# AES Based End-to-End Encryption Scheme for Secure Communication on Internet of Things (IoT)

**[1]Umashankar Biswal, [2]Raurab Paul, [3]Saumendra Pattnaik, [4]Binod Kumar Pattanayak***

Department of Computer Science and Engineering, Institute of Technical Education and Research, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, Odisha, India
*: Corresponding Author

**Abstract—**Temperature and humidity are two factors that are crucial for monitoring tasks, whether they are being done indoors or outside. In this study, we design, construct, and demonstrate the functionality of a portable indoor sensor system that is linked to a smartphone-based user interface for humidity and temperature monitoring. An ESP8266 microprocessor, DHT11 temperature sensor, rechargeable battery, and a charger circuit are among the parts of this sensor system that are crammed into a single little box. Data are then shown on the user website after the sensor system and application have been evaluated to determine their performance. According to the test, the system can communicate with the smartphone via the host. It is also more secure because AES encryption and MD5 hashing are used, and only the user may see the data. There has been a lot of research done to find effective security solutions for IoT. In this research, an Advanced Encryption Standard (AES)-based solution to protect IoT systems is provided.

**Keywords—IoT, Security, AES, MD5**

## 1. Introduction

Since pandemic indoor activities occur more often. And maintaining air quality in any indoor area has been a major issue. It is said that the growth and evolution of the Internet are exponential. The Internet has grown over the past 25 years, connecting people worldwide via computers, laptops, cellphones, and other gadgets [1]. The highly varied characteristics of IoT in the current environment make it difficult to guarantee security and privacy. IoT's highly scalable and distributive characteristics need the creation of a flexible and novel security architecture to address these issues and support the functionality of IoT devices over the long term [2]. Temperature & humidity are 2 of the most prominent monitoring parameters in the Internet of Things. Temperature & humidity sensors are among the most commonly used environmental sensors. Typically, web, mobile, or application-based user interfaces are used for monitoring. Many applications of the previously indicated approach may be deemed unfeasible, particularly for large-scale production & integration into IoT networks. . It is possible to use Arduino or ESP boards, which are suitable for small-scale applications but problematic for widespread implementation due to speed and cost restrictions. In order to make the sensor system lightweight to implement and quick to install, compact and low-power features were taken into account in the design. It's also a good idea to merge the sensors as well as various other components, like the power, microcontroller, & communication, into a small package. To monitor the temperature and humidity we have made a monitoring device which will be a one to one user based and is secure to all users as we used AES and md5 algorithms in our model for maintaining integrity of the data and properly encrypted and decrypted so as it does not get to all users. The data is kept limited so that no one except the user can have a view. It can be used in our daily lives to monitor the

attributes and can also help us in different ways , i.e. if the temperature goes above a limit like say 50°C we can send a notification to the user that the house may be or is in danger of fire. To take early precautions this device can be helpful.

Rest of the paper has been organized as follows. Section 2 includes description of various IoT architectural aspects. Related work is elaborated in Section 3. Section 4 covers the proposed security model. Analysis of results is discussed in Section 5 and Section 6 concludes the paper with probable extensions to the current work.

## 2. IOT Architecture

With regards to the Internet of Things, the architecture is a structure that says the physical parts, the utilitarian association and design of the organization, functional techniques and the information arrangements to be utilized. There is not a simple blueprint which can be followed for all potential executions. IoT architecture can really change altogether relying upon the execution; it should be open enough with open protocols so it can uphold multiple network applications.

There are many type of architecture they are

1. 3 layered architecture
2. 4 layered architecture
3. 5 layered architecture

A. 3 layered architecture

A three-layer architecture is t1.1he normal and it is called structure. Basically it is implemented in the initial study of IoT. It demonstrates three levels: perception, network, and application.

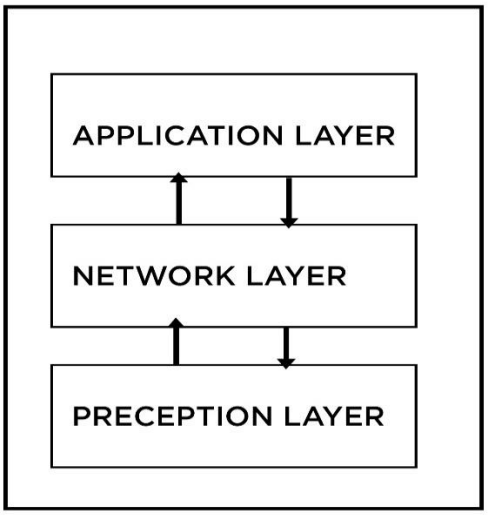


**Figure 1: 3 Layered Architecture**

Perception Layer

The IoT architecture contains the perception layer. The perception layer is made up of a variety of sensors for gathering environmental data and gateways allowing wireless devices to connect to the network [3]. These can be the edge device, sensors, and actuators that cooperate with their environment. It identifies specific spatial boundaries or other

things/objects in the environmental elements.

Network Layer

The information that devices use to gather should be communicated & processed. Main job of the network layer. It interfaces the particular gadgets to another smart object, network device and servers. This takes care of the data or information transmission.

Application Layer

The application layer is basically used for communication. It is responsible for furnishing the user with software resources . Example: by pressing a button in an application to switch on the coffee machine which is used in smart homes.

