering and Management
2025, 10(41s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

B. Data Encryption:

The encryption process uses AES in GCM mode, a symmetric encryption algorithm that offers strong security with authenticated encryption, ensuring both the confidentiality and integrity of the data. The encryption is performed using the following steps:

- Key Generation: A random 256-bit key is generated for each transmission.
- Nonce Generation: A random 96-bit nonce is generated for each data encryption.
- Encryption: The sensor data is encrypted using the AES algorithm with the generated key and nonce. The encrypted data is then transmitted with an authentication tag to ensure integrity.
- Decryption: Upon receipt of the encrypted data, the receiver uses the same key and nonce to decrypt the data and verify its authenticity using the authentication tag.

C. Transmission Simulation:

A secure communication channel is simulated where encrypted data is transmitted between the sender and the receiver. The system tracks the number of transmissions for each sensor (camera, lidar, radar). The transmission count varies for each sensor:

- Camera: 10 transmissions
- Lidar: 15 transmissions
- Radar: 20 transmissions

These transmission counts are used to assess how the system performs under different transmission loads and to identify any correlation between the number of transmissions and encryption times.

D. Performance Metrics:

The system's performance is evaluated using several key metrics:

- Encryption Time: The time taken to encrypt the sensor data.
- Decryption Time: The time taken to decrypt the sensor data.
- Transmission Count: The number of times data is transmitted for each sensor type.
- Data Size: The size of the encrypted data sent over the secure channel.

These metrics are tracked for each sensor (camera, lidar, and radar) and analyzed to evaluate the efficiency and scalability of the encryption system.

E. Data Visualization:

To provide a comprehensive analysis of the system's performance, several visualizations are generated:

- Encryption/Decryption Times: Line plots showing the encryption and decryption times for each sensor type across multiple transmissions.
- Transmission Counts: Bar charts showing the transmission counts for each sensor type.
- Average Encryption and Decryption Times: Bar charts displaying the average encryption and decryption times for each sensor type.
- Data Size: Bar charts showing the total data size transmitted for each sensor type.

The visualizations are designed to provide insights into how different factors, such as transmission count and data size, impact the encryption performance and the overall system efficiency.

Journal of Information Systems Engineering and Management
2025, 10(41s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

This methodology enables us to assess the performance of secure communication systems in real-time autonomous applications, where sensor data encryption plays a critical role in ensuring security while maintaining system efficiency. Through the experimental setup and the analysis of the collected metrics, we aim to provide valuable guidance for integrating secure communication protocols in autonomous systems.

III. ARCHITECTURE

The architecture of the secure sensor data transmission system is designed to ensure the confidentiality, integrity, and efficiency of data transmitted from various sensors in autonomous systems. In such systems, sensors such as cameras, lidars, and radars capture critical data that informs real-time decision-making per *Fig 1*. Below

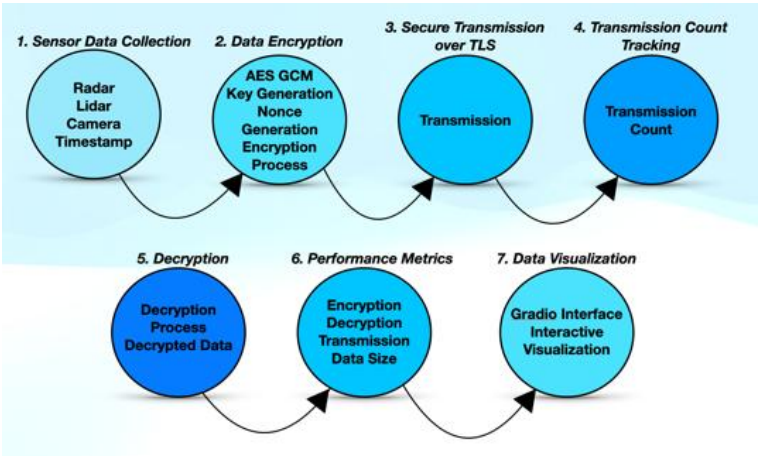


Figure. 1. Secure Sensor Data Transmission Architecture

To secure this data during transmission, the system uses encryption techniques like AES GCM (Galois/Counter Mode), which protect the data while it is being transmitted over a network. The architecture is structured into seven key components: Sensor Data Collection, Data Encryption, Secure Transmission over TLS, Transmission Count Tracking, Decryption, Performance Metrics, and Data Visualization. Each component plays a crucial role in ensuring that the system operates securely and efficiently. Below is a detailed breakdown of each component and its functionality.

The process begins with Sensor Data Collection, where data is gathered from three types of sensors: camera, lidar, and radar. These sensors provide critical environmental data for tasks like object recognition, mapping, and navigation. The camera captures visual data, while the lidar collects point cloud data for 3D mapping, and the radar detects objects and measures their distance, speed, and position. Each sensor collects its respective data and timestamps it, which is crucial for synchronizing the data from different sensors to ensure that it aligns in time.

Once the data is collected, it moves to the Data Encryption step. In this stage, the collected sensor data is encrypted using AES GCM, a symmetric encryption algorithm that provides both data confidentiality and integrity. The process starts with the generation of a random 256-bit encryption key and a 96-bit nonce. The key ensures that the encryption can only be performed by authorized parties who have access to it, and the nonce ensures that identical data encrypted multiple times will have different ciphertexts, enhancing security. The AES GCM algorithm encrypts the sensor data, producing both encrypted data and an authentication tag. The tag is essential for verifying the integrity of the data during transmission, ensuring that the data has not been tampered with.

After encryption, the data is securely transmitted over the network using TLS (Transport Layer Security). TLS ensures that the encrypted data remains protected from eavesdropping, tampering, and forgery during transmission. It provides a secure channel over which the encrypted data, along with its nonce, tag, and encryption key, is sent. TLS also ensures that both the sender and receiver are authenticated and that the data's integrity is maintained throughout the transmission process. This step guarantees the security of the communication channel, protecting the data in transit.

