INJ INTERNATIONAL NEUROUROLOGY JOURNAL

Implementation of Authorization and Authentication techniques in IoT objects for Industrial Applications

Sandeep. K. V. [1] **& Dr. T. C. Manjunath**[2]

[1] Research Scholar, VTU Research Centre, Dept. of Electronics & Communication Engineering,Dayananda Sagar College of Engineering, Bengaluru-560111, Karnataka
Visvesvaraya Technological University, Belagavi-590018, Karnataka
&
Assistant Professor, Electronics & Telecommunication Engineering, Dayananda Sagar College of Engineering, Bengaluru-560111, Karnataka

[2] Research Supervisor, Professor & HOD, Dept. of Electronics & Communication Engineering, Dayananda Sagar College of Engineering, Bengaluru-560111, Karnataka
Visvesvaraya Technological University, Belagavi-590018

Abstract

The usage of internet enabled gadgets has significantly increased in recent years in both the industrial and consumer sectors. IoT, or the IoT, is a network of interconnected "things" offering remarkable method for social and economic growth but at the expense of delivering a startling quantity of potentially sensitive data via unreliable networks. For avoiding, or at least minimize, the security and privacy risks w.r.t IoT architecture, we have examined the security limits of this emerging technology. In this paper, authentication and authorization in IoT devices are presented primarily for industries where parameters like humidity, temperature, and gas/smoke leak can be detected and monitored in real-time. Data can be accessed at any time and place by using a laptop, computer, or even a mobile device using the existing application. In this proposed work, we categorize and propose user authentication and authorization schemes in industrial applications for IoT environments. This proposed system mainly has applications in industries where parameters like humidity, temperature, and leakage of any gases or smoke can be monitored 24x7. In this work, designing & implementing of IoT systems can be utilized for industrial applications are proposed along with the experimental results using the concepts of data authentication process and validation of users by username and password and also Base64 algorithm is implemented for data encryption & decryption process.

Keywords: Security, Internet of Things, Python, humidity, temperature, decryption, encryption, Base64 algorithm.

Introduction

Cyber-attacks are a critical threat to organizations today. As more people work remotely and cloud computing becomes the norm across industries, the threat landscape has expanded exponentially in recent years. As a result, 94% of enterprise organizations have experienced a data breach—and 79% were breached in the last two years, according to a recent study by the Identity Defined Security Alliance (IDSA).

Additionally, research by Cyber security Insiders found that 90% of survey respondents experienced phishing attacks in 2020, and another 29% experienced credential stuffing and brute force attacks—resulting in significant helpdesk costs from password resets. With global cybercrime costs expected to grow by 15% per year over the next five years, reaching $10.5 trillion USD annually by 2025, it's important to organizations to protect themselves.

As a result, authentication has become an increasingly important mitigation strategy to reduce risk and protect sensitive data. Authentication helps organizations and users protect their data and systems from bad actors seeking to gain access and steal (or exploit) private information. These systems can include computer systems, networks, devices, websites, databases, and other applications and services. The suggested approach discussed here provides a solution to security problems in the domains of IoT device authorization and authentication.

(A) Authentication

Prior to granting access to a system or resource, authentication is the process of confirming a user or device. Its a procedure for confirming a user's or piece of information's identification. Confirming a person's identity when they connect into a computer system is known as user authentication. This assures that only user's with authorized credentials gain access to secure systems. When a user attempts to access the info on a network, they

DOI: 10.5123/inj.2023.4.inj52

must provide secret credentials to prove their identity. Upon authentication, can be granted permission to the user with confidence. Authentication is part of a three-step process for gaining access to digital resources:
(i) Identification- who are you?
(ii) Authentication-Prove it.
(iii) Authorization — Do you have permission?
Identification requires a user ID like a username. But without identity authentication, there's no way to know if that username actually belongs to them. That's where authentication comes in—pairing the username with a password or other verifying credentials. In this proposed method, authentication is a done by verifying login username and password.

(B) Authorization

A server assesses if a client has permission to utilize a resource or access a file through the authorization procedure. Authorization with authentication are frequently combined so as to server can identify the client making the access request. Different types of authentication may be required for authorization; some may not require passwords. In some circumstances, authorization is not required; any user may access a resource or a file by requesting it. Most Internet web sites don't need authorization or authentication. Once a person or process is authenticated, they're often put through an authorization procedure to assess if they should be granted permission to a certain protected resource or system. If fails in user authorization, not permitted to access data.
Lot of confusion between the words authorization and authentication. Before granting access to secure networks and systems, an authenticated user or process's identity is verified. Authorization, a more detailed procedure, verifies that the authenticated user or process has been given authorization to access the requested resource. Access control describes the procedure used to limit particular users' access to such resources. Always, the authorization procedure happens after the authentication step.