B. 4 layered architecture

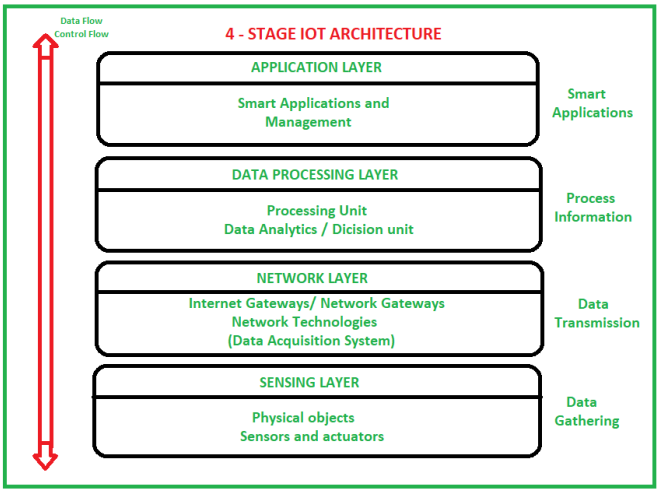


**Figure 2: 4 Layered Architecture**

Along these lines, from the above diagram obviously there is 4 layers are available that can be partitioned as follows:

a. Sensing Layer
b. Data processing Layer
c. Network Layer
d. Application Layer

Sensing Layer

In this sensing layer sensor, actuators and devices are available. Here the Actuators accept the data, process the data and discharge the data through the network.

Network layer

The major components of the network layer are the Data Acquisition System (DAS) and the network gateway. The Data Acquisition System primarily performs data aggregation & conversion activities (gathering & aggregating data & converting analogue data to digital data) (DAS). The network layer here conducts network and transport layer functions [4]. The connection between the sensor network and internet is opened up by Advance gateway which also performs many other basic gateway functionality.

Data processing Layer

This is a kind of processing unit for the IoT environment. Where the information / data is examined and pre-processed previously transferred to the data center from where the data is gained by programming applications regularly named as business applications where information is checked and managed and further activities are also prepared.

Application Layer

This is the IoT architecture's final layer. Cloud or the data center is the management stage of the data where the data can be managed which are basically used by the end- user.

C. 4 layered architecture

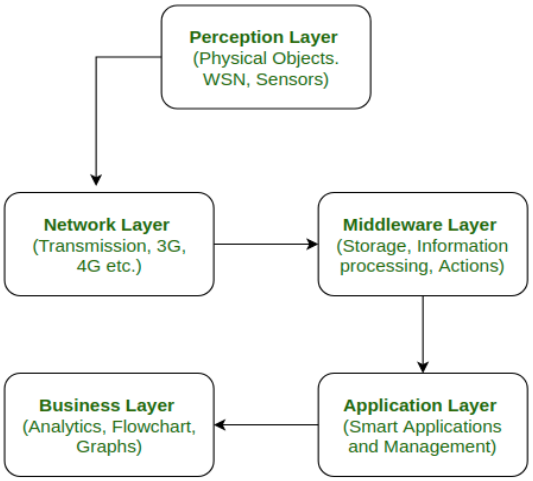


**Figure 3: 5 layered Architecture**

The Five layer design is ideally suited for employment in a variety of design work where cutting-edge technology and a large application area are required. This model is deemed as an expansion to the essential design of IoT on the grounds that it has two extra layers to the fundamental model.

Along these lines, from the above diagram obviously there is 5 layers are available that can be partitioned as follows

a. Precipitation layer
b. Network layer
c. Middleware layer
d. Application layer
e. Business layer

Precipitation layer

In the five Layer architecture it is the 1st layer. Here many actuators and sensors are utilized to assemble valuable data like temperature, dampness content, interloper identification, sounds, and so forth. To acquire data from surroundings & to pass information to another layer is the main objective of this layer then pass information to other layer so that a few activities should be possible.

Network layer

According to the name it is the associating layer among Precipitation layer and middleware layer. The data is provided by the Precipitation layer, then the data is processed and the processed data is sent to the middleware layer utilizing networking technology, for example 2G,4G, etc. The network layer is also termed a communication layer since it helps in communication between those layers. Those data were transferred safely having the confidentiality of the data.

Middleware layer

Middleware Layer has a few progressed highlights like stockpiling, calculation, handling, activity taking capacities. It stores all information collection and in light of the gadget address & name it gives proper information to that gadget. It could likewise take choices in light of estimations done on informational collection acquired from sensors.

Application layer

Basically this layer deals with all applications in view of data obtained from the middleware layer. This layer includes sending the message, device on and off, smartwatch, etc.

Business layer

The success of every product is determined not only by the technology it employs, but also by how it is represented to its clients. These responsibilities are handled by the device's business layer. Making flowcharts, charting, analyzing data, considering ways to improve the gadget, and other related tasks are part of it.

## 3. Related Work

To safeguard the data produced by IoT devices, numerous methods & techniques have been developed. The data produced by IoT devices is encrypted using a variety of modern security algorithms and techniques, which we have shown in the section below.

According to the reference paper [5] IoT Security utilizing AES Encryption Technology based ESP32 Platform.