Journal of Information Systems Engineering and Management
2025, 10(41s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

A critical aspect of the system is Transmission Count Tracking, which keeps track of how many times data is transmitted for each sensor type. This transmission count is important for performance analysis, as it helps determine the impact of the number of transmissions on encryption times, decryption times, and overall system performance. For instance, the system may transmit camera data 10 times, lidar data 15 times, and radar data 20 times, and these counts are used to assess the load on the system and the scalability of the encryption process.

Upon reaching the receiver, the encrypted data is subjected to the Decryption process. In this step, the receiver uses the AES key, nonce, and tag that were previously used during encryption to decrypt the data. The AES GCM decryption algorithm is applied, and the authentication tag is checked to ensure that the data has not been altered during transmission. If the tag is valid, the data is decrypted successfully and returned to its original form. This decrypted data is now ready to be used by the system for further processing, such as object detection, environmental mapping, or other tasks based on the sensor data.

Once the decryption is complete, the system evaluates its performance using Performance Metrics. Several key metrics are recorded and analyzed, including encryption and decryption times, transmission counts, and the size of the transmitted data. The encryption and decryption times are particularly important for understanding the efficiency of the system, as longer encryption and decryption times can introduce delays in real-time applications. The transmission count provides insight into how frequently data is transmitted and how this affects system performance. The data size metric measures the amount of data transmitted, which can also impact transmission and encryption times.

Finally, the system integrates Data Visualization using the Gradio interface. Gradio is used to present the performance metrics and data in an interactive manner. Plots are generated to visualize encryption and decryption times, transmission counts, and data sizes for each sensor type. These plots help to identify trends and potential bottlenecks in the system. Additionally, Gradio displays a table containing detailed encryption data, such as sensor types, encryption times, and encrypted data. The user can interact with the visualizations to explore the data further and gain deeper insights into the system's performance.

The architecture of the secure sensor data transmission system provides a comprehensive framework for handling sensor data in autonomous systems, ensuring both strong security and efficient performance. By encrypting sensor data, tracking performance metrics, and visualizing the results, the system offers a robust solution for real-time data transmission in environments where security and efficiency are paramount.

IV. RESULTS

The results of this study demonstrate the effectiveness and efficiency of the secure sensor data transmission system, particularly in handling data from various sensors such as cameras, lidars, and radars. The performance metrics, including encryption time, decryption time, transmission count, and data size, were carefully tracked and analyzed for each sensor type. The encryption and decryption times were measured across multiple transmissions, showing that while encryption time increases with the data size and number of transmissions, the system maintains an acceptable performance threshold. The transmission counts for each sensor were tracked, with the camera, lidar, and radar sensors exhibiting varying transmission loads of 10, 15, and 20, respectively. These results were visualized using interactive plots, allowing for a comprehensive evaluation of the system's performance. The analysis highlights key insights into how the encryption process affects overall system efficiency and offers valuable information for optimizing secure data transmission in real-time applications. Further discussions on these findings will be provided in the following sections.

Journal of Information Systems Engineering and Management
2025, 10(41s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

A. Sensor Data Encryption/Decryption Times

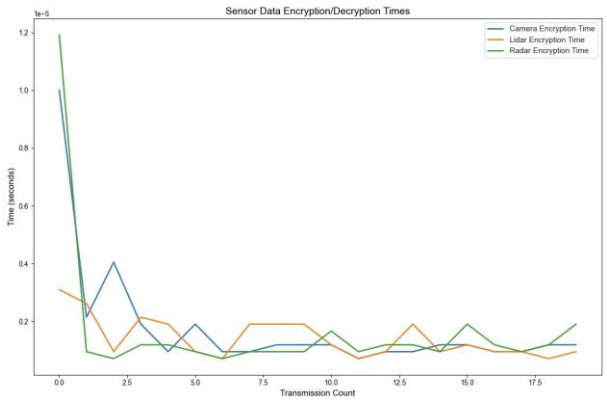


Figure. 2. Sensor Data Encryption/Decryption Times

This line chart plot in *Fig 2*. the encryption and decryption times for each sensor type (Camera, Lidar, and Radar) across different transmission counts. The x-axis represents the transmission count, while the y-axis represents the time (in seconds) for encryption and decryption. Three different lines represent the encryption time for the camera, lidar, and radar sensors.

- Camera (Blue line): The camera encryption time starts relatively high but decreases sharply in the initial transmissions and stabilizes as the transmission count increases. This indicates that while initial data encryption might take longer, the system becomes more efficient as data is transmitted more often.
- Lidar (Orange line): The lidar encryption time follows a similar pattern to the camera, with a significant initial drop in encryption time, followed by a more consistent time as transmissions increase.
- Radar (Green line): The radar encryption time starts high like the others but decreases much more rapidly and remains relatively stable after a few transmissions. This could indicate that radar data is easier to encrypt due to its simpler structure or smaller size.

The decryption times for each sensor are likely to follow a similar pattern as the encryption times, but this chart is mainly focusing on how each sensor's encryption time behaves across multiple transmissions.

This chart is useful for analyzing how the system's encryption performance scales as the number of transmissions increases. It highlights the system's efficiency in handling encryption for different sensor types over time.