Literature Review

A novel authenticating procedure that provides two distinct kinds of primitives for authentication was developed by Kumar, Hemanta, and others [1]. BAN logic was used to show how secure the authentication mechanism was. Individual user, communications security assurance, and password error detections were all provided by the system. Secure channel technique necessitates the transmission of some security primitives. Through a centralised control server, authentication data is sent. The study identifies a few potential authentication techniques that might be applied to lessen threats and keep data secure.
A token-based safety for IOT objects is proposed by Muhammad Naveed, Sachin, et. al. [2]. Every device must authenticate itself with an authorisation server before it can connect to a server on another IoT device. The authorisation device grants the permission together with a security context that specifies the session's duration and scope. Therefore, two phases are examined here, the primary of which acquires security procedure from an authorisation server and the secondary of which enters a service.
An improved IoT device authentication mechanism has been offered by Noquia Fatima Tareen, et.al. [3]. The technique of authentication's efficacy is essential. The most important component is multi-factor authentication, which puts less stress on the device while yet being efficient and secure.
Farhad Seyed Sima Arasteh and colleagues [4] The client's distinct finger impression, their ID, and a undisclosed phrase are used as three verification elements in [18]'s secure confirmed key trade protocol. This technique utilises a bio-hashing function to transform each client's unique fingerprint into an outstanding format. The steps that make up the recommended method are client registration, login, validati0n, & session key agreement.
A completely new, lightweight authentication method for IoT has been suggested by Jun-Ya Lee et al. [5]. This uses XOR manipulation instead of complex encryption using a hashing to preserve anonymity.
For IoT-based e-Health applications, Zaman et al. [6] suggests a novel method that is both safe and portable. To enable secure communication b/w the IoT devices and the e-Health application, the method combines symmetric and asymmetric encryption. Simulations are used to assess the suggested scheme and shows how successful it is w.r.t efficiency and security.

BASE64 Algorithm

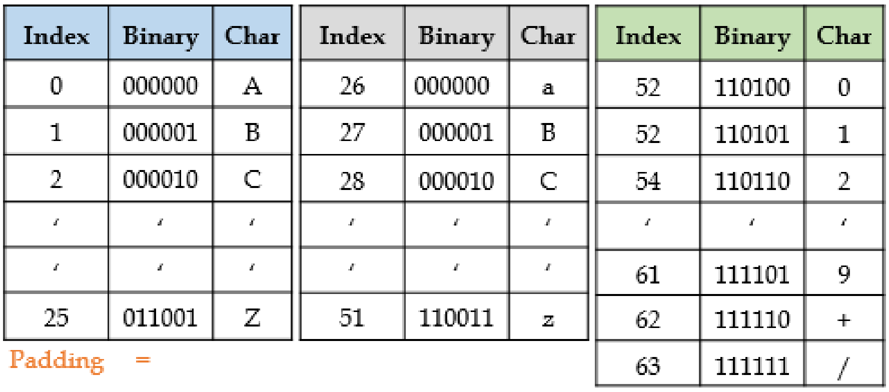
Data transformation known as encryption includes making data unintelligible to anybody who lacks a decryption key. A collection of binary-to-text encoding techniques called Base64 encode binary data into 24-bit sequences that may presented by 4 six-bit of Base64 digits. Base-64 is the collection of related binary-to-text encoding techniques that convert binary data in representation of a radix-64 and then encode it as an ASCII string. When binary data has to be encoded before being saved or transmitted across media that can handle ASCII interchange, Base64 encoding techniques are frequently utilised. This will guarantee that the data gets transported without being altered in any way. Base64 is often used in a variety of contexts, like email via the Multi-purpose Internet Mail Extension (MIME) and storing complicated data in XMLBase64 encoding is frequently used on the web to encode binary data so that it may be included to a data: URL. Base-64 is a standard method for converting binary data to text and is then transferred data between channels that can only reliably handle text content.

Data of Six-bits may be encoded using Base64, which has 64 possible values for each digit. Different implementations use different characters representing the 64 values. The typical approach is to select 64 printable characters that are similar to most encodings. This combination makes it difficult for the data to be altered while being transmitted across 8-bit-unclean information channels like email. For the first 62 values, A-Z, a-z, and 0-9 are used in MIME's Base64 implementation.

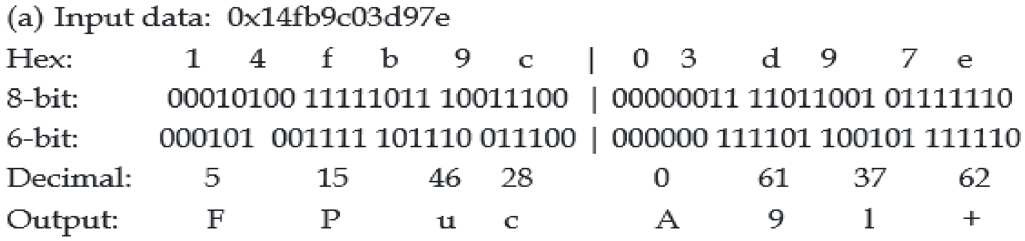
The Base 64 encoding is intended to represent arbitrary sets of octets in a way that supports both upper- and lowercase letters but is not required to be legible by humans. The table of Base64 information described in RFC 4648 is shown in Table 1 below.

Also in the below section, fig. 1(a) and 1(b) presents the few examples and operation of Base64 algorithm are presented to understand encryption & shows how an Input data is encrypted with algorithm.

