Design and Implementation of a Secure IoT-Based Environmental Monitoring System

Jitendra Singh Kustwar[1], Gourav Shrivastatva[2], Chandan Kumar[3]

[1]Sanjeev Agrawal Global Educational (SAGE) University, Bhopal, India, 462026
[2] Sanjeev Agrawal Global Educational (SAGE) University, Bhopal, India, 462026
[3] Sanjeev Agrawal Global Educational (SAGE) University, Bhopal, India, 462026
[1]jeetsin312@gmail.com, [2] gourav.s@sageuniversity.edu.in, [3]chandan.k@sageuniversity.edu.in

Abstract. *Security is a primary concern for IoT-based environmental parameter monitoring systems that utilize sensors. Humidity significantly influences various chemical and biological properties of the environment, making it essential to monitor humidity and temperature using appropriate sensors. This paper outlines the design of a smart IoT-based secure environmental data monitoring system. A multi-level security system for the transceiver is developed. The proposed system employs secure key-based data encryption standards at the IoT transmitting sensor end. Encrypted data is transmitted via IoT over a private cloud channel. Ensuring data security during cloud server download to receivers is a key focus, addressed by a smart decryption algorithm. A smart IoT-based system for monitoring environmental parameters, specifically humidity and temperature, is designed, with the transceiver at its core. The paper discusses the results of secure encryption and decryption algorithms, presented in two phases: validation of standard AES encryption and implementation of multilevel public and private key decryption at the cloud end. The main focus is on designing a secure key-based AES encryption standard. Data is converted to 16-bit format and encrypted using AES-ECB2 mode before being uploaded to the IoT cloud. Results and timing analysis demonstrate the efficiency of the proposed encryption framework in decrypting data from the cloud. The transceiver employs a NodeMCU-based ESP8266 Wi-Fi module. Additionally, a Base64 AES decoding algorithm is presented in this work.*

Keywords: **-** Environmental Monitoring, IOT, AES, Encryption, Humidity sensor, Temperature sensor.

Introduction

With the exponential growth in population and industrial activities, environmental pollution has become a pressing concern, posing significant threats to human health [1]. Among the various forms of pollution, air pollution stands out as a major contributor. The dense population contributes to increased environmental humidity, exacerbating air quality issues [2]. In response to these challenges, the Internet of Things (IoT) based sensor networks offer promising solutions for remote monitoring of environmental parameters. In recent years, there has been substantial development in smart monitoring systems for sensor data, driven by advancements in IoT technology [3]. For applications such as environmental weather forecasting, ensuring the privacy and integrity of measured data is crucial. Consequently, designing secure parameter monitoring systems has become an essential aspect of IoT technology [4]. Sensors, which measure physical data such as humidity and air temperature, play a critical role in this process, converting these measurements into equivalent digital values [5]. The IoT has gained significant popularity due to its potential to revolutionize various industries, including environmental monitoring [6]. However, the implementation of IoT systems still faces numerous connectivity issues and technical challenges. Dense deployment of wireless sensors, which can significantly improve capacity, is primarily limited by power availability [7]. This dense deployment has become feasible with the advent of devices such as the ESP8266 router [8]. The ESP8266 integrates data processing and Wi-Fi connectivity on a single board known as NodeMCU [9]. This integration simplifies the development of IoT networks, providing wireless access through internet connectivity. As a result, IoT network developers can now deploy comprehensive environmental monitoring systems more efficiently [10]. Despite these advancements, there remain significant challenges to overcome. These include ensuring long-term power supply for sensors, maintaining data security [11], and managing large-scale sensor networks. Future research must focus on addressing these issues to fully harness the potential of IoT-based environmental monitoring systems [12].

II. Basic Architecture of SEMS

There is much architecture for smart environments; in fact, the architecture [13], in most of the cases, is based on the application. However, in this section, we present the most common architectures for smart environmental monitoring applications [14]. As can be seen in Figure 1, there are four basic layers of the architecture, which are the physical layer, OS abstraction layer, middleware layer, and application components layer [15]. The physical layer consists of different components, including the smart devices and the communication interface [16]. Smart devices could be smart sensors, such as pH, oxygen, SO2, etc., as well as humidity and temperature sensors [17]. The communication component could be wired as in indoor applications or wireless as in outdoor applications. Smart environmental networks are usually utilizing the ZigBee [18] and Bluetooth [19] communication standards in addition to satellite communication.

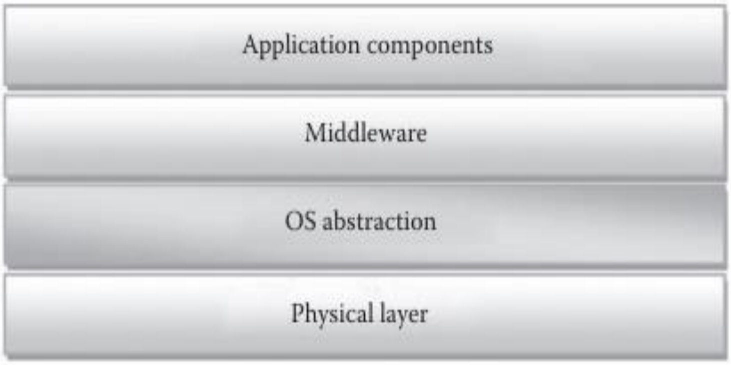


Figure 1: Basic components of smart environmental monitoring system.