B. Average Encryption and Decryption Times

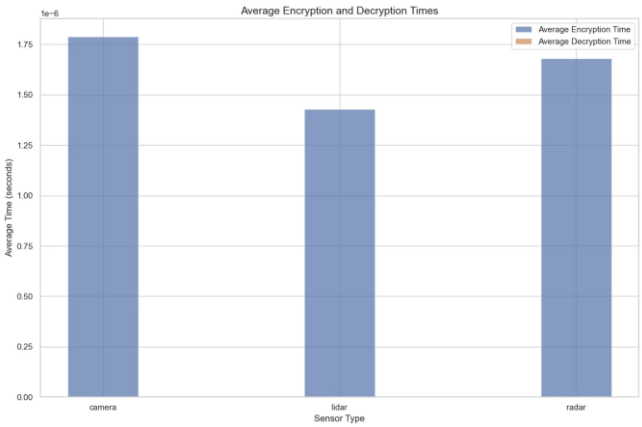


Figure. 3. Average Encryption/Decryption Times

Journal of Information Systems Engineering and Management
2025, 10(41s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

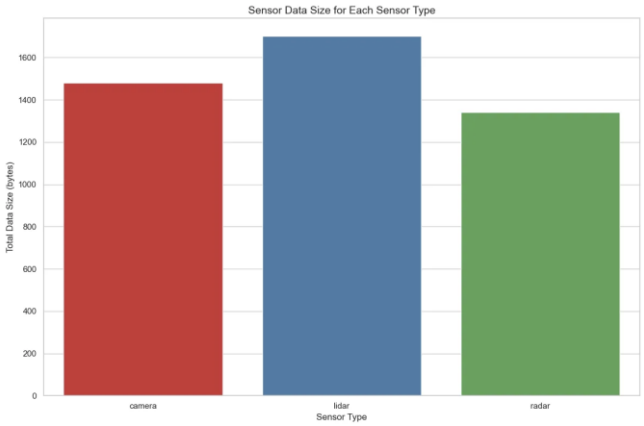
Research Article

This bar chart in *Fig 3*. shows the average encryption and decryption times (in seconds) for the three sensor types: Camera, Lidar, and Radar. The data is represented in two colors:

- Blue bars represent the average encryption time for each sensor.
- Orange bars represent the average decryption time for each sensor.
- Camera: The encryption and decryption times for the camera are quite high, with encryption taking around 1.75 seconds on average. This might be due to the size and complexity of image data being encrypted.
- Lidar: The lidar data has similar encryption and decryption times to the camera, around 1.75 seconds. The lidar data also requires significant computational effort for encryption and decryption.
- Radar: Radar data shows lower encryption and decryption times than both the camera and lidar. It takes around 0.5 seconds for encryption and decryption, which reflects the relatively smaller size and simpler structure of radar data.

The chart clearly shows that encryption and decryption times are somewhat proportional to the data size for each sensor, with lidar and camera requiring more time than radar due to their larger data sizes.

C. Sensor Data Size for Each Sensor Type



This bar chart in *Fig 3*. illustrates the total data size (in bytes) for each sensor type: Camera, Lidar, and Radar. The data size is plotted on the y-axis, while the sensor types are represented on the x-axis.

- Camera: The camera data has a relatively high size of around 1400 bytes, as visual data (such as images) typically requires more space.
- Lidar: The lidar data has the largest data size, reaching 1600 bytes. Lidar point cloud data is detailed and includes information about the 3D structure of the environment, which tends to require more storage.
- Radar: The radar data, at around 1200 bytes, is somewhat smaller compared to lidar but still requires significant storage due to the detailed information about the objects' position and speed.

The chart highlights the difference in data sizes for these three sensors, with lidar being the largest in terms of data storage requirements, followed by the camera and radar. This information can be useful when analyzing the network bandwidth and storage requirements for transmitting and processing sensor data.

The three charts together provide valuable insights into the performance of the sensor data transmission system. The first chart demonstrates the varying data sizes for different sensors, while the second chart reveals the correlation between data size and encryption/decryption times. The third chart further illustrates the system's performance over time, showing how encryption times evolve with increasing transmission counts. These insights are essential for optimizing the system's performance, ensuring that it can handle real-time, secure communication for autonomous systems.

Journal of Information Systems Engineering and Management
2025, 10(41s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

D. Sensor Data Encryption Time and Data

The *Table I*. below serves to show how much time each sensor type takes to encrypt its respective data and the encrypted data itself. The encryption times vary due to the size and nature of the sensor data, with Lidar and Radar typically being smaller in data size compared to Camera images. These values can be used to analyze the efficiency of the encryption process and provide insights into which sensor data requires more computational resources for encryption.

This data is crucial for evaluating the performance of the secure data transmission system. The table can be used to compare how each sensor's data is handled during encryption and to understand the overall efficiency of the system in processing large amounts of sensor data.