Table. 1: Base64 data

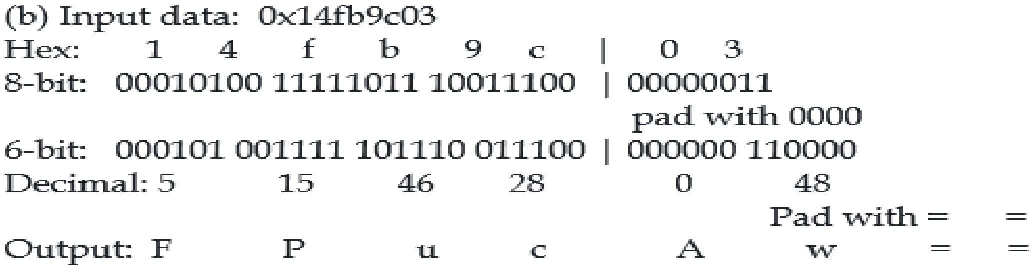


Index	Binary	Char	Index	Binary	Char	Index	Binary	Char
0	000000	A	26	000000	a	52	110100	0
1	000001	B	27	000001	B	52	110101	1
2	000010	C	28	000010	C	54	110110	2
'	'	'	'	'	'	'	'	'
'	'	'	'	'	'	61	111101	9
25	011001	Z	51	110011	z	62	111110	+
Padding	=					63	111111	/



```
(a) Input data:  0x14fb9c03d97e
Hex:           1    4    f    b    9    c   |  0   3     d    9     7    e
8-bit:        00010100 11111011 10011100  | 00000011 11011001 01111110
6-bit:       000101  001111 101110 011100 | 000000 111101 100101 111110
Decimal:        5         15        46    28         0        61       37        62
Output:         F          P         u     c         A         9        1         +
```

Fig. 1(a): Example 1 of Base64 algorithm



```
(b) Input data:  0x14fb9c03
Hex:       1     4     f      b      9     c    |    0    3
8-bit:  00010100 11111011 10011100  | 00000011
                                       pad with 0000
6-bit:  000101 001111 101110 011100 | 000000 110000
Decimal: 5          15         46        28          0        48
                                                     Pad with =      =
Output:  F          P          u         c           A        w          =       =
```

Fig. 1(b): Example 2 of Base64 algorithm

Test Vectors: Few test vectors of Base64 algorithm is shown below:

```
BASE64("") = ""
BASE64("f") = "Zg=="
BASE64("fo") = "Zm8="
BASE64("foo") = "Zm9v"
BASE64("foob") = "Zm9vYg=="
BASE64("fooba") = "Zm9vYmE="
BASE64("foobar") = "Zm9vYmFy"
```

Hardware Requirements

Hardware connection and setting up of this model as seen from the above figure 2. Hardware components list for the proposed work is shown in the below table 2. In the below table, raspberry-pi model 3B+ for connecting IoT devices or sensors, DHT11 (digital temperature and humidity) sensor, MQ-2 gas sensor for detecting leakage of gas or smoke or even it can detect and fire in industry. Jumper wires are used to connect all the 3 components.

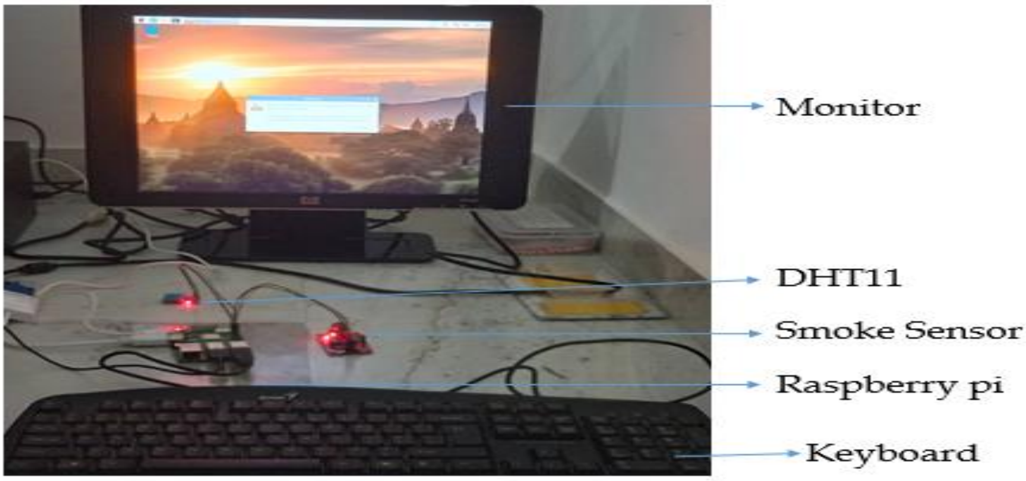
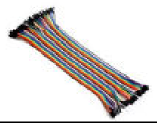


Fig. 2: Hardware setup

Table 2: Hardware components

Sl No.	Component	
1		Raspberry-pi model 3B+
2		DHT11
3		MQ-2 Gas sensor
4		Jumper Wires

System Architecture

Fig. 3 below shows the system architecture of the proposed model for the implementing authenticity & authorization in the IoT. As discussed in the above sections, the proposed architecture is proposed for industrial applications for monitoring some crucial parameters leakage of gas / smoke and even this smoke sensor can also acts like a fire alarm system. Only an authenticated person needs to have access to this IoT sensor data since it is crucial for subsequent processing. To prove identity of a user, a unique username and password check is done before giving access to IoT sensor data. The proposed diagram is shown the fig 4 below.