The OS abstraction layer is the kernel of any smart environmental monitoring in terms of including the basic required instructions for environment operations [20]. Above the OS abstraction layer sits the middleware layer, which includes the message controllers and communication channel identifier. In addition, it is the interface between the application components and the OS abstraction layer in which different applications might require different requirements [21]. The final layer is the application components layer, in which it is assumed that the application consists of many components that might be implemented separately. At the same time, these components can communicate to each other through the middleware layer [22].

III. Literature Review

There are many researchers who have worked over environmental parameter monitoring. This section sequentially reviewed the related work in this domain.

Renee Obringer and Roshanak Nateghi [1] reviewed recent literature on smart cities and climate change, focusing on urban resilience aspects like infrastructure, public health, accessibility, sustainability, and governance. They found a higher emphasis on infrastructure resilience and varied connections between smart city initiatives and climate change efforts. P. Velmurugadass et al. [2] proposed a novel framework for monitoring data activities using Cloud-based Software Defined Network (SDN) with IoT devices, blockchain, and encryption algorithms. Their system showed improved performance in various metrics. Meryam Saad Fadhil et al. [3] designed a secure IoT system using Lightweight AES encryption on Raspberry Pi, showing better encryption/decryption time and throughput compared to related works. Qing Song Zhang et al. [4] proposed a cloud-based system utilizing wireless sensor networks for environmental pollution monitoring in smart cities, demonstrating effective pollution reduction through continuous monitoring. Samiulla Itoo et al. [5] presented a protocol for mutual authentication and key exchange in smart agriculture monitoring systems, ensuring superior security and efficiency compared to existing protocols. J. Ananth et al. [6] developed a system for individual vehicle air pollution monitoring using IoT, aiming to reduce air pollution-related deaths in India. Asia Othman Aljahdali et al. [7] proposed an access control mechanism using distributed ledger technologies for data protection, demonstrating feasibility and security. Pranav Gangwani et al. [8] explored blockchain's role in environmental monitoring, focusing on IOTA for secure data sharing among IoT devices. Y Hajjaji et al. [9] conducted a systematic review highlighting big data and IoT integration's potential for smart environment applications. Joonsuu Park and KeeHyun Park et al. [10] proposed a hierarchical smart dust monitoring system to reduce traffic load and increase device connectivity, achieving significant performance improvements. Rehman et al. [11] introduced a security model for smart cities utilizing the Green Internet of Things with Cloud Integrated Data Management, addressing connectivity, key distribution, and transmission security. Bingbing Fang et al. [12] reviewed AI applications in waste management, showing improvements in efficiency and cost savings. Seyyed Keyvan Mousavi et al. [13] compared encryption algorithms for IoT security, finding Elliptic Curve Cryptography to be superior in performance. Amin Ullah et al. [14] discussed challenges and case studies of IoT and machine learning in smart cities, emphasizing the transformative potential and associated challenges. Md. Milon Islam et al. [15] proposed a smart healthcare system using IoT sensors for real-time patient monitoring, demonstrating effectiveness in healthcare management. Silvia Liberata Ullo et al. [16] reviewed sensor, IoT, and machine learning applications in environment monitoring, suggesting advancements in smart monitoring systems. L. Mary Shamala et al. [17] discussed IoT cryptography, security threats, and solutions, highlighting the importance of securing IoT systems. Faris A. Almalki et al. [18] presented a low-cost platform for environmental monitoring using flying IoT, enhancing crop productivity and farm management. Anuj Kumar Singh et al. [19] proposed an energy-efficient data transmission model for WSNs, comparing private key cryptography algorithms for encryption. M. Udin Haru et al. [20] developed an IoT device for environmental monitoring using the KAA platform, achieving real-time monitoring and data accessibility. Riyadh Arridha et al. [21] extended the SEMAR project with water quality analytics, achieving high estimation accuracy using machine learning algorithms. Yakub Kayode Saheed et al. [22] developed an IDS using DRNN and supervised machine learning for cyber threat classification in IoMT environments, achieving high accuracy. Sharnil Pandya et al. [23] discussed federated learning applications for smart cities, outlining current and future developments and research challenges. Hayder Mahmood Salman et al. [24] discussed the use of artificial intelligence in healthcare environmental design, highlighting advancements and challenges in smart systems.

These studies collectively contribute to advancing knowledge and technologies for creating more sustainable, efficient, and secure smart cities and IoT systems.