An instance of the Internet of Things was demonstrated by linking a few sensors. The data is received by the card designed and developed by Espressif Systems (ESP8266) module, where it is encrypted before being sent to the internet site through an authorized person to be received from anywhere. It is also feasible to acquire it via a public IP address that is disclosed within the ESP32 device module's internal network. To discover the sensors' actual values, the decryption component is finally proposed.

The Internet of Things (IoT) component was represented by a variety of sensors, including the DH11 temperature and humidity sensor, ultrasonic sensor, and LED IoT Security. 217 lights) were connected to the ESP32 using AES Encryption Technology. The ESP32 chip receives the data, encrypts it, and then sends it to a special internet page that must be accessed securely using a username and password. The data is also broadcast on the IP address of the ESP32 chip within the very same service provider's local Internet network. Following receipt of the data, it could be decrypted using an AES program available on the Internet & as a mobile device application to identify the real values of the data received. As a result, if someone gains access to the informational page & attempts to steal the data, determining the real values will be difficult since they will need to be knowledgeable of the encryption technique and key used

in this approach. As a result, security can be strongly achieved.

The AES encryption technique was developed on the ESP32 platform, & the design was also applied to numerous sensors to symbolize the IoT component of a smart home or another application. This method was created to improve and reinforce the security of the Internet of Things. We settled on a sensor with two lights and sensors for temperature, humidity, and distance. Because of the ESP32 chip's ability to communicate with IoT and the efficacy of AES technology in encrypting and safeguarding data received or delivered, this architecture might be utilised to preserve and secure incoming information from the Internet of Things.

According to the reference paper [6] Analyzing the AES Encryption Resource Utilization on IoT Devices,

The Advanced Encryption Standard (AES) is investigated in this study in terms of length and energy consumption on two resource-constrained IoT edge devices utilizing both software & hardware with variable key and buffer size settings. We particularly note that: (1) when compared to software, hardware execution is more sensitive to buffer size settings and only uses less time and energy overall whenever the buffer length is sufficiently large; (2) an uptick in key size results in increased consumption of resources in all cases; and (3) when comparing the two IoT boards, the CYW board uses less resource because it is built with a faster fallback CPU clock rate and more memory. These findings not only provide insight on the future development of lightweight security designs but also help expand our understanding of the trade-off between the security requirements & resource consumption of IoT devices.

In this study, three distinct AES implementations on two different IoT boards are thoroughly analyzed in terms of energy consumption and encryption duration for all key sizes and buffer sizes that are supported. Algorithms for security must share resources with other computational tasks or I/O, therefore it is essential that they are effective in order to reduce the overall cost of security and be utilized in conjunction with other capabilities.

According to the reference paper [7] VLSI IMPLEMENTATION OF CRYPTOGRAPHIC ALGORITHMS IN INTERNET OF THINGS ,

In this paper a cryptographic method is put forth which makes use of MD5 and AES algorithm to attain security & privacy. Both these algorithms are simulated in Modelsim 6.5 & Xilinx 14.2 tools using verilog HDL. Proposed method of chaining the two algorithms twice provides better security and privacy. On integrating these algorithms into an RFID tag, a secure means of communication can exist between surrounding things and thus making way for the acceptance of Internet of Things in society.

An efficient method of cryptographic chaining approach has been introduced in this paper. A secure means of data transfer can take place in IoT by integrating the chained cryptographic algorithms into the tag. The code is simulated in verilog and the efficiency is verified.

According to the reference paper [8] Secured IoT Through Hashing Using MD5 ,

When an embedded device is affiliated to the internet and made to function as a part of the Internet of Things(IOT), the system is recommended to allow the use of MD5 to secure it from attacks. The Internet of Things and embedded systems have a preface. We've talked about the

different requirements that a security algorithm must meet. This served as the foundation for testing the proposed system's compatibility and fulfillment of the necessary requirements, which led to the successful use of MD5 in an embedded system for IoT security.

The suggested system may be employed in a range of internet-connected embedded gadgets to enable safe communication & data exchange with the network's other internet-connected devices. This may be discovered by examining the outcomes and comparing them to the results of the other methods of communication security. The transmitter can be thought of as the sensor network and the receiver as the HUB in the context of a WSN (Wireless Sensor Network), which consists of numerous sensors communicating with a single HUB. In this case, the resource-constrained sender can utilize the proposed paradigm to send any form of information over the network, not only the 2 control actions, with no increase in complexity. The receiving end, or HUB, will only be affected by the expansion of the number of possible combinations (N). The development of the hash collection for big N values will not be a problem either because the HUB does not have as many restrictions as the sender, allowing for quick and secure connection regardless of the data being transferred.

According to the reference paper [9] AES And MD5 Based Secure Authentication In Cloud Computing ,

In this study, we suggest a current method of data encryption and decryption at the time of login, however authentication is not offered at that time. Security is not offered because it is only based on the foundation of trust.

One of the fastest-growing areas of research is cloud computing, and security is one of the most pressing issues because as the amount of data stored in the cloud grows, so do the potential risks. In our proposed work, we first secure user login by encrypting login id and password in our database to save logins from man-in-the-middle attacks. However, if an attacker learns your login password, they will be unable to confirm the login even without the user's thumbprint, thus we additionally enable user authentication & thumb image transfer over the network in encoded form using MD5.