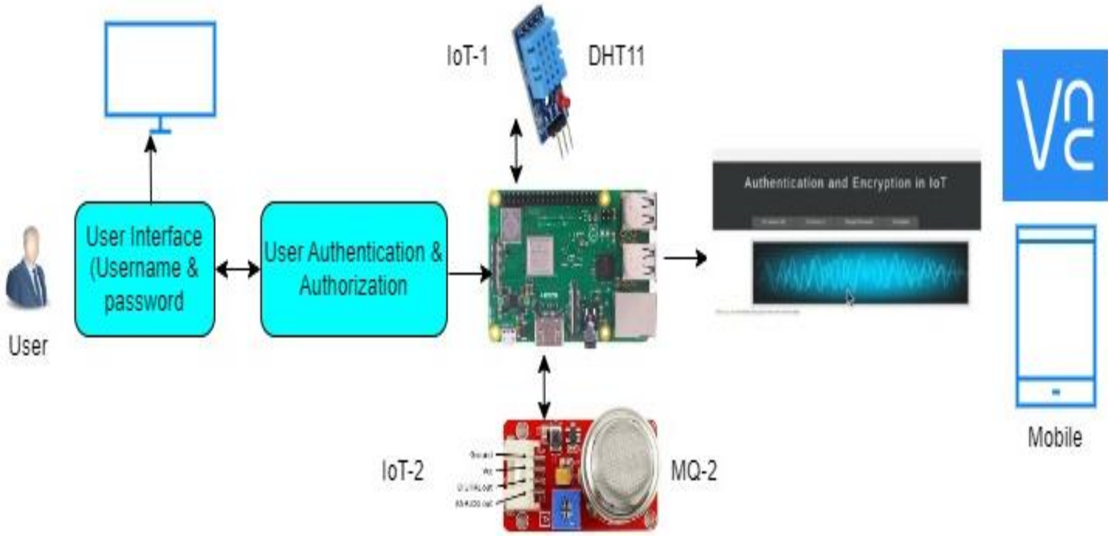

IoT-1
DHT11
User
User Interface (Username & password
User Authentication & Authorization
Authentication and Encryption in IoT
Mobile
IoT-2
MQ-2

Fig. 3: Proposed system architecture

Figure 4 below shows diagram of proposed system. This system consists of 3 parts, raspberrypi 3 Model-B+ single-board computer, digital humidity and temperature sensor, gas sensor.

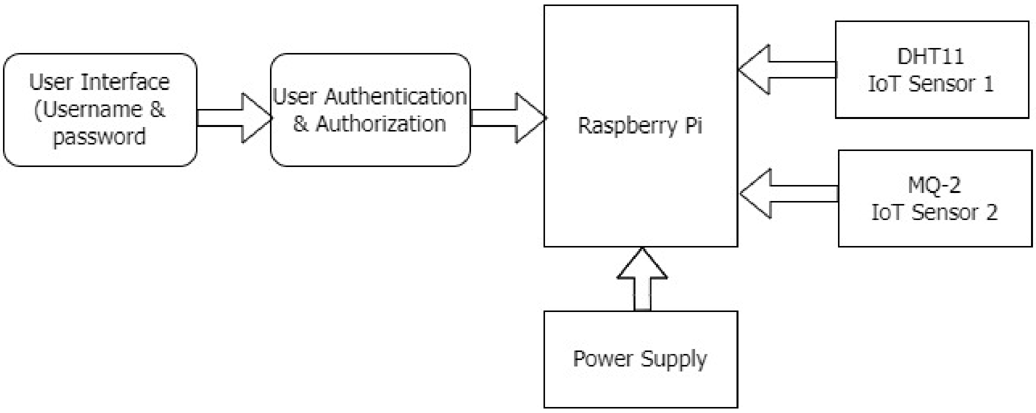

User Interface (Username & password
User Authentication & Authorization
Raspberry Pi
DHT11 IoT Sensor 1
MQ-2 IoT Sensor 2
Power Supply

Fig. 4: Block diagram of model

Raspberrypi was selected since it has tech specs, performance is high, fast processing of data and economical. DHT11 sensor is to observe humidity and temperature of the environment. MQ-2 gas sensor for detect leakage of gas / smoke detected from the environment. This MQ-2 smoke sensor can also acts like as alarming system when fire catches in a firm.