IV. Proposed IoT Based Environmental Monitoring System (EMS)

The dense population significantly contributes to increased environmental humidity. This paper describes the design of an IoT-based smart monitoring system utilizing sensor-based data acquisition for environmental parameter monitoring. The IoT sensor network enables remote monitoring of environmental conditions, with a strong emphasis on data privacy and integrity for weather forecasting applications. To address these needs, the EMS system incorporates a NodeMCU-based ESP8266 Wi-Fi router at the transceiver end. This section presents a secure environmental monitoring system suitable for applications in air conditioning, ventilation, and weather monitoring, with data stored on the ThingSpeak cloud server. Connectivity issues pose a significant challenge for IoT system design. Thus, the proposed EMS system aims to efficiently design the transmitter and incorporate secure data encryption using a key algorithm. Dense deployment of wireless sensors enhances capacity, constrained only by power availability, made feasible by devices such as the ESP8266 router. This chapter details the transceiver's hardware setup and components, along with technical specifications, and describes the basic encryption and decryption processes for IoT data. The proposed EMS system's block diagram is illustrated in Figure 2. This secure EMS system offers wireless access to sensor data via internet connectivity, using the NodeMCU integrated Wi-Fi board for data transmission over the IoT network. The work focuses on designing a secure IoT-based environmental parameter monitoring system, emphasizing the successful implementation of an AES encryption framework for sensor data. Humidity and temperature, critical environmental health indicators, are measured using the DHT11 digital sensor, which reads both parameters via a three-pin interface.

The proposed system promotes secure data transmission within organizations, addressing critical security issues in the IoT framework. It not only encrypts sensor data but also enhances security at the cloud server, ensuring protection all the way to the client end. This approach leverages Advanced Encryption Standard (AES) cryptographic methods to achieve high efficiency during the encryption process. In detail, the system collects data from various IoT sensors, such as temperature and humidity sensors, air quality monitors, and other environmental parameter sensors. Once the data is gathered, it undergoes a robust encryption process using AES before being transmitted. AES, known for its reliability and strength, ensures that the data is securely encrypted, making it extremely difficult for unauthorized parties to access or tamper with the information. After encryption, the securely encrypted data is uploaded to cloud storage. The use of distributed storage solutions enhance the reliability and accessibility of the data, allowing authorized users to retrieve it from anywhere while maintaining high security standards. The cloud server itself is fortified with additional security measures to prevent breaches and ensure data integrity.

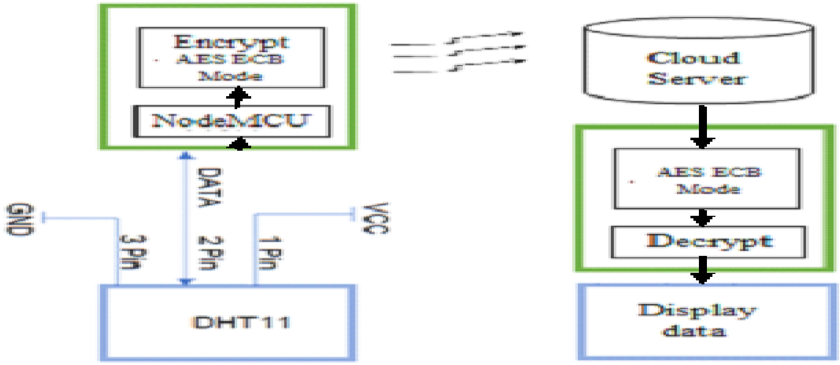


Figure 2: Proposed block diagram of the EMS.

This comprehensive approach not only safeguards the data during transmission but also ensures that it remains protected while stored in the cloud. By implementing these measures, the system addresses common vulnerabilities in IoT networks, providing a robust solution for secure environmental data monitoring. This ensures that the information is reliable and can be trusted for critical applications such as weather forecasting, air quality monitoring, and other environmental assessments. Furthermore, the system's design is scalable, allowing it to adapt to the increasing volume of data generated by the expanding network of IoT devices. This scalability ensures that as more devices are added, the system continues to perform efficiently, maintaining the security and integrity of the data. Overall, the proposed system represents a significant advancement in the secure transmission and storage of IoT sensor data, making it a valuable asset for organizations concerned with environmental monitoring and other applications where data security is paramount. The proposed structure used in this research of the IOT based Environmental monitoring system (EMS) is shown in the Figure 3.

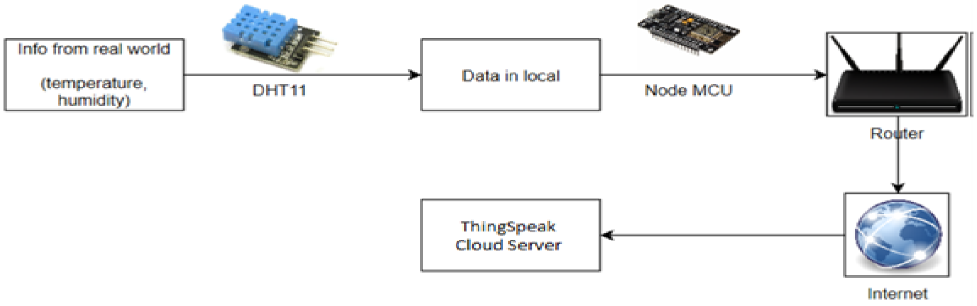


Figure 3: Proposed structure of the IOT based Environmental monitoring system (EMS).