One of the most fascinating areas of study where a lot of work is done in this area is cloud computing. We use cryptography methods and authentication to provide security in our work. Future operating systems will be secure, preventing cloud computing exploits from being easily portrayed.

According to the reference paper [10] A Review of Data Security and Cryptographic Techniques in IoT based devices
This paper offers an overview of significant lightweight cryptography methods applied to Internet of Things (IoT) devices. The analysis of the literature review reveals that utilizing cryptography alone to safeguard the data produced by IoT devices does not perform as well as using cryptography and steganography together.

The comparison of IoT's light-weight security algorithms is shown in Table 1. The comparison demonstrates that we could achieve Data Integrity when we just utilize cryptographic approaches to encrypt the data produced by IoT devices. It also demonstrates how we can

secure the data produced by IoT devices by utilizing a combination of cryptography & steganography techniques to ensure Confidently, Data Integrity, and Authentication. Data can be concealed via steganography in an image, carrier file, video & audio file, so that data created by IoT devices cannot be detected while traveling over the internet [11].

Researchers have recently become interested in the IoT, a new growing field. And during the past few years, internet usage has also grown significantly. Data security presents a significant barrier to computer users. The two main methods for data security are cryptography and steganography. Steganography is used for communication security, while cryptography is used for data security/encryption [12]. Integrity (no one can alter or modify the data) is provided via cryptography, and confidentiality is provided by steganography (no one can sense or access the data). According to the research review, using both Steganography and cryptography together increases security. Combining cryptography and steganography strengthens each other's flaws. Increasing security and making it more difficult for hackers to access or steal data.

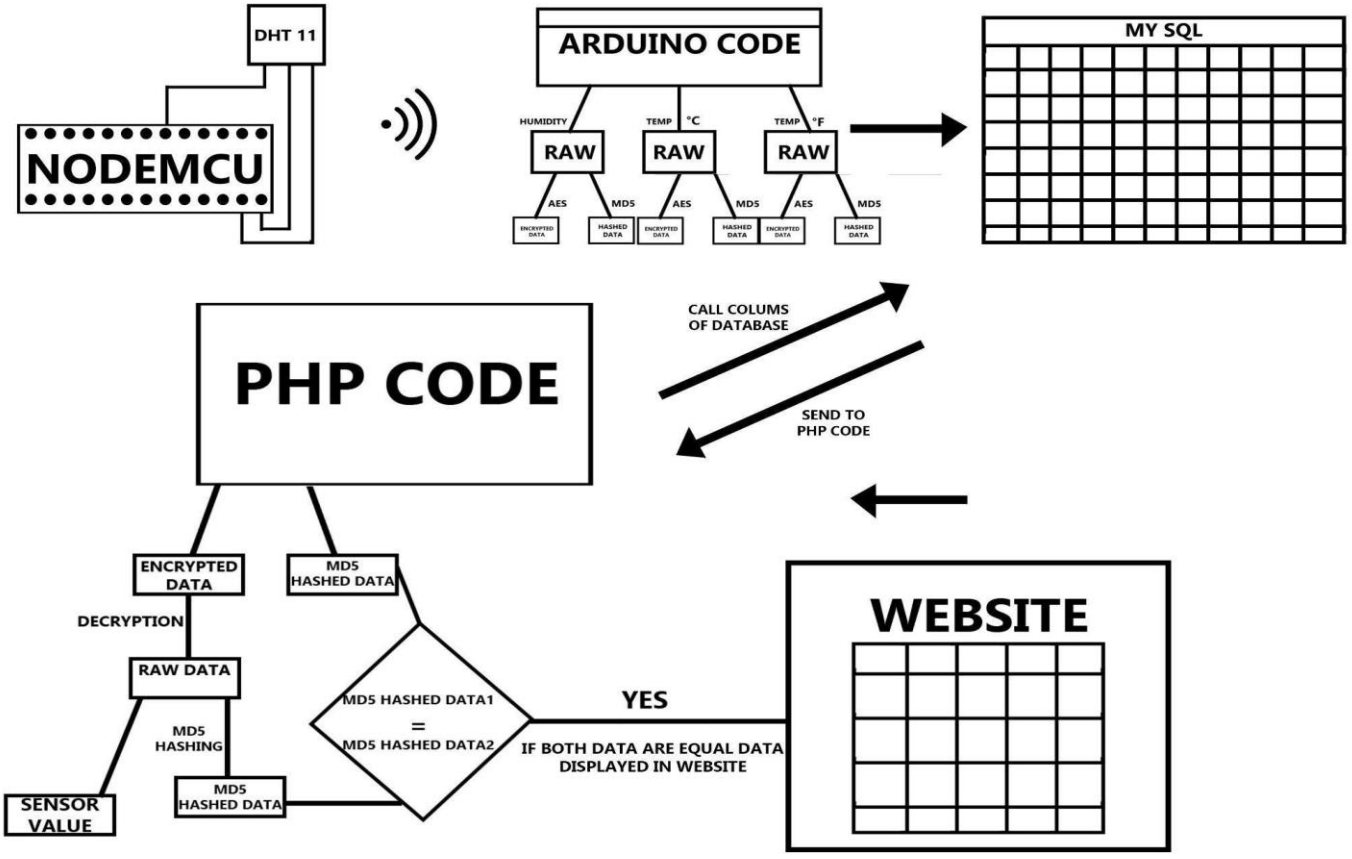


**Figure 4: Proposed Model for IoT System**

## 4. PROPOSED MODEL

### 4.1 Hardware

For the NodeMCU open source firmware, open source prototype board designs are available. It is a Wi-Fi capable gadget that utilizes the esp-8266 Wi-Fi SoC. It features built-in Wi-Fi, which lowers the cost and facilitates remote access; once input is provided, it can also function without internet access. A humidity and temperature sensor called the DHT11 that produces

digital output can be incorporated into microcontrollers.