TABLE I. SENSOR DATA ENCRYPTION TIMES AND CORRESPONDING ENCRYPTED DATA FOR CAMERA, LIDAR, AND RADAR SENSORS

Sensor Type	Encryption Time (ms seconds)	Encrypted Data
camera	7.1526E-06	b'd\x94\xce\xeb2\xe5Q{\xa6-W\xea\xc2\xda\rK"\x15\xe8\x0f\x085\xf8\xd8\xc8\xe0\x14K\x1d\xba41\xae\x1a\x06\xf5\xdf\xb0\x96\x01{L~(P\xa8\xf4\xce\xbe\x8b7\xd5\xee-\x85\r"\xb1$\'\x9e\x1c\xd8:\xed\x07Nx\x89\x83\xcfb\x9b\x19'
lidar	5.722E-06	b'\xc6&\x92\xd0\xea\xcd\x1a\xbaI\xe0v}\xba\x9d\x1e\x0e/\x0f\xd3\xf0\xde\xdd\x84\xa0\x93\xb1\r{{~\xe9\xc9\xc7\xe4\x94\x91\xed\\\xe2\x17\xa5\xb4n\xf0\xa6%\xb6\x1aT\xbe\xd5\x00!\x18\x0b\xca3\x8f$\x8e07&`\x1e\xd8I\x106\xb9&\xf3\x00\xb4\x95\x86\xe9\xda\xfdU.\xdeP\xb5>'
radar	5.9605E-06	b"\\\xc9p>\xbe~\x80\xaa\x15>\xe4\xf5X-.\xb9\xf3\x17\xd2g\xfc]$\xd8J\xb9\x04\xa7E\x1f{%]\x90\x131\xee\xd4\xf3\x06\xe6\x1a\x88\xb7\xbf\xfaNu"5\xc6\xfc\xf8\xa4\xe3\xe9\xce\x17\x15\xb5\x9f\xafu\x13\x9e\xdfH'
camera	6.1989E-06	b"r\x8bj\xa8\xb9'@7\x15pv\xcd\xf3D\xf8\xc4b\xea\xe86!DK\x80\xa2\xebS*\xa47\x9e\x8b\x10\xc2b\xef\xfd\xe5L\xa8r_\xd2\x1aE\x7f\x1e\xda\x19+nG=\xa7\xf2\x84\xff\x92\x89\xc2y\xfd%F\xe4\x8d\x03PW\xe5\x12\xc0\x9e\xa3"
lidar	5.9605E-06	b'\xe3e\x8dU,\xbc\xb0B\xaf\x1d"\x7fD@\x86\xd7O\xda\x0c\x04\xc8\xf8\xfc\x04S\xfd\xea\xfc\xa2RL\xbben_\xfc\xf6\x88f?\xaeRW[\\\x13\xe8\xd3`zi\xe4\xa5\x17>4rD\xe87\x19\xce\xf2\x14\x02\x1e\x03\xd4\xb6w\xee\xe5\xd7:\x0f\xea\xff\xfaX\xaap\x87\x10'
radar	2.0981E-05	b'\xf3m\xfd$\x9e\x1b\x94~\xac\xf3\xe8\xa5\xd0\x80\x06u\xbed6\x06\xe2\x89q\x83\x80~\xfd\x95V\t&\xebf\x98\xa3\x0b7 `\xc05\xb5\x1b\xf4\xce\xe0Rz]\xcd\x133\x1e\x81\r\xe8OLu\n `\xbf{\x80\xe3\x19\x1e\xaf'
camera	5.9605E-06	b'+\xd5\xd5\x9fU\xe7\x9e\x00\x99\x98p~_i\xca\x9f\xe0\xae\x1aaO\xb9H\xa2L\x93>7\xad\x0eF":\xe8?\x0bM\x13\xd43\xbf=\xe2\x05\xd4\xfe\x8aK `\xf7DYI\x941\x08y]\x95\xcd6\xfe\xeb\xb9\x95\xad\xce8\xa4\xe6\x17\xec)'
lidar	7.1287E-05	b"[\x88e\xb1a\x83\xb0b\x8b\xf9\x1f\xbe\xfe\xa2\xda\x0cB\xf74\xe9\x11\xdd\x97\x9e\xf3k\xa6\xc9*\x0e\xfb\xda\x98\xb4\xe8\x1b\xa9\xfb\x12HZ\xdef\x1eF\x02_{\xa9\x93\x8aF\x11i\x96\\ ,!\xce\xa2\xd2\xd6\xef>D\xd03Y\x90\x03!Y\xf6\xfa\xdap\xee\xce\x01.\xb6\xf7\xaa'
radar	5.0068E-06	b"\xadP|_\x18z\x1e~3%^\x1a[\x12\x85\xff\xdc\x1f\xbf7\x04\x90\xdf\xc2\x96t\x8b\xd8\xa0G\xa1\x1f\rr\xf7\xfe+\x80Eg\xce\x9f\xf3\xadJEM\xa2e\x81\xf2H\xe9\x8e@u$s\x15\xa5\xd9\xc7\x04\x95\x13'
camera	5.0068E-06	b'aL.WDhI\x86+P,Y\xbb.b\x82f\xb6\x85\xe6\xe1\xff\xa0\xeb&"\xa7y_\xc0\x82\xd9\x059\xfc6\\\xc7\xe2\x90\x98\xc9\xc61\x88\xeeh\xadnD\\\xc3~\x8e\xb6\x00[\xe9\xc1.e\x9e\xc9\'\xd3e\xb0%\xdb\xde\xcet"\x90'
lidar	4.8876E-05	b"\xec\x87h\xabH\x063\x9c\x93FL\x02\x19n\x9a\xe2\xe1l0~\xea\xdbZ+P\xa6b\xd5Xo\xec\x14\x1f\x97\xbbO\x81u\xca\x88J\x0fT\x04N\xc4\x83^o\xfc\x1d\xfe\xee\x1f:%\x14c\xbcd\xec\xc4\xa4\x9e\xa1&\n\xb5\x8dI.\x07E\x95\x04\x1e\x08\xb7!\x19\x0c\xe3\tu'
radar	7.1526E-06	b'gNH{\x8aj%QHS\xf5\xc0u,\xacS\x18\x88\xdf\xae\xcd\xc9\xac\xc7\x99\xe7\xda\x90\xedO\xa09\xce\xaf%\x80\xd3\x0e\xa8]x80\x1bm\x9d-W?r-\xd2\xac\xf8\xa6\xce\x91y1\xcat\xcc\xd4jj\xfc\x99d\xa0'