Component Specifications Used For the Setup-

- **Bread Board-** For testing purposes of multiple pin connections
- **NoDeMCU-** It is an easily available open source Wi-Fi enabled integrated board for the IoT platform. Board is a ESP-12 module consisting of the ESP8266 Wi-Fi modem and hardware interfacing General-Purpose I/O port (GPIO), on board pins for (PWM), an 10 bit internal ADC.
- **DHT11-** It is a type of digital senior capable of simultaneous measurement of humidity and temperature. The sensor is used to monitor environmental parameters using capacitive humidity sensor and a Thermistors for air temperature using digital signal data.
- **Connecting Leads-** Mail pin and female pin connectors
- **Power Adopter-** A 5V fixed supply power adaptor to bring the NodeMCU continuously in network.

The DHT11 sensor is used in the proposed work for its cost-effectiveness and compatibility with NodeMCU and MicroPython firmware. It measures humidity resistively and temperature with a negative temperature coefficient (NTC). The sensor's benefits include a good response time and excellent interference avoidance, making it suitable for noisy environmental parameter monitoring systems. For this work, the DHT11 was carefully calibrated in a controlled laboratory environment.

The DHT11 sensor is advantageous due to its compact size and low energy consumption, with a transmission range of up to 20 meters. It features a 4-pin configuration arranged in a single row. Figure 4 provides a detailed description of the pin configuration for the DHT11 sensor.

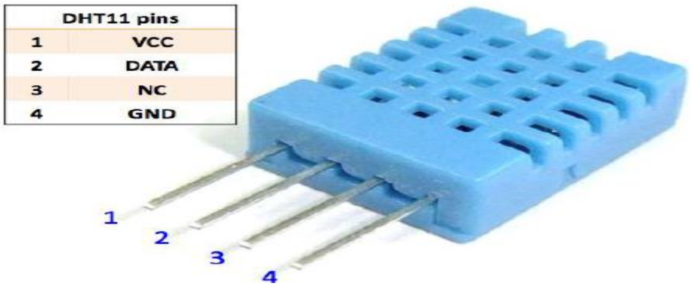


Figure 4: Pin description of the DHT sensor with 4 pin configuration.

V. IOT SECURITY IMPLEMENTATION

This paper proposes the use of the Advanced Encryption Standard (AES) algorithm to secure sensor data over an IoT cloud. While it does not delve into computation limitations, it emphasizes the security benefits of AES encryption. The goal is to implement a simple and efficient AES encryption method for IoT data.

Encryption at IoT Transmitter

1. **Connect to Access Point-** Connect to the designated Wi-Fi network using the SSID and password.
2. **Initialize DHT11 Sensor-** Set up the DHT11 sensor to measure temperature and humidity.
3. **Initialize AES ECB Mode-** Configure the AES encryption in Electronic Codebook (ECB) mode.
4. **Sample DHT11-** Measure temperature and humidity using the DHT11 sensor.

5. **Add Padding-** Ensure the temperature and humidity data lengths are multiples of 16 by adding padding.
6. **Encrypt Data-** Use a symmetric private key to encrypt the padded temperature and humidity data.
7. **Encode to Base64-** Convert the encrypted data to Base64 encoding.
8. **Convert to String-** Format the Base64 encoded values as a UTF-8 string.
9. **HTTP POST to IoT Cloud-** Send the data to the IoT cloud using an HTTP POST request.
10. **Repeat-** Return to step 4 to continue the process.