It measures the air around it and produces a digital signal on the data pin using a thermistor and a capacitive humidity sensor. It uses a DHT11 library in the Arduino IDE. The sensor contains 3 pins – Vcc, Gnd and data. In our implementation we have connected the Gnd to Gnd, Vcc to 3V3, data to D2 from DHT11 Sensor to nodeMCU respectively.

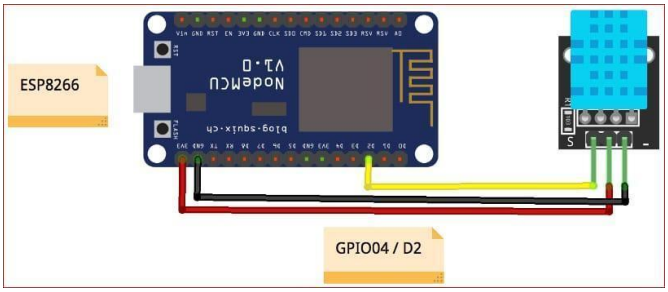


**Figure 5: hardware pin connection**

## 4.2 Arduino Code

The data from the sensor is received by arduino code from the sensor. It gives 3 values: Humidity, Temperature in Celsius and temperature in Fahrenheit.
We haveused the AES algorithm for encryption and md5 with hashing to check the integrity. The data is sent from Arduinoto the mysql database after encryption and md5 hashing and then stored in 6 different columnsin the database

### 4.2.1 AES Encryption

The National Institute of Standards & Technology (NIST) of the U. S. developed the Advanced Encryption Standard (AES) in 2001 as a standard for electronic data encryption. AES is extensively used nowadays owing to its significantly better strength than DES & triple DES, although it is more difficult to construct. Instead of using a Feistel cipher, AES uses iteration. It is based on substitution & permutation networks, which are two popular approaches for encrypting & decrypting data (SPN). Block cipher algorithms perform a number of mathematical operations called SPN. AES has a constant plaintext block size of 128 bits (16 bytes). AES employs a byte matrix, which is represented as just a 4x4 matrix.

Additionally, the number of rounds in AES is a critical component. The length of the key determines how many rounds there will be. The AES technique uses 3 different key sizes to decrypt & encrypt data, including (128, 192 as well as 256 bits).

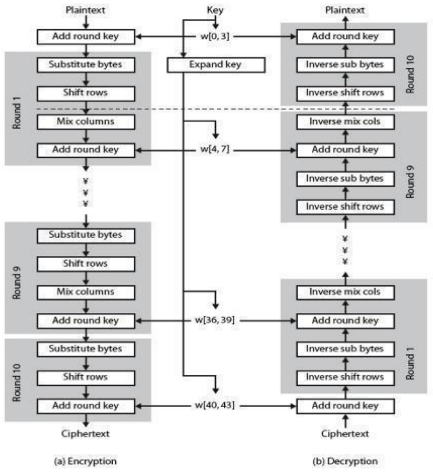


**Figure 6: Basic Structure of AES**

Security of AES

Security was one of the most important factors taken while choosing an algorithm by NIST. The primary causes of this were evident given that one of the key goals of AES was to enhance the security flaw of the DES algorithm. When compared to other suggested algorithms, AES has the best ability to keep hackers away from sensitive data and prevents them from decrypting it. This was accomplished by extensively testing AES against both theoretical and real-world attacks.

- AES is indeed a block cipher.
- The key size might be 128/192/256 bits.
- Data is encrypted in 128-bit chunks.

Working of AES

There are three block ciphers in AES:

1. AES-128 encrypts & decrypts a block of messages with a 128-bit key length.
2. AES-192 encrypts & decrypts a block of messages using a 192-bit key length.
3. AES-256 encrypts & decrypts a block of messages using a 256-bit key length.

Each code uses 128, 192, or 256-bit cryptographic keys to scramble & decode information in blocks of 128 pieces. Secret key figures, also known as symmetric figures, use the same key for both encoding and decoding. There must be a similar mystery key that both the source and the recipient are aware of and can use. The government categorizes data as secret, confidential, or top secret. Any key length can be used to secure the Confidential & Secret levels. For extremely confidential data, cycle key lengths of 192 or 256 are necessary. There are ten rounds for 128-bit keys, twelve rounds for 192-digit keys, and fourteen rounds for 256-digit keys. A round is made up of multiple handling procedures, such as substituting, interpreting, and combining plaintext to generate the final product of ciphertext.