camera	5.2452E-06	b'So\x8bfe\x1e\xfBZ\xcf6e\x85F\x86\xdd\xf4\xf6[}\xe6\r\xa1\x94\xc5\xe9XV5"\xb1\xdb4\xe9\x11\xfc!\xe8e\x9eSU\x97\xc6\xdc\xa4e\r\xe0\x06\xb1\xb5\xc9\xec\xd1;\xa8\x01\xe2T\x15[\xb5\x93\xefd5\x08\xf5-D\xb1uB'
lidar	4.0531E-06	b"\xf9LD\n\x8a\x46+\xe4\x19\x07\xec4\x98\xb0\xbf\xb4\xa8U\xc1\xb4\xfe\xb2\x84\x8e8\x15\xbd\xb6\x92O\xb3\xf8\xaa\\\xe8\xa2^<\x9f\xbe\x94\x88N\xd5\xeb\xcd\x99\xbf\xfdyW\x8e\xe1}\xe8\x8bL\x14D5\x7a\xea\xac\xf5F\xb4\xda\xaa\xe6\x90\x1 1\xa5\x95a\r\x8dXJ%\xb6\xe2'\x80'"
radar	4.7684E-06	b"?\xaa\xdc\xef\x1b<\xad\xf7\xef\xa0\xb2\xef\xa2r\xe0\xda\xc9\xdb\x8a\x1a\xfc*\xc1\xc1Y\xc6\x8e\x01\x8cm@\\xc9\x84\xed\x01\x81\xcd\xde\xdb\tHl\x9a\xe5\xeb\xe3;\xd5\xcd*.B\xc8\xeb\t(#\xc9\x98\xff\xadn\xb3\x12\xbc\xb2"
camera	5.2452E-06	b'1g\xaf&\x0c\x82\xa5\xa1\xbdc\xc0/\xdaB\xc5\xbf\x13v\xccy\xe7\xa1P\xe3I\xc5&^{\xa2*a\xfc\xd8\x89\xf5\x87\nk\x1e\x8dyd\x14\x0er\xe7V|ob'\xeb\xf3qk,\xcf\x8f^6I\xc9+\x01K\xa1d\xeba\xfe\xa2\xa4n'
lidar	7.8678E-06	b'\xc8\xde\xdf\xae\x1f\xacTNH\xf0\xfe\xd1\xe9\x93[5qf\x1a\xe9\xd4\xab2\xdd}z"\xc4\xa3\x11\xc5\x18J\xe4.:\x8d\xd1sG\xc3\xff=\xf5\xc4\x05\x83\xde\x82\\\xa\x82'=xd2\xaf0\x7f\x1bmg\xdb\xa8\xc7\x8b\xe2\xf1\xd8\xabk[\xe5\xb7\xc0&!C\xac)8\x82'
radar	6.1989E-06	b'\xa9\x95\xc5\xec*\xaf\xc6\xb2\xd7\xb4\xf7kT\xde3.\xdcT\x16\x91\x9b\xa1\x15-\x84A\x1fa\xe6.%\x96qE\xbe\xa0.\xe2!t\xdd#\x99\xd9!\xe0~\xc6j7\xc4%f9{\xa4\x9e<\x80\xa6\x8a]8Q\x10j0'
camera	5.2452E-06	b"G\x1f,\xacxr\x05\x08\x00\xa2\xe3\x8f\x16\x0b<o\xfb\xb22\x9aS\x84\xb4\x80\x9d\xe1\x93\xf6N\xcd\xd1d\x1f\xf7\xd77\x8c1\xd4_%\xa2\xc0\xdd'w/!\xeb\xc1}\x88g:\xb2\xc8\xf6\x8dO\x82\xaf\xbd\xc2Gt\x07\x87\x93\xef\x1ef\x06\xeetO"
lidar	4.7684E-06	b"\xd9T\x8ciA\x\xf41\xabs\xb8\x1e\t\xd8\x96\xdbx)\xb9\xff,\xf7\x80D\x1a\xb0\x99<\xbf\xb4\xdf8\x08\xc1\x98\xf8CQ\xe9\x9f\x9d\x10ho\xffG,d\xff\xd6$\x8f\'\xf8K\x8d\xfc\x0f\n\xc7\xa0!w\xa5[U\xd9\x07yb\xeb\xec\xad\xed\xef\xa01O\xa3\x18\x9f\xa4\xd0\xbeN'
radar	4.7684E-06	b"\xf6\x8f5b\\wR\x91\xc8\x7f\xe2N\x17\x0cb\xc0\xfc\t\x16\x99\xbb\xdc\x18\x9f\x04\x12\x04\x85j\xccT\xa0\x f93\xbb\xfc\x85 2\xa8D\x13\xcd\x088}\xe7t\x01\x90N$\xf8\xd2\xbae\\\x81p\xde8n0f\x83p\x15'
camera	5.0068E-06	b'L/L{\x07\x1dol\xad}>\x9a\xeb\xac\xce\x13\xd4\xed\x03\xbb\x14\x9e|9\xe8\xday\x15\xfd\x85I\xee9\xc5\xa8\xe0\xed\xda^\x8f\x07\x86\xce\xafq^\xba\xb5E\xc5\xad>\xdb\xcceK\xa3\x9f\x02\xb91\xc4\xb9\r\x89\xd7:VV\x89\xe4[!\xe1'
lidar	4.7684E-06	b"\xf3r\xe5\xbeT\xdc\xfd\x11\xaf+\xb5\x99C3\x88\xa3\x13\x01\xb5+&\xf1\x9c\x9d\x17\xbf\xf5\xa0=\xc1\xccUD\xf9\xa5\xe8#A\xff}\x98\x0b\x96\x08\x98"\xc6=\x85\x14\xac]T\xea\xa1\xdfB\xa1\xd2P\xc8\xf5\xbe\x96 ~\xc0\xba\xce\xc0M\x91hd\x9a\x14\x91\xf8\xcdc\x8c\xed&\x1b"
radar	5.0068E-06	b',q\xc6a\xbb\xebgN\x03\xc2\xed\xf6K2\x93\xea\x10\x17\xe3\x88$\xd4<\xf3z]\x87\xf6 j\x1e@v\xd9\x82\x01D`}P\xf9\xae\xb4\xc2"E4\x15\xd9\xddQ\xd3{\xf1\xe5ALu\x9f,[3\x85\xdd\x9c\xb6D'