Implementation & Methodology

As discussed in the above block diagram in fig 5.4, implementing the process of authorization & authentication is seen from flow chart in figure 5

Raspberry-pi Model 3B+

The most recent Raspberry model model 3B+, features a 64-bit four core CPU operating at 1.4 GHz, dual-band 2.4 GHz and 5 GHz wireless LAN, Bluetooth 4.2/BLE, faster Ethernet, and Power over Ethernet (PoE) functionality through an additional PoE HAT. The modular compliance certification for the dual-band wireless LAN enables the board to be included into finished products with much less wireless LAN compliance testing, reducing both cost and time to market. The mechanical footprint of the Raspberry-pi model 3B+ is identical to that of the raspberry-pi model 2B and the raspberry-pi model 3B.

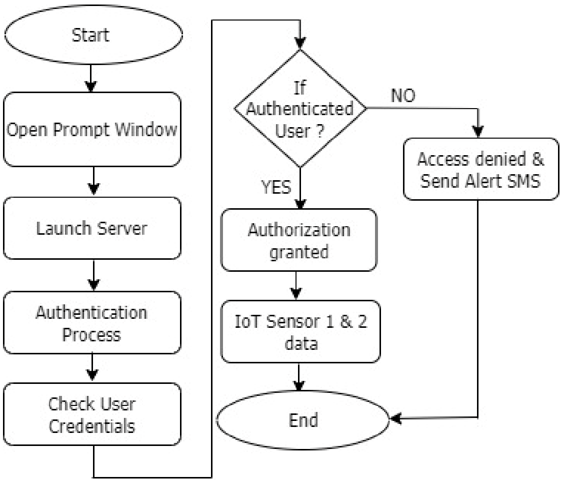


Fig. 5: Flow Chart

Raspberry Pi Specification:

- Processor: BCM2837BO Broadcom, 64-bit S0C Cortex-A53 at 1.4 GHz
- Memory: SDRAM of 1 GB LPDDR2
- Connectivity: 2.4 G Hz and 5 G Hz IEEE 802.11.b/g/n/ac w/l LAN, blue-tooth 4.2, BLE, USB 2.0 over GB Ethernet (maximum throughput 300Mbps), 4 × USB 2.0 ports.
- Access: 40-pin extended GPI0 header
- Input power: 5V/2.5 Amps DC via the micro USB connector, 5V DC via GPIO header

DHT11 sensor

We can check the digital temp. & humidity measurements. The module is essentially a printed circuit board (PCB) with a 4-pin DHT11 sensor connected to it along with a few other components. It has three wires: VCC, GND, & DATA. From DHT11, we can measure humidity in the 20% to 90% range (at 25 °C). Following are the specifications of DHT11 sensor:

- Dimensions 13 x 29 x 8 mm
- 3 - 5.5V power supply
- Average power consumption 0.2 mA
- Communication: 1-Wire
- Temperature measurement range: 0 - 50 ° C
- Humidity measurement range: 20%-90% RH (at 25 ° C) - in the full temperature range - 30% -80%

MQ-2 Sensor

Monitoring a quality of air, gas leak alarm, and maintaining environmental requirements in Industries may all benefit from air gas detection. In both the home and the workplace, gas leak detection equipment uses the MQ-2 Sensor. These sensors are effective in picking up smoke, methane, alcohol, i-butane, propane, and LPG. Fig 5.2 above, from elecrow MQ-2 datasheet, shows the internal organisation & structure of MQ-2 gas sensor. It is composed of a micro AL2O3 ceramic tube (5 in picture), a SnO2 Tin Dioxide sensitive layer (1), a measuring electrode (2) and a heater (4). These are fixed and made of plastic and net of stainless-steel. A potentiometer (RL) makes it possible to tune the sensor in different temperature/humidity conditions, and/or to set a threshold when an analog port is unavailable and you want to get a digital output: with gas presence in air, the MQ-2 resistance decreases, so resulting in higher current passing to the external microcontroller measuring the sensor

DOI: 10.5123/inj.2023.4.inj52

analog output. For delicate components, the heater offers the ideal working environment. Four of the six pins on the wrapped MQ-2 are used to retrieve signals, while the other two are utilised to supply heating current.

Below are the pin details of MQ-2 sensor (Crowtail Gas Sensor 2.0)

- Vcc : From this PIN the MQ-2 sensor takes the positive power
- Ground : This PIN goes to the microcontroller ground
- Digital Out: This PIN outputs a 0 or 1 when a threshold is reached.
- Analog Out : This PIN gives the analog output depending on gas intensity in air

Results & Discussion

IoT sensor 1 (DHT11) and IoT sensor 2 (MQ-2) may now be utilised to monitor environmental temperature, humidity, and gas leaks after the basic hardware connection setup, as illustrated in the following Figure 6.

The Raspbian Wheezy OS was choosen for the Raspberry Pi since the manufacturer recommends it. The Raspberry Pi runs a Linux-based operating system. The Debian Cheezy OS has been tweaked to become the Raspbian Wheezy. The Win32 Disc Manager was used to extract the OS image to an SD card. At first, the OS was conFigured with login information and IP address settings. To ensure that the Raspberry Pi won't ever request login information in the event of a power outage, the login at startup feature was deactivated. To turn the raspberry-pi into a web server, the HTTP webserver was installed. The IDLE programming environment is used to develop the Raspberry Pi's code.