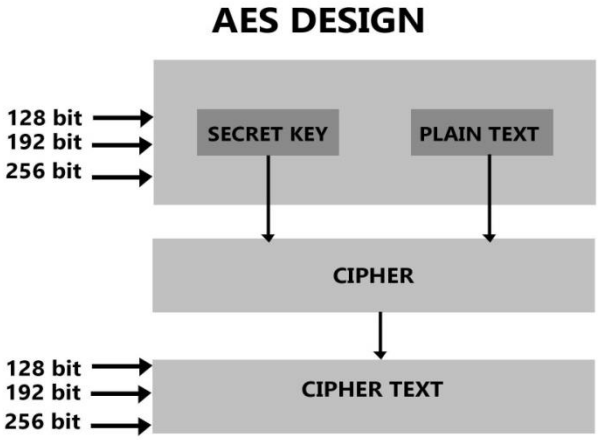


**Figure 7: AES algorithm design**

Encryption

The AES encoding calculation defines numerous modifications to be done on data stored in just a cluster. The 1st stage in the code is to insert the data into an exhibit, following that the code modifications are rehashed across many encryption cycles. The first modification to the AES encoding figure is the use of an alternate table to replace information. The consequent modification moves information lines. The 3rd combines portions. A distinct component of the encryption key is used to accomplish the continuation change on each segment. Longer keys need more adjustments to finish.

For a 128-bit block, you do the following AES encryption steps:

1. Derive the set of round keys from the cipher key.
2. Initialize the state array with the block data (plaintext).
3. Add the initial round key to the starting state array.
4. Perform nine rounds of state manipulation.
5. Perform the tenth and final round of state manipulation.
6. Copy the final state array out as the encrypted data (ciphertext).

Decryption

The steps in the rounds are simple to reverse because they each contain an opposite that, when used, undoes the modifications. Depending on the key size, each of the 128 blocks is processed via 10 , 12, or 14 rounds. The stages of each round in decryption is as follows :

- Inverse MixColumns
- Add round key
- ShiftRows
- Inverse SubByte

The decryption method is the reverse of the encryption procedure, thus I'll explain the processes with notable distinctions.

Summary
We have implemented AES-128 in our project for better security of the data. And so the three realtime data we get from the DHT11 sensor is encrypted and sent to the database.

### 4.2.2 MD5 Hashing

In order to safely connect the embedded device to the internet, a number of security requirements must be met. A message digest of a fixed length is produced using the hashing method MD5 from an input of any length.

Hash function
A hash capability is a one-way encryption capability that takes a variable-size input plaintext m and produces a fixed-size hash yield. It is computationally difficult to unravel the hash and any endeavor to break it is basically infeasible. A "secure" hash capability ought to have the option to oppose pre-picture assaults and impact assaults. Because of the categorized standard and birthday mystery, there will be a few data sources that will deliver a similar hash result. The result length is of fixed size 128 pieces, making a sum of 2128 potential result hash values. This worth, as large as it might appear, isn't boundless, subsequently bringing about impacts.

Md5 algorithm
Ron Rivest developed the MD5 (Message Digest Algorithm) in 1991. A variable-length message is converted by MD5 into a result with a fixed length of 128 pieces. A well-known hashing capability is MD5. It chips away at blocks of 512-pieces, and cycles each block through 4 rounds, where each round in go cycles 16 sub-hinders (each 32-bits). The 512-cyclemessage is separated into 16 sub-blocks prior to handling. On the off chance that a message block doesn't depend on 512-bits, it is padded as displayed.

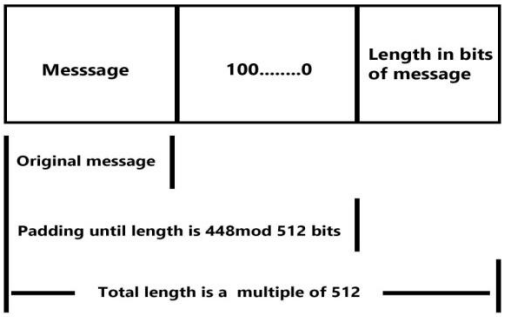


**Figure 8: DECRYPTION DESIGN**

Use of Md5 hashing
The data set's plaintext storage of passwords is incredibly unstable. Instead of storing the plaintext passwords in the data set, MD5 computations can be used to hash the initial passwords and the hash values to increase the security of passwords. During validation, the info secret key is likewise hashed by MD5 along these lines, and the outcome hash esteem is contrasted and the hash esteem in the data set for that specific client.

Example

Alisha reviewed the pair with Bob and said something particular, as we should have expected. Bob uses the cryptographic hash capabilities on the received message to verify its veracity and receives a second opinion. Presently, Bob will look at the new overview and the review sent by Alisha. On the off chance that both are the same Bob is certain that the first message isn't changed.

This message and review pair is identical to an actual report and unique finger impression of an individual on that record. Not at all like the actual archive and the unique finger impression, the message and the overview can be sent independently. Above all, the overview ought to be unaltered during the transmission.

The cryptographic hash capability is a one-way capacity, which means it is nearly impossible to rearrange. This cryptographic hash function takes a variable-length message as input and generates a hash/unique/summary mark of set length, which is used to verify the message's integrity.

The message digest ensures the accuracy of the report. To increase the message's authenticity, the digest is encoded using the source's secret key. This overview is now known as advanced signature, and it can be simply unscrambled by the receiver who possesses the source's public key. The beneficiary may now validate the source & verify the reliability of the communication.