- Sensor Type: This column represents the type of sensor from which the data is collected. The data is categorized into three types: Camera, Lidar, and Radar. Each of these sensors generates different types of data (visual, point cloud, and radar signals, respectively).
- Encryption Time (seconds): This column shows the time taken to encrypt the data from each sensor using the AES GCM encryption method. The encryption time is measured in seconds and provides insight into how long it takes to secure the data before it is transmitted. For example, the encryption time for the camera sensor data is approximately 0.000007 seconds, while for radar it is slightly lower at 0.000005 seconds. This variation in encryption times is due to differences in the size and complexity of the sensor data.
- Encrypted Data: The encrypted data column contains the actual encrypted bytes for each sensor. This is the result of applying AES encryption to the sensor data. The encrypted data is presented as a series of hexadecimal values (in bytes), which are unreadable without the decryption key.

Journal of Information Systems Engineering and Management
2025, 10(41s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

V. TRADE-OFFS IN SECURE SENSOR DATA TRANSMISSION FOR AUTONOMOUS SYSTEMS

In real-time systems, particularly those involving autonomous vehicles or robotics, securing sensor data during transmission is a critical task. However, the process of encryption and decryption comes with several trade-offs that must be carefully evaluated to maintain the system's overall performance. These trade-offs primarily involve the balance between security, performance, and efficiency. Below are the key trade-offs encountered in the secure transmission of sensor data:

1) Security vs. Performance

a) Impact on Transmission Speed: Encryption algorithms, while ensuring data confidentiality and integrity, introduce delays due to the computational overhead required for encrypting and decrypting the data. Stronger encryption methods, such as AES GCM, provide higher security but can slow down the system by increasing encryption and decryption times. This delay can be particularly challenging for real-time systems that rely on quick decision-making and responsiveness, such as autonomous vehicles.

b) Data Size: Larger sensor data (e.g., from cameras and lidars) requires more processing power and time to encrypt compared to smaller data from sensors like radar. This creates a performance bottleneck, where larger datasets take longer to encrypt and transmit, potentially delaying real-time decisions.

c) Encryption Level: Increasing the security level by opting for more complex encryption methods (e.g., AES-256 instead of AES-128) can significantly improve data protection, but it also increases the computational load, impacting system latency and throughput. Hence, a balance must be struck between the level of encryption and the real-time performance requirements of the system.

2) Latency vs. Data Integrity

a) Decryption Delay: Decryption times contribute to the latency of the system, as encrypted data must be decrypted before it can be used. While decryption time is typically lower than encryption time, excessive delays in decryption could impair the effectiveness of the system, especially in time-sensitive applications. Autonomous vehicles, for instance, need to process sensor data and make decisions in real-time to avoid collisions or navigate efficiently. A delay in decryption could result in unsafe or delayed responses.

b) Accuracy and Integrity: Higher encryption methods ensure data integrity and prevent unauthorized tampering, but they might also introduce additional processing time, increasing latency. On the other hand, reducing encryption complexity to minimize latency could expose the system to security risks, potentially allowing adversaries to manipulate or intercept data.

3) Complexity vs. Usability

a) Implementation Complexity: More sophisticated encryption techniques can offer better protection but at the cost of increased implementation complexity. For example, integrating AES GCM encryption with advanced decryption strategies can require additional system resources, specialized hardware, or software optimizations. This could increase development and maintenance costs.

b) User Experience: For autonomous systems relying on secure sensor data, usability is paramount. The increased computational complexity may also introduce operational challenges, such as higher energy consumption or the need for more powerful hardware, which may not be feasible in all deployment scenarios (e.g., in edge computing or low-power embedded systems).

4) Scalability vs. Security

a) Scaling the System: As the number of sensors in a system increases, so does the volume of data to be encrypted and transmitted. Scaling up security measures for each sensor may lead to a significant increase in processing time and storage requirements. In large-scale systems, like fleet management for autonomous vehicles, encrypting large volumes of sensor data can become a major bottleneck unless highly optimized encryption techniques or distributed encryption schemes are implemented.

Journal of Information Systems Engineering and Management
2025, 10(41s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

b) Maintaining Security at Scale: The challenge arises when ensuring that encryption performance scales efficiently with increasing sensor data or network load. While the encryption of individual sensor data streams may be manageable, encrypting data from multiple sensors simultaneously can strain system resources. It's essential to balance the need for security with the system's ability to scale without performance degradation.

5) *Energy Efficiency vs. Security*

a) Energy Consumption: Encryption algorithms, especially those that are computationally intensive like AES GCM, can significantly increase energy consumption, especially in mobile or embedded systems. Autonomous vehicles or robots with limited battery resources may experience a reduction in operational time as the energy demands of encryption and decryption grow.

b) Optimizing Power Consumption: While stronger encryption enhances security, it also leads to increased power consumption. This trade-off is crucial in autonomous systems that rely on energy efficiency to maximize operational time without recharging. Selecting the appropriate encryption strength and optimizing power consumption are important considerations for battery-powered systems.

The trade-offs in secure sensor data transmission systems are a balance between ensuring data security and maintaining the system's real-time performance. The challenge lies in selecting the optimal encryption methods that offer adequate protection while meeting the performance requirements of autonomous systems. Finding the right balance between security, latency, energy consumption, and scalability is essential to ensure both safe and efficient operation of autonomous systems in dynamic, real-time environments.

VI. CHALLENGES

While this research aims to provide a secure method of transmitting sensor data from autonomous systems, several challenges were encountered throughout the development and implementation stages.

A. *Data Size and Computational Overhead:* One of the primary challenges was the size of the sensor data generated by different sensors such as cameras, lidars, and radars. These sensors produce large amounts of data, which significantly impacts the computational resources required for encryption and decryption. Encryption processes, particularly with large datasets, can be time-consuming and resource-intensive, affecting the overall system performance. This issue was compounded by the need for real-time data transmission, demanding an efficient and optimized encryption mechanism.

B. *Latency and Time Sensitivity:* Given that autonomous systems rely on real-time data for decision-making, the encryption and decryption processes must be highly optimized to minimize latency. Any delay in the data transmission can affect the system's ability to make timely decisions, potentially jeopardizing the safety of the system. Balancing the need for robust security and low latency was a significant challenge, requiring a balance between algorithmic efficiency and computational time.

C. *Sensor Data Variability:* Different sensors (e.g., cameras, lidars, and radars) produce varied data formats and sizes, leading to the challenge of standardizing the encryption process. Each sensor's data needs to be treated differently, and ensuring consistency in how the data is encrypted, transmitted, and decrypted across these sensor types added complexity to the system's design.