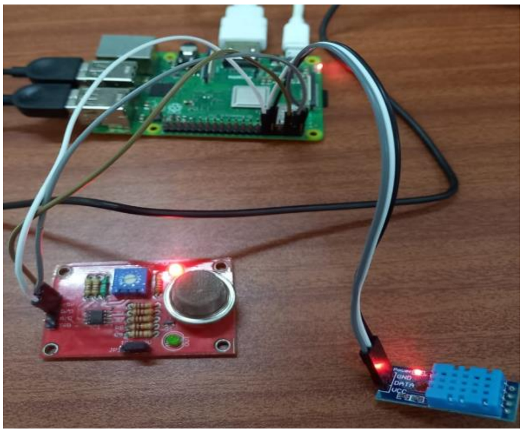
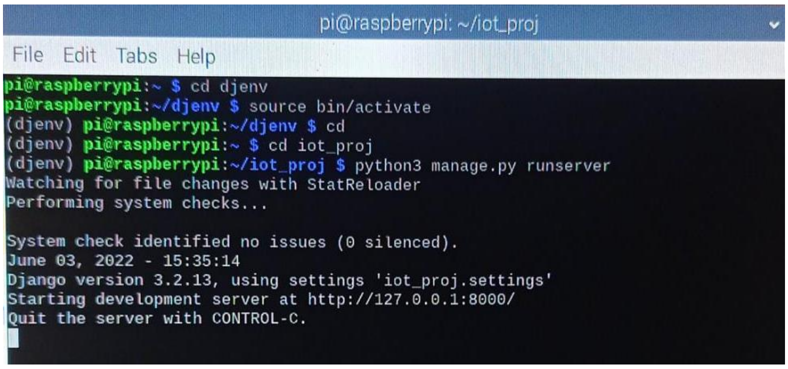


Fig. 6: Connection of IoT 1 & 2 to raspberry-pi

In the Figure 7 below, prompt window is seen with the commands to run python as a webserver. Once the code is executed, web server will be launched with the address http://127.0.0.1:8000/



```
pi@raspberrypi:~ $ cd djenv
pi@raspberrypi:~/djenv $ source bin/activate
(djenv) pi@raspberrypi:~/djenv $ cd
(djenv) pi@raspberrypi:~ $ cd iot_proj
(djenv) pi@raspberrypi:~/iot_proj $ python3 manage.py runserver
Watching for file changes with StatReloader
Performing system checks...

System check identified no issues (0 silenced).
June 03, 2022 - 15:35:14
Django version 3.2.13, using settings 'iot_proj.settings'
Starting development server at http://127.0.0.1:8000/
Quit the server with CONTROL-C.
```

Fig. 7: cmd promt window showing server address

Figure 8 below shows the webpage to enter login username and password to access the monitored parameters like temperature, humidity and gases. The user who needs to access these parameters, user need to enter correct username and password. If the entered username and password is matched with the original ones, then access is granted to the user to monitor these parameters.

After entering the correct username and password, the user name check the parameters in the next web page whose screenshot shown in the Fig. 9 below.

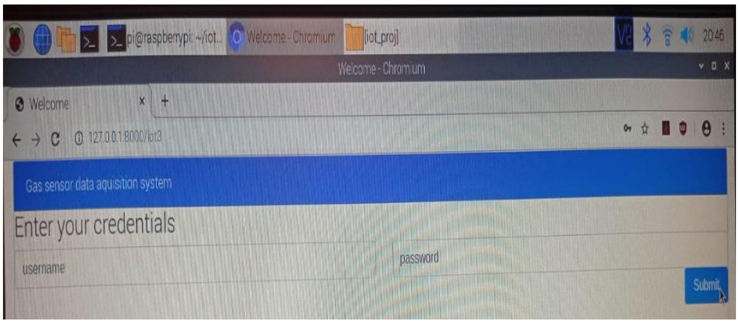


Fig. 8: Page to enter login credentials

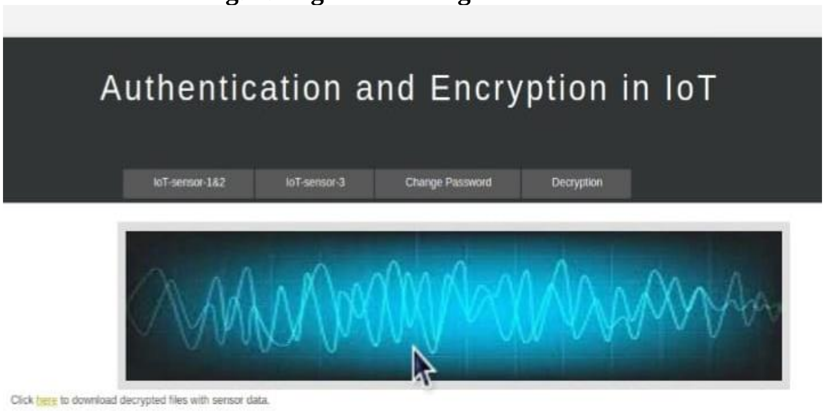
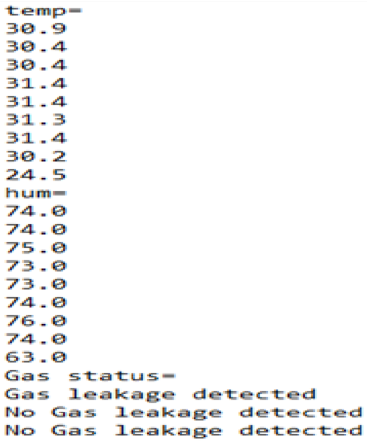

Authentication and Encryption in IoT
IoT-sensor-1&2
IoT-sensor-3
Change Password
Decryption
Click here to download decrypted files with sensor data.