Summary
The integrity of messages is frequently verified using the MD5 hashing technique. MD5 creates a 128 bit digest by dividing the message into 512 bit chunks (typically, 32 Hexadecimal digits). After getting the value from the sensor we hashed it using the md5 algorithm so as to check the integrity of the whole process as display the result checking it.

**4.3 Connection to database**
phpmyadmin
PhpMyAdmin is one of the most famous applications for MySQL information base administration. It is a free PHP-based gadget. This tool allows you to create, modify, delete, import, and export MySQL information base tables. You can conduct MySQL queries, upgrade, repair, and examine tables, modify resemblance, phpMyAdmin programming which is coordinated in cPanel.

My-sql Database
By itself, MySQL is a pretty potent programme. It manages a sizable portion of the most pricey and potent database products' features. A standardized version of the well-known SQL data language is used by MySQL.

Summary

After getting the values from the DHT11 sensor , we encrypted the data and also hashed the data using md5 hashing. Then we send all the six values and a timestamp to the database and

store it. The datatype of the columns are based on their values.

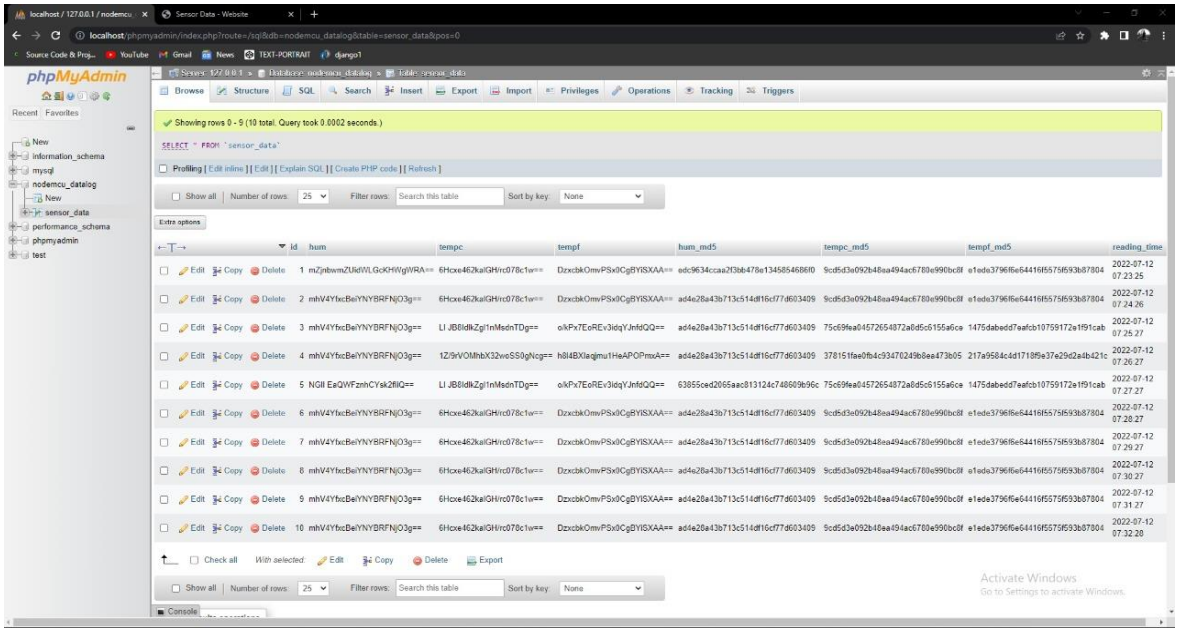


**Figure 9: Mysql database**

## 4.4 Fetching the data

php code
In order to create dynamic and interactive websites, PHP (Hypertext Preprocessor), a general-purpose scripting language, can be used. Because it was one of the first server-side languages that could be included into HTML, it made it easier to add functionality to web sites without having to refer to other files.

Receive data
By using the PHP function mysql_query, the SQL SELECT statement can be used to retrieve data from MySQL tables. To retrieve data from MySQL, you have a variety of options. The most popular choice is to utilize the function mysql_fetch_array ().This function can return a row as an associative array, a numerical array, or both. If there are no more rows, this method returns FALSE.

Show data
Now that we have fetched the data from the database we create a new php file and then add stylings as per our requirements, for now we have used a table for sequential data. Then we check if $fetchData is an array or not with if & else condition then if so apply for each loop to fetch data. After the data is fetched from all the columns it will now be displayed in the table.

Summary
The php code in our code is the connection code between the mysql database and the website .It gets the data from the database then it decrypts the data and converts it to md5 hashing .then it compares the previously stored md5 data to the newly generated one for integrity. If it

is equal it displays it in the website otherwise not. The html webpage created is inside the php code,we can change the styling from there.