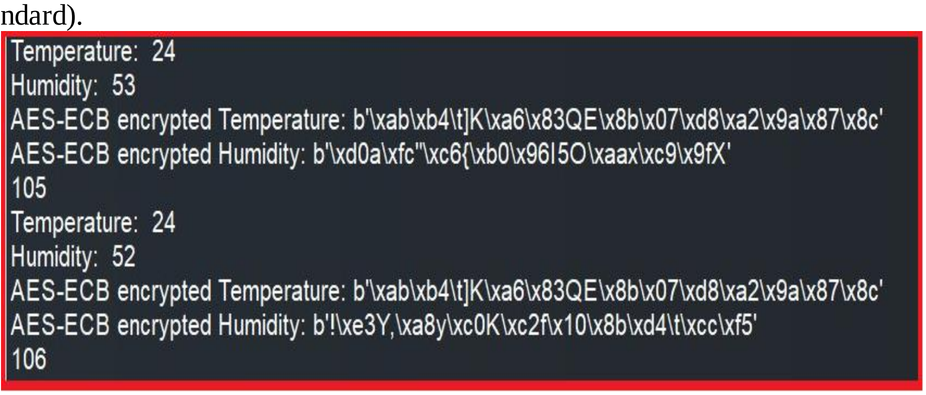
Decryption at IoT Receiver

1. **Connect to Access Point-** Connect to the designated Wi-Fi network.
2. **Initialize AES ECB Mode-** Configure the AES decryption in ECB mode.
3. **HTTP GET from IoT Cloud-** Retrieve the data from the IoT cloud using an HTTP GET request.
4. **Extract Cipher Text-** Extract the Base64 encoded temperature and humidity strings from the payload.
5. **Decode Base64-** Decode the Base64 encoded strings to retrieve the encrypted data.
6. **Decrypt Data-** Use the symmetric private key to decrypt the data.
7. **Repeat-** Return to step 3 to continue the process.

VI. Sequential AES Encryption Validation

Sequential measurement at AES encrypted data taken from sensor for humidity and temperature as shown in the Figure 5. Temperature in figure is 24 ^{o}C and Humidity is reported 53%.

The section presents the sequential results of encryption and decryption processes using AES (Advanced Encryption Standard).

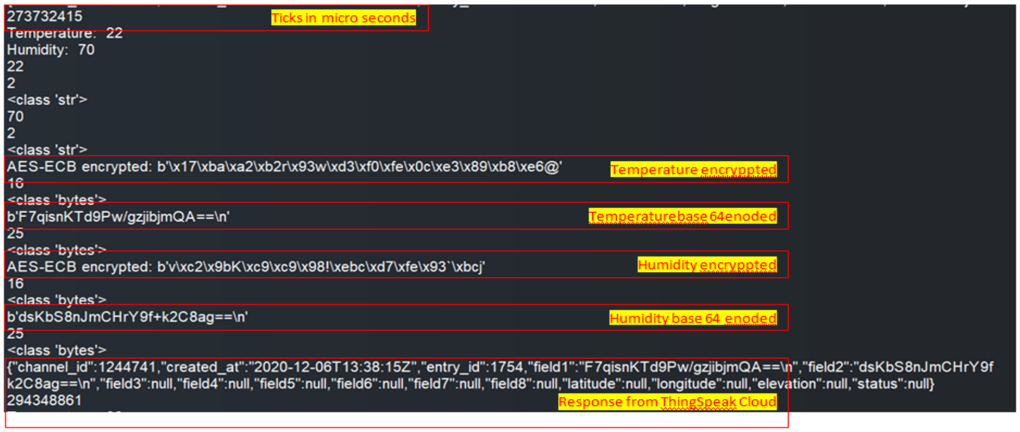


```
Temperature:  24
Humidity:  53
AES-ECB encrypted Temperature: b'\xab\xb4\t]K\xa6\x83QE\x8b\x07\xd8\xa2\x9a\x87\x8c'
AES-ECB encrypted Humidity: b'\xd0a\xfc"\xc6{\xb0\x96I5O\xaax\xc9\x9fX'
105
Temperature:  24
Humidity:  52
AES-ECB encrypted Temperature: b'\xab\xb4\t]K\xa6\x83QE\x8b\x07\xd8\xa2\x9a\x87\x8c'
AES-ECB encrypted Humidity: b'!\xe3Y,\xa8y\xc0K\xc2f\x10\x8b\xd4\t\xcc\xf5'
106
```

Figure 5: Data encryption at transmitter.

Figure 6 illustrates the flow of the AES encryption process implemented at the transmitter end using NodeMCU. This process involves steps such as connecting to Wi-Fi, initializing the DHT11 sensor for temperature and humidity data acquisition, encrypting the data using AES in ECB mode, encoding the cipher text to Base64, and finally sending the encrypted data to the IoT cloud via HTTP POST.