Fig. 9: User Interface to fetch sensor data

Figure 10 below displays a snapshot of the results from the IoT sensor DHT11 which is termed as IoT sensor 1 and 2, & MQ-2 gas sensor which is termed as IoT sensor 3.



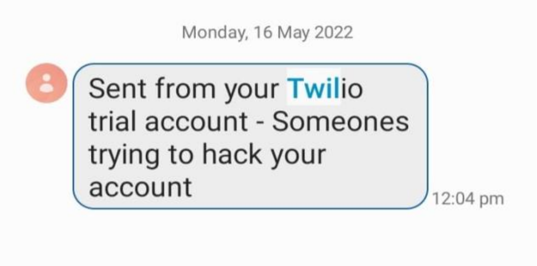
```
temp=
30.9
30.4
30.4
31.4
31.4
31.3
31.4
30.2
24.5
hum=
74.0
74.0
75.0
73.0
73.0
74.0
76.0
74.0
63.0
Gas status=
Gas leakage detected
No Gas leakage detected
No Gas leakage detected
```

Fig. 10: Results of IoT sensor data

If any user tries to fetch the sensor data / hack the sensor data and enters the password wrongly, permission is not granted to the user and immediately short message service (SMS) using twilio trial account (free of cost).

SMS will be sent to mobile number of an authorized / designated person appointed for monitoring IoT sensor data. SMS from Twilio trial account is as shown in the below Figure 11. Message will be sent as “someones trying to hack your account”. With this SMS, owner can immediately change the login id and password. This way, data can be protected from unauthenticated users.

We can send SMS with Twilio Programmable Messaging. Twilio makes sending and receiving SMS easy. Twilio provides products and services to help our innovative Internet of Things (IoT) applications, connect them to reliable cellular networks worldwide, and secure them for life. Developer documentation will show you how to integrate our connectivity and device builder tools into our products and our cloud. We can choose from our programming language and dive in. We’ve got helper libraries and Quick starts to get you sending SMS and MMS in our web app, fast.



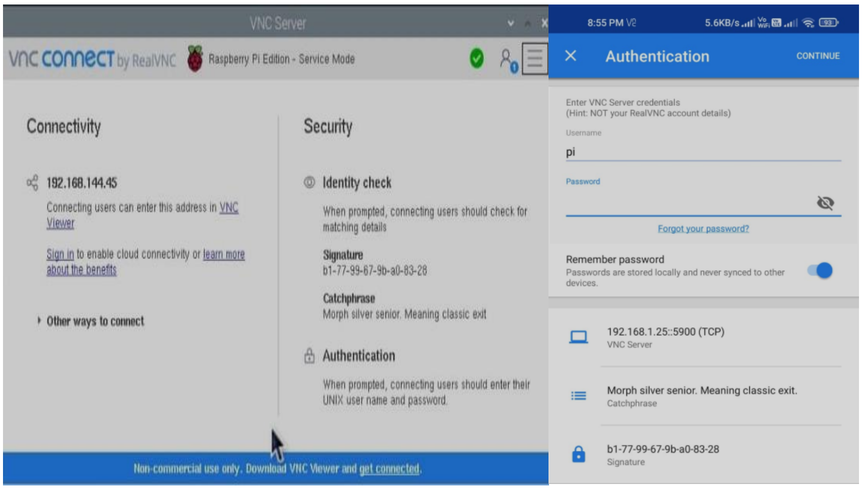
Monday, 16 May 2022

Sent from your Twilio trial account - Someones trying to hack your account

12:04 pm

Fig. 11: SMS alert

We can also use VNC viewer tool in personal laptop, system or even in mobile to monitor parameters from a different locations. A screen shot of VNC viewer is given in the below Figure 12.



Connectivity

192.168.144.45

Connecting users can enter this address in VNC Viewer

Sign in to enable cloud connectivity or learn more about the benefits

Other ways to connect

Security

Identity check

When prompted, connecting users should check for matching details

Signature
b1-77-99-67-9b-a0-83-28

Catchphrase
Morph silver senior. Meaning classic exit

Authentication

When prompted, connecting users should enter their UNIX user name and password.

Non-commercial use only. Download VNC Viewer and get connected.

Authentication

Enter VNC Server credentials
(Hint: NOT your RealVNC account details)

Username: pi

Password:

Forgot your password?

Remember password
Passwords are stored locally and never synced to other devices.