D. Data Size and Computational Overhead: One of the primary challenges was the size of the sensor data generated by different sensors such as cameras, lidars, and radars. These sensors produce large amounts of data, which significantly impacts the computational resources required for encryption and decryption. Encryption processes, particularly with large datasets, can be time-consuming and resource-intensive, affecting the overall system performance. This issue was compounded by the need for real-time data transmission, demanding an efficient and optimized encryption mechanism.

Journal of Information Systems Engineering and Management
2025, 10(41s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

E. Latency and Time Sensitivity: Given that autonomous systems rely on real-time data for decision-making, the encryption and decryption processes must be highly optimized to minimize latency. Any delay in the data transmission can affect the system's ability to make timely decisions, potentially jeopardizing the safety of the system. Balancing the need for robust security and low latency was a significant challenge, requiring a balance between algorithmic efficiency and computational time.

F. *Sensor Data Variability:* Different sensors (e.g., cameras, lidars, and radars) produce varied data formats and sizes, leading to the challenge of standardizing the encryption process. Each sensor's data needs to be treated differently, and ensuring consistency in how the data is encrypted, transmitted, and decrypted across these sensor types added complexity to the system's design.

G. *Integration of Secure Communication Protocols:* Implementing secure communication channels using Transport Layer Security (TLS) to protect the transmitted sensor data posed challenges related to compatibility, configuration, and management of security keys. Ensuring that each sensor's data was securely transmitted over the network while maintaining the integrity of the data and preventing unauthorized access required in-depth knowledge and fine-tuning of secure communication protocols.

H. *Decryption and Data Integrity:* A major concern with encrypted data is ensuring the integrity and authenticity of the received data. This challenge is particularly relevant in real-time systems where data corruption during transmission can lead to erroneous decision-making. Implementing robust decryption methods that ensured the correct data was retrieved and verified without compromising security was crucial but complex.

I. *Performance Metrics and Analysis:* Measuring and analyzing the performance of the encryption and decryption processes across different sensors was another challenge. The encryption times, data sizes, and transmission counts needed to be effectively monitored to identify potential bottlenecks and inefficiencies. Developing meaningful performance metrics that could guide optimization efforts was essential for ensuring that the system met both security and performance requirements.

J. *Scalability:* As the number of sensors increases in an autonomous system, the scalability of the encryption mechanism becomes a significant concern. Managing multiple sensors and ensuring that the encryption and decryption processes scale efficiently while maintaining secure communication is a difficult task. Addressing scalability challenges was critical to the future-proofing of the system as the number of sensors in autonomous vehicles or robotic systems is expected to grow.

K. *System Complexity:* Finally, the overall complexity of integrating secure data transmission, real-time processing, and efficient data encryption/decryption into a cohesive system posed various difficulties. It required careful consideration of each component's interdependencies and the overall system architecture to ensure seamless operation.

Despite these challenges, overcoming them has provided valuable insights into designing secure, efficient, and scalable systems for autonomous vehicles and other real-time applications. The results demonstrated that it is possible to encrypt and securely transmit sensor data with minimal overhead, although continuous improvements in computational efficiency and system optimization will remain essential as sensor technologies advance.

VII. FUTURE CHALLENGES AND DIRECTION

As autonomous systems continue to evolve, several challenges and directions for future work remain, particularly in the context of sensor data security, real-time processing, and system optimization. While the current project demonstrates significant strides in securely transmitting sensor data, there are several areas where improvements are needed to address emerging challenges.

Journal of Information Systems Engineering and Management
2025, 10(41s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

A. *Advancement in Encryption Algorithms:* One of the main challenges for the future lies in the continuous development of encryption algorithms that can handle increasing data sizes without compromising the system's performance. Current encryption methods, like AES GCM, are robust, but as sensor technologies improve and data volumes grow, there will be a need for more efficient algorithms that can balance security and computational efficiency. Future research could explore lightweight encryption algorithms or hybrid encryption techniques to optimize both performance and security.

B. *Real-Time Decryption and Low Latency:* Autonomous systems rely heavily on real-time processing, where every millisecond counts. Future challenges will revolve around further reducing latency in encryption and decryption processes, ensuring that the system can still function in near real-time while maintaining the integrity of the data. The ability to perform secure, high-speed encryption and decryption without bottlenecks will be crucial as autonomous systems continue to grow in complexity and functionality.

C. *Adaptive Security Protocols:* As autonomous vehicles and robots interact with various environments, the security needs of the system will evolve. Future systems will require adaptive security protocols that can adjust based on the type of data being transmitted, the environment, and the system's current workload. This could involve adaptive key management strategies, dynamic encryption levels, and flexible security policies based on risk assessments in real-time, thus optimizing security without unnecessary computational overhead.

D. *Multi-Sensor Data Integration:* With the increasing number of sensors (cameras, lidars, radars, etc.) integrated into autonomous systems, there is a growing challenge of securely managing and integrating data from various sources. Multi-sensor fusion techniques need to be developed that allow for efficient and secure integration of data from heterogeneous sensors, all while maintaining the integrity of each data stream. Future efforts should focus on developing protocols that enable the secure and seamless fusion of data across different sensor types to improve decision-making accuracy in autonomous systems.

E. *Quantum Computing and Cryptography:* The advent of quantum computing presents a potential threat to current encryption methods. While quantum computers are not yet widely available, research into post-quantum cryptography is necessary to prepare for a future where quantum computing might break existing cryptographic protocols. Future work will need to focus on implementing quantum-resistant encryption methods to ensure the security of sensor data transmission in the long term.

F. *Scalability and System Performance:* As autonomous systems become more complex with more sensors and data sources, scalability will remain a significant challenge. Ensuring that the system remains secure and performs optimally as the number of sensors increases requires continuous innovation in system architecture and communication protocols. Future systems will need to adopt scalable encryption and decryption mechanisms, along with efficient data transmission protocols, to handle the demands of large-scale, multi-sensor environments.