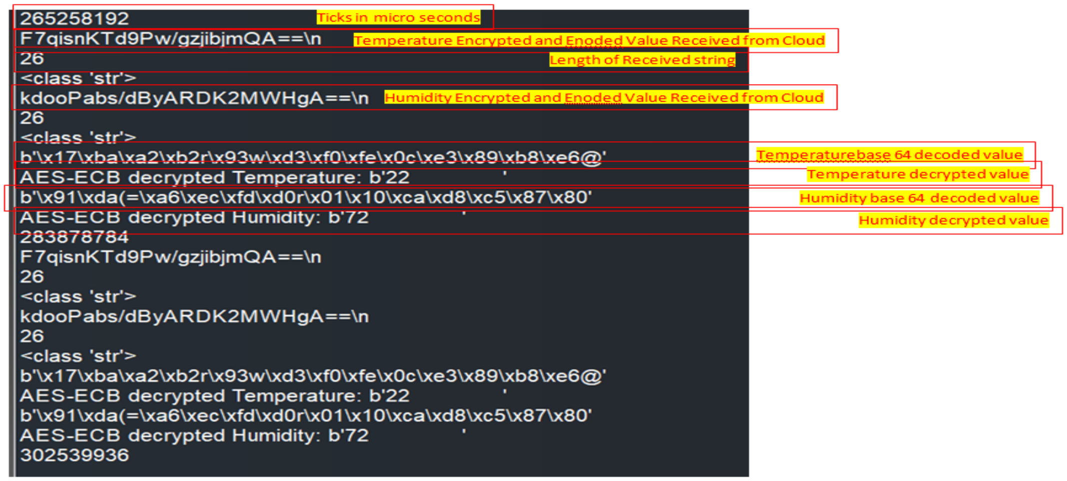
Figure 7, on the other hand, outlines the flow of the AES decryption process at the receiver end, retrieving data from the ThingSpeak cloud. This process includes connecting to Wi-Fi, performing an HTTP GET request to fetch the encrypted data from the cloud, decoding the Base64 encoded cipher text, decrypting the data using AES in ECB mode with the symmetric private key, and repeating this process as necessary. The decryption process ensures that the encrypted data received from the cloud is securely decoded and decrypted back into its original form (temperature and humidity readings) for further analysis or display.



```
273732415
Temperature:  22
Humidity:  70
22
2
<class 'str'>
70
2
<class 'str'>
AES-ECB encrypted: b'\x17\xba\xa2\xb2r\x93w\xd3\xf0\xfe\x0c\xe3\x89\xb8\xe6@'
16
<class 'bytes'>
b'F7qisnKTd9Pw/gzjibjmQA==\n'
25
<class 'bytes'>
AES-ECB encrypted: b'v\xc2\x9bK\xc9\xc9\x98!\xebc\xd7\xfe\x93`\xbcj'
16
<class 'bytes'>
b'dsKbS8nJmCHrY9f+k2C8ag==\n'
25
<class 'bytes'>
{"channel_id":1244741,"created_at":"2020-12-06T13:38:15Z","entry_id":1754,"field1":"F7qisnKTd9Pw/gzjibjmQA==\n","field2":"dsKbS8nJmCHrY9f
k2C8ag==\n","field3":null,"field4":null,"field5":null,"field6":null,"field7":null,"field8":null,"latitude":null,"longitude":null,"elevation":null,"status":null}
294348861
```

Figure 6: Timing flow at the IOT Transmitter during Encryption.

These figures and processes collectively demonstrate the end-to-end implementation of AES encryption and decryption in an IoT environment using NodeMCU and cloud services, ensuring secure transmission and retrieval of sensor data.

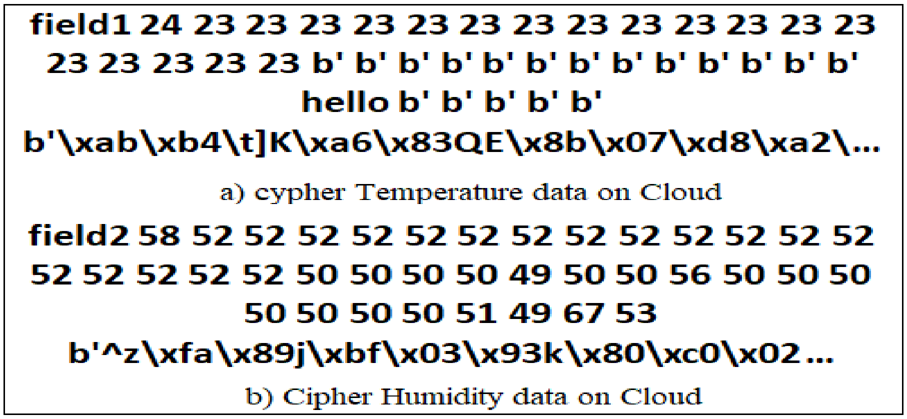


```
265258192
F7qisnKTd9Pw/gzjibjmQA==\n
26
<class 'str'>
kdooPabs/dByARDK2MWHgA==\n
26
<class 'str'>
b'\x17\xba\xa2\xb2r\x93w\xd3\xf0\xfe\x0c\xe3\x89\xb8\xe6@'
AES-ECB decrypted Temperature: b'22              '
b'\x91\xda(=\xa6\xec\xfd\xd0r\x01\x10\xca\xd8\xc5\x87\x80'
AES-ECB decrypted Humidity: b'72              '
283878784
F7qisnKTd9Pw/gzjibjmQA==\n
26
<class 'str'>
kdooPabs/dByARDK2MWHgA==\n
26
<class 'str'>
b'\x17\xba\xa2\xb2r\x93w\xd3\xf0\xfe\x0c\xe3\x89\xb8\xe6@'
AES-ECB decrypted Temperature: b'22              '
b'\x91\xda(=\xa6\xec\xfd\xd0r\x01\x10\xca\xd8\xc5\x87\x80'
AES-ECB decrypted Humidity: b'72              '
302539936
```

Ticks in micro seconds
Temperature Encrypted and Enoded Value Received from Cloud
Length of Received string
Humidity Encrypted and Enoded Value Received from Cloud
Temperaturebase 64 decoded value
Temperature decrypted value
Humidity base 64 decoded value
Humidity decrypted value

Figure 7: Timing flow at the IOT Receiver during decryption.