### 4.5 Website

After the data is decrypted and the integrity of the data is matched using md5 hashing. If the integrity remains to be equal then the decrypted data is displayed to the website otherwise not. We have created a webpage for the values to be systematically displayed in the form of a table with proper timestamps.


| ID | Date - Time | Humidity % | Temperature °C | Temperature °F |
|---|---|---|---|---|
| 10 | 2022-07-12 07:32:28 | 88.00 | 30.80 | 87.44 |
| 9 | 2022-07-12 07:31:27 | 88.00 | 30.80 | 87.44 |
| 8 | 2022-07-12 07:30:27 | 88.00 | 30.80 | 87.44 |
| 7 | 2022-07-12 07:29:27 | 88.00 | 30.80 | 87.44 |
| 6 | 2022-07-12 07:28:27 | 88.00 | 30.80 | 87.44 |
| 5 | 2022-07-12 07:27:27 | 89.00 | 30.20 | 86.36 |
| 4 | 2022-07-12 07:26:27 | 88.00 | 30.30 | 86.54 |
| 3 | 2022-07-12 07:25:27 | 88.00 | 30.20 | 86.36 |
| 2 | 2022-07-12 07:24:26 | 88.00 | 30.80 | 87.44 |
| 1 | 2022-07-12 07:23:25 | 90.00 | 30.80 | 87.44 |

**Figure 10: Website snapshot**

## 5 ANALYSIS OF PROPOSED MODEL

In this paper, we suggested The Internet of Things was demonstrated by the linking of a few sensors. The card receives the data. Espressif Systems (ESP8266) module created & developed it, & it may be received through a public IP address broadcast inside the internal network of ESP32 device module. It is then encrypted and sent to the website through an authorized person to be received from anywhere. The decryption component is finally suggested to determine the sensors' actual values. The procedure is cyclical. Prior to being transferred to a MySQL database, the sensor values are first encrypted and hashed using MD5. After the encrypted data is decrypted and returned back to the raw value. Then the raw value is hashed using the md5 algorithm and then compared with the previous hashed value for checking the integrity of the data. Then if the integrity returns to be equal the raw value of the respective cell is displayed over to the website otherwise it is shown null. This device can be helpful to us in day to day lives and as well it is more secure for the end users too.

## 6 CONCLUSION AND FUTURE WORK

For contributors that upload sensing data, we suggested an end-to-end security solution. This technique enables end-to-end data encryption to safeguard data in transit. All IoT system restrictions are taken into account in the suggested middleware solution. The process is a cyclic process. The sensor readings are first encrypted and hashed using md5 and then sent to the MySQL database. After the encrypted data are decrypted and returned back to the raw value. Then the raw value is hashed using the md5 algorithm and then compared with the previous hashed value for checking the integrity of the data. Thenif the integrity returns to be equal to the raw value of the respective cell, it is displayed over to the website, otherwise it's shown as null. This device can be helpful to us in day to day lives and as well it is more secure for the end users too.By implementing this project we can make this as a smart home product for real day to day life temperature and humidity monitoring devices to get quick updates. We can add on some new features like weather forecasting, fire alarm and authentication to make it more secure for all the users. We can also make different devices withthe help of this secured algorithm model

## REFERENCES

[1] D. D. Ramlowat.,Binod, Kumar, P., 2019. "Exploring the Internet of Things (IoT) in Education: A Review," Springer Nature Singapore

Pte Ltd., pp. 244-254.

[2] Mohammad, Reza, H., &Binod, Kumar, P., 2020. "Security Issues in Internet of Things (IoT): A Comprehensive Review," Springer

Nature Singapore Pte Ltd, pp. 354-364.

[3] Mohammad Reza H., Binod, Kumar, P., 2021, "An End-to-End AES Based Cryptographic Authentication Mechanism for Communication

on Internet of Things (IoT) Using MQTT" Nat.Volatiles&Essent. Oils.

[4] Reza H., and Binod, Kumar, P., and S. Wedig, 2019. "A Secured Communication Model for IoT," Springer Nature Singapore Pte Ltd,

pp.190-196.

[5] Mohammad, A., & Mohamed S., March 2022. IoT Security Using AES Encryption Technology based ESP32 Platform. The International

Arab Journal of Information Technology, Vol. 19, No. 2,

[6] Pedro, S., Nam, T., Brandon, C., Behnam, D., &Yuhong, L., 12-15 November 2018. Analyzing the Resource Utilization of AES

Encryption on IoT Devices.Proceedings, APSIPA Annual Summit and Conference.

[7] J.GOPIKA, R., ASWATHY, N., 05th April-2014. VLSI Implementation of Cryptographic Algorithms in the Internet of Things. In

Consumer Electronics (ICCE), 2018 IEEE International Conference on (pp. 1-5). IEEE.

[8] Irfan, A. L., &Hannan, S., (2018). "4th International Conference on Advances in Electrical, Electronics, Information, Communication, &

Bioinformatics (AEEICB-18)".

[9] Shefali, o., and Prof.Vikram, r. International conference on I-SMAC "AES And MD5 Based Secure Authentication In Cloud Computing"

(IoT in Social, Mobile, Analytics and Cloud).

[10] M. Ghulam, A. Rehan, M. Muhammad, J. Abid, and M. Muhammad, "A Review of Data Security & Cryptographic Techniques in IoT-

based Devices." 2018.

[11] Rath M. and Pattanayak B. K., Technological Improvement in Modern Health Care Applications Using Internet of Things (IoT) and Proposal of Novel Health Care Approach, International Journal of Human Rights in Heathcare, Vol.12, No.2, pp.148-162, 2019.

[12] Rath M., Pati B. and Pattanayak B. K., Relevance of Soft Computing Techniques in the Significant Management of Wireless Sensor Networks, Soft Computing in Wireless Sensor Networks, pp.75-94, 2018.