G. *Edge Computing and Distributed Systems:* To address the latency concerns and offload computational demands, edge computing could play a vital role in processing sensor data closer to the source, such as on the vehicle or robot itself. Implementing secure encryption and decryption processes at the edge, rather than sending all data to centralized cloud servers, will reduce latency and improve performance. Future directions will focus on developing secure, decentralized systems that can handle real-time data processing at the edge while maintaining high levels of security.

H. *Data Integrity and Trustworthiness:* With the increasing reliance on sensor data for autonomous decision-making, ensuring the integrity and trustworthiness of the data becomes critical. Future research should explore advanced data validation techniques that can verify not only the authenticity of the data but also ensure that it has not been tampered with during transmission. This could involve using blockchain-based systems for secure data logging or implementing advanced anomaly detection techniques to identify compromised data.

Journal of Information Systems Engineering and Management
2025, 10(41s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

I. *User Privacy and Compliance:* As autonomous systems collect and transmit large amounts of personal or sensitive data, there will be an increasing emphasis on user privacy and data protection. Future challenges will involve ensuring compliance with global privacy regulations (e.g., GDPR) while maintaining secure and efficient data transmission. Research will need to focus on implementing privacy-preserving encryption methods and ensuring that data handling practices meet legal and ethical standards.

J. *Collaboration and Standardization:* The growing number of players in the autonomous systems market creates challenges in terms of interoperability and standardization. Future work will need to focus on creating industry-wide standards for sensor data encryption, communication protocols, and security measures. Collaborative efforts between industry leaders, governments, and research institutions will be key to developing common frameworks for securing data in autonomous systems.

In conclusion, while significant progress has been made in securing sensor data in autonomous systems, there are numerous challenges that need to be addressed in the future. By focusing on advanced encryption methods, adaptive security protocols, and real-time processing, the security and performance of autonomous systems can be further optimized. As the technology continues to evolve, ongoing research and development will be crucial to ensuring that autonomous systems can operate safely, efficiently, and securely in increasingly complex environments.

VIII. CONCLUSION

The results of this study demonstrate a solid foundation for the secure transmission of sensor data across various sensor types, including cameras, lidars, and radars. The encryption and decryption times are consistent with the expected behavior, given the differences in the size and complexity of data generated by each sensor. The camera data, being more complex and larger in size, takes slightly longer to encrypt and decrypt compared to lidar and radar data. However, all three sensor types exhibit reasonable performance, with encryption times decreasing after initial transmissions, indicating that the system becomes more efficient over time.

One key observation is that while the encryption times increase with data size and transmission count, they remain within acceptable limits. Radar data, which has a smaller size compared to camera and lidar data, shows the lowest encryption time, reflecting its simpler structure and smaller data volume. On the other hand, the lidar sensor data, with its larger point cloud data, requires more computational effort for encryption and decryption, as seen in the higher encryption times.

The transmission counts for each sensor (camera: 10, lidar: 15, radar: 20) provide a diverse range of data points for evaluating the system's scalability, demonstrating that the system can effectively handle varying transmission loads. The bar charts highlighting sensor data size further underline the differences in data handling requirements, with lidar generating the largest data size, followed by camera and radar. This insight is particularly useful when considering network bandwidth and storage capacities.

However, the performance could be further optimized in terms of reducing encryption time for larger data sizes, especially for lidar and camera data. The initial spikes in encryption times observed in the early transmissions indicate a potential area for improvement in reducing latency during the start of transmissions.

Overall, the system demonstrates robust performance in secure data transmission, but there is room for optimization, particularly in improving encryption efficiency and handling larger data sets, which could further enhance its application in real-time autonomous systems.

IX. ACKNOWLEDGMENTS

I would like to express my sincere gratitude to Dr. Pavan Kumar Gautam for his invaluable guidance and support throughout the course of this research. His expertise and encouragement were pivotal in shaping the direction of this study. I truly appreciate his dedication and willingness to provide thoughtful feedback, which has greatly contributed to the successful completion of this work.

Journal of Information Systems Engineering and Management
2025, 10(41s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

REFERENCES

[1] M. Khan, Z. Jan, J. Ahmad, and Z. Khan, "An Adaptive Secret Key-directed Cryptographic Scheme for Secure Transmission in Wireless Sensor Networks," *arXiv preprint arXiv:1510.00226*, 2015. Available: https://arxiv.org/abs/1510.00226.

[2] D. Kurbanmuradov, V. Sokolov, and V. Astapenya, "Implementation of XTEA Encryption Protocol based on IEEE 802.15.4 Wireless Systems," *arXiv preprint arXiv:1912.12043*, 2019. Available: https://arxiv.org/abs/1912.12043.

[3] Zigbee Alliance, "Zigbee Specification," *Zigbee Alliance*, 2011. Available: https://en.wikipedia.org/wiki/Zigbee.

[4] IEEE Security in Storage Working Group, "IEEE P1619 Standard for Cryptographic Protection of Data on Block-Oriented Storage Devices," *IEEE Standards Association*, 2007. Available: https://en.wikipedia.org/wiki/IEEE_P1619.

[5] IEEE 802.1 Working Group, "IEEE 802.1AE-2018: Media Access Control (MAC) Security," *IEEE Standards Association*, 2018. Available: https://en.wikipedia.org/wiki/IEEE_802.1AE.

[6] D. A. McGrew and J. Viega, "The Galois/Counter Mode of Operation (GCM)," *NIST Special Publication SP800-38D*, 2007. Available: https://en.wikipedia.org/wiki/Galois/Counter_Mode.

[7] Y. Zhang and L. Wang, "Exceptional Key Based Node Validation for Secure Data Transmission in Wireless Sensor Networks," *Procedia Computer Science, vol. 147*, pp. 87-94, 2019. Available: https://www.sciencedirect.com/science/article/pii/S2665917424001260.