In the proposed system, encrypted sensor data in cipher form, represented as strings, is uploaded to the ThingSpeak cloud platform. Figure 8 displays the cipher text specifically for temperature and humidity sensors, showcasing how the encrypted data appears in its cryptographic form. Figure 9 illustrates the stored data within the ThingSpeak channel dedicated to humidity. This figure presents how the encrypted humidity data is organized and stored within the cloud platform, facilitating secure storage and subsequent retrieval as needed. These figures demonstrate the practical implementation of the AES encryption method for securing sensor data during transmission and storage on the IoT cloud. By encrypting data before uploading it to ThingSpeak, the system ensures that sensitive environmental parameters, such as temperature and humidity, remain protected and accessible only to authorized parties with the appropriate decryption keys. This approach enhances data security and integrity in IoT-based environmental monitoring systems.



```
field1 24 23 23 23 23 23 23 23 23 23 23 23 23 23
23 23 23 23 23 b' b' b' b' b' b' b' b' b' b' b' b' b'
hello b' b' b' b' b'
b'\xab\xb4\t]K\xa6\x83QE\x8b\x07\xd8\xa2\...
```

a) cypher Temperature data on Cloud

```
field2 58 52 52 52 52 52 52 52 52 52 52 52 52 52
52 52 52 52 52 50 50 50 50 49 50 50 56 50 50 50
50 50 50 50 51 49 67 53
b'^z\xfa\x89j\xbf\x03\x93k\x80\xc0\x02...
```

b) Cipher Humidity data on Cloud

Figure 8: Cipher data over the cloud server in string form.

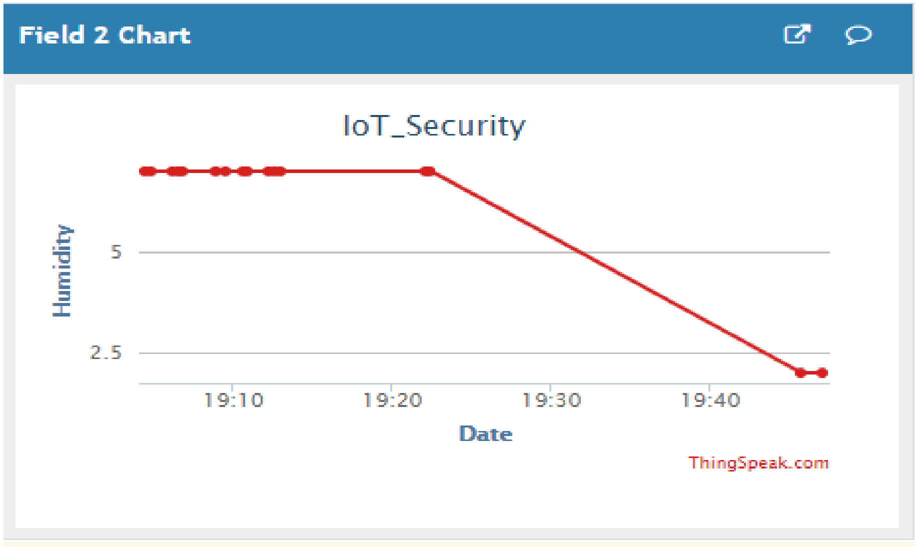


Figure 9: Results of the data over cloud.

VII. Conclusion and Future Scope

The rapid growth of population and industrialization has led to significant environmental pollution, particularly air pollution, which poses serious health risks. This paper presents a smart IoT-based environmental data monitoring system designed to address these challenges by providing reliable and secure monitoring of humidity and temperature. The system incorporates NodeMCU-based ESP8266 Wi-Fi routers and employs AES encryption with secure key standards to ensure data security during transmission and storage on the ThingSpeak cloud server. The use of humidity and temperature sensors allows for accurate monitoring of environmental parameters, which are crucial for various chemical and biological processes. The developed encryption framework efficiently converts data to 16-bit format and encrypts it using AES ECB mode, ensuring secure transmission of cipher data to the IoT cloud. The decryption algorithm at the receiver end further enhances data security. The experimental results demonstrate the system's effectiveness, with temperature readings ranging between 23-27°C and humidity between 48-56% in a room environment, and a data error of less than 4%. This secure and efficient environmental monitoring system has significant applications in air conditioning, ventilation, and weather monitoring, providing a robust solution for mitigating the effects of environmental pollution. Many more sensor nodes can be easily added to the system to expand its functionality. In the future, new sensors can be integrated with the system for added features to make the system more useful.