192.168.1.25::5900 (TCP)
VNC Server

Morph silver senior. Meaning classic exit.
Catchphrase

b1-77-99-67-9b-a0-83-28
Signature

Fig. 12: VNC Viewer (a) computer (b) Mobile Application

Conclusion

The proposed work here, utilises the Base64 encryption and decryption technique, that illustrate the principles of user authentication and authorization in an IOT environment. The raspberry-pi model, combined with IoT sensors named DHT11 and MQ-2, used to observe the parameters like temperature, humidity, and gas detection in real-time on a round-the-clock basis. The info is delivered to raspberry-pi by these IoT sensors. This device then encrypts the parameter data, and since a powerful Base64 algorithm is employed for both encryption and decryption, the user can view the decrypted data following the process of decryption. By verifying the user's user credentials, authentication and authorization concepts are applied. There will be fewer opportunities to decode it because a strong username and password have been created. Since every user or computer can be authenticated and Base64 encryption methods are used, this method solves some of the security problems in IoT networks.

References

1. Kumar Sekhar Roy, Hemanta Kumar Kalita, “A Survey on Authentication Schemes in IoT”, IEEE International Conference on Information Technology, 2017.
2. Muhammad Naveed Aman, Sachin Taneja,” Token-Based Security for the Internet of Things with Dynamic Energy-Quality Trade off”, IEEE Internet of Things Journal, 2018.
3. Zahoor Ahmed Alizai, Noquia Fatima Tareen, “Improved IoT Device Authentication Scheme Using Device Capability and Digital Signatures”, International Conference on Applied and Engineering Mathematics, 2018.
4. Seyed Farhad Aghili, Sima Arasteh, Hamid Mala, “A New Lightweight Authentication and Key agreement Protocol For Internet of Things” , 13th International ISC Conference on Information Security and Cryptology (ISCISC2016) September 7-8, 2016.
5. Wei-Cheng Lin, Jun-Ya Lee, “A Lightweight Authentication Protocol for Internet of Things”, IEEE, 2014.
6. Almulhim, M. and Zaman, N., “Proposing Secure and Lightweight Authentication Scheme for IoT Based E-Health Applications”, International Conference on Advanced Communications Technology (ICACT), ICACT 2018, pp.481–487, Feb. 2018.
7. Sandeep K.V., Dr.T.C.Manjunath, “Design and implementation of security mechanism by user authentication for voting system based on Fernet encryption and Blockchain technique”, Scopus Indexed Journal Article, SCImago Journal & Country Rank - Quartile 3 (Q3) Journal, SJR 2022 Rating 0.25, *Journal of European Chemical Bulletin*, Section A-Research paper, e-ISSN 2063-5346, H-Index 11, Vol. 12, Special Issue 6 (Si6), pp. 3354 – 3369, 2023.
8. Sandeep K V, Dr.Sayed Abdulhayan, “Implementation of Data Integrity using MD5 and MD2 Algorithms in IoT Devices”, *Palarch's Journal Of Archaeology Of Egypt/Egyptology-PJAEE(Scopus Q3), vol. 17, no. 7, pp. 7388 - 7395, ISSN: 1567-214X, Nov. 2020.*
9. Sandeep K.V, Dr. T.C. Manjuanth, “A Novel Mechanism for Design and Implementation of Confidentiality in Data for the Internet of Things with DES Technique”, *6th IEEE International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)*, I-SMAC 2022, Dharan, Tribhuvan University, Purwanchal Campus, Nepal, IEEE XPLORE COMPLIANT ISBN: 978-1-6654-6941-8, IEEE DVD Part Number: CFP22OSV-DVD; ISBN: 978-1-6654-6940-1, Paper ID 561, 10-12, pp. 109-114, November 2022. DOI:10.1109/I-SMAC55078.2022.9987268.
10. Sandeep K.V, Dr. T C Manjuanth, “Design & Implementation of data privacy & security using IoT sensors in remote health monitoring system”, *Tuijin Jishu/ Journal of Propulsion Technology,* SCImago Journal & Country Rank - Quartile 3 (Q3) Journal, SJR 2022 Rating 0.32, H-Index 24, Scopus Indexed Q3, ISSN:1001-4055, Vol. 44, Issue No. 3, Oct 2023.
11. Dr. T C Manjuanth., Dr. Sandeep K.V., "Development of the Implementation of Secured Data Communication on IoT Applications with hardware prototype development in IoT Devices", *Indian Patent No. 202341067445*, October 13th 2023.
12. “Design & implementation of security mechanism by user authentication for voting system based on Fernet encryption and blockchain technique”, *adapted from Copyright by Dr. Sandeep K.V., Dr. T C Manjuanth*, Oct 2023, Diary No. 27462/2023-CO/L
13. Sandeep K.V, Dr. T. C. Manjunath, “Design of a user authentication system & data secured hardware prototype model of a next-gen intelligent voting system based on Blockchain & IoT”, Journal of Research Administration, Scopus Indexed Journal, SCImago Journal & Country Rank - Quartile 4 (Q4) Journal, SJR 2022 Rating 0.15, H-Index 3, ISSN:1539-1590, E-ISSN:2573-7104, Vol. 5, Issue No. 2, pp. Nov 2023.