REFERENCES

[1] Obringer, Renee, and RoshanakNateghi. "What makes a city ‘smart’in the Anthropocene? A critical review of smart cities under climate change." Sustainable Cities and Society 75 (2021): 103278.

[2] Velmurugadass, P., et al. "Enhancing Blockchain security in cloud computing with IoT environment using ECIES and cryptography hash algorithm." Materials Today: Proceedings 37 (2021): 2653-2659.

[3] Fadhil, MeryamSaad, AlaaKadhimFarhan, and Mohammad NatiqFadhil. "A lightweight aes algorithm implementation for secure iot environment." Iraqi Journal of Science (2021): 2759-2770.

[4] Zhang, Qing Song. "Environment pollution analysis on smart cities using wireless sensor networks." Strategic Planning for Energy and the Environment (2023): 239-262.

[5] Itoo, Samiulla, et al. "A Secure and Privacy-Preserving Lightweight Authentication and Key Exchange Algorithm for Smart Agriculture Monitoring System." IEEE Access (2023).

[6] Maguluri, LakshmanaPhaneendra, et al. "Smart Vehicle-Emissions Monitoring System Using Internet of Things (IoT)." Handbook of Research on Safe Disposal Methods of Municipal Solid Wastes for a Sustainable Environment. IGI Global, 2023. 191-211.

[7] Aljahdali, Asia Othman, AfnanHabibullah, and Huda Aljohani. "Efficient and Secure Access Control for IoT-based Environmental Monitoring." Engineering, Technology & Applied Science Research 13.5 (2023): 11807-11815.

[8] Gangwani, Pranav, et al. "Securing environmental IoT data using masked authentication messaging protocol in a DAG-based blockchain: IOTA tangle." Future Internet 13.12 (2021): 312.

[9] Hajjaji, Yosra, et al. "Big data and IoT-based applications in smart environments: A systematic review." Computer Science Review 39 (2021): 100318.

[10] Park, Joonsuu, and KeeHyun Park. "Construction of a remote monitoring system in smart dust environment." Journal of Information Processing Systems 16.3 (2020): 733-741.

[11] Rehman, Amjad, et al. "M-SMDM: a model of security measures using green internet of things with cloud integrated data management for smart cities." Environmental Technology & Innovation 24 (2021): 101802.

[12] Fang, Bingbing, et al. "Artificial intelligence for waste management in smart cities: a review." Environmental Chemistry Letters 21.4 (2023): 1959-1989.

[13] Mousavi, SeyyedKeyvan, et al. "Security of internet of things based on cryptographic algorithms: a survey." Wireless Networks 27.2 (2021): 1515-1555.

[14] Ullah, Amin, et al. "Smart cities: The role of Internet of Things and machine learning in realizing a data-centric smart environment." Complex & Intelligent Systems 10.1 (2024): 1607-1637.

[15] Islam, MdMilon, AshikurRahaman, and MdRashedul Islam. "Development of smart healthcare monitoring system in IoT environment." SN computer science 1 (2020): 1-11.

[16] Ullo, Silvia Liberata, and Ganesh Ram Sinha. "Advances in smart environment monitoring systems using IoT and sensors." Sensors 20.11 (2020): 3113.

[17] Shamala, L. Mary, et al. "Lightweight cryptography algorithms for internet of things enabled networks: An overview." Journal of Physics: Conference Series. Vol. 1717. No. 1. IOP Publishing, 2021.

[18] Almalki, Faris A., et al. "A low-cost platform for environmental smart farming monitoring system based on IoT and UAVs." Sustainability 13.11 (2021): 5908.

[19] Singh, Anuj Kumar, et al. "Secure and energy efficient data transmission model for WSN." Intelligent Automation & Soft Computing 27.3 (2021): 761-769.

[20] Al Rasyid, M. UdinHarun, M. HusniMubarrok, and JauariAkhmadNurHasim. "Implementation of environmental monitoring based on KAA IoT platform." Bulletin of Electrical Engineering and Informatics 9.6 (2020): 2578-2587.

[21] Arridha, Riyadh, et al. "Classification extension based on IoT-big data analytic for smart environment monitoring and analytic in real-time system." International Journal of Space-Based and Situated Computing 7.2 (2017): 82-93.

[22] Saheed, YakubKayode, and MichealOlaoluArowolo. "Efficient cyber attack detection on the internet of medical things-smart environment based on deep recurrent neural network and machine learning algorithms." IEEE Access 9 (2021): 161546-161554.

[23] Pandya, Sharnil, et al. "Federated learning for smart cities: A comprehensive survey." Sustainable Energy Technologies and Assessments 55 (2023): 102987.

[24] Salman, HayderMahmood, et al. "Smart Environment Network Design for Healthcare based on Artificial Intelligence." 2023 International Conference on Emerging Research in Computational Science (ICERCS). IEEE, 2023.