Lightweight Framework to Secure the End-to-end Communication of IoMT Devices

Author

Noor Mujdded Choudary

Regn Number

00000328645

Supervisor

Dr. Bilal Muhammad Khan

DEPARTMENT OF CYBER SECURITY

PAKISTAN NAVY ENGINEERING COLLEGE

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

ISLAMABAD

OCTOBER 2023.

Lightweight Framework to Secure the End-to-end Communication of IoMT Devices

Author

Noor Mujdded Choudary

Regn Number

00000328645

A thesis submitted in partial fulfillment of the requirements for the degree of

MS Cyber Security

Thesis Supervisor:

Dr. Bilal Muhammad Khan

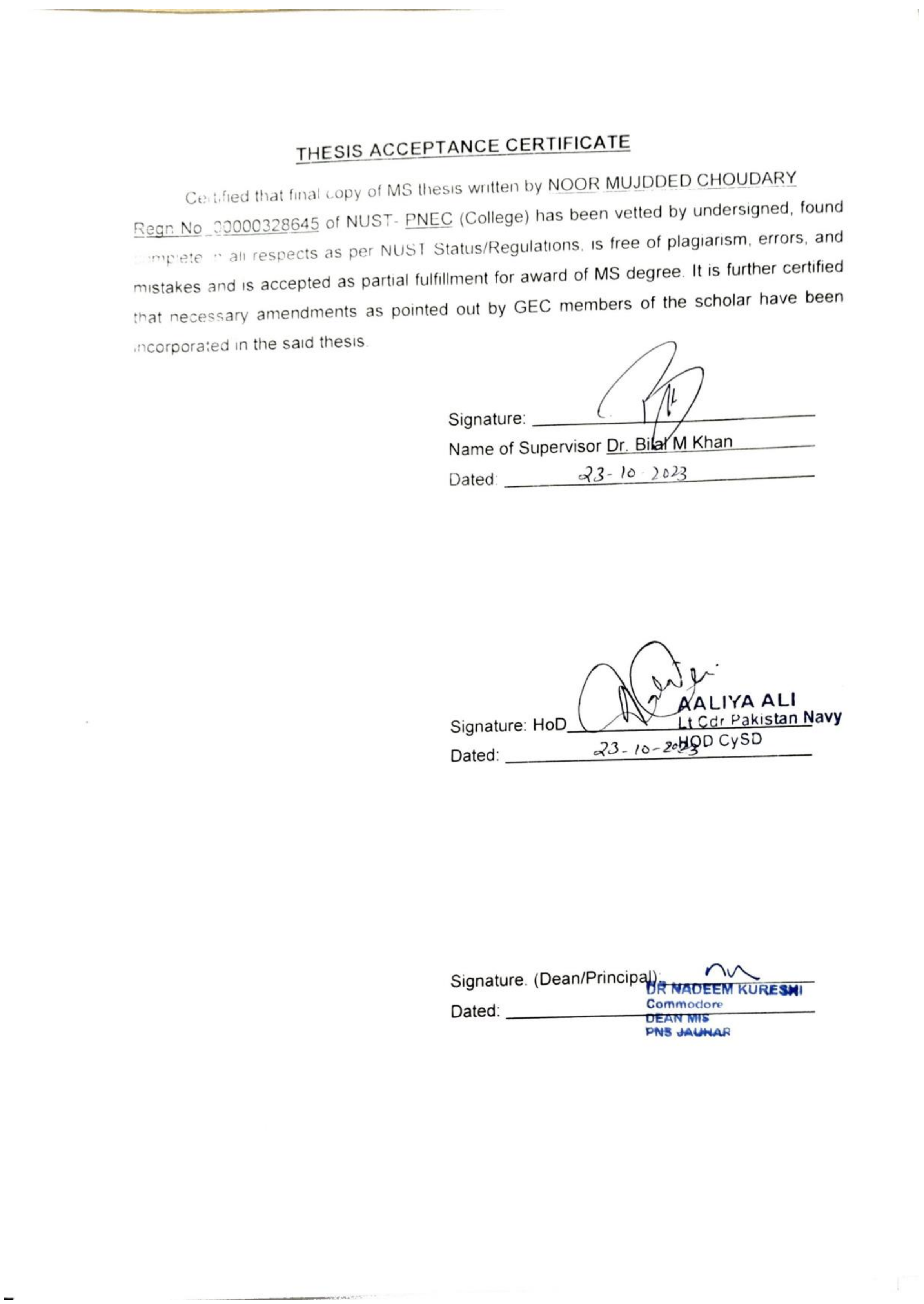
DEPARTMENT OF CYBER SECURITY

PAKISTAN NAVY ENGINEERING COLLEGE

NATIONAL UNIVERSITY OF SCIENCES AND TECHNOLOGY

ISLAMABAD

OCTOBER, 2023



THESIS ACCEPTANCE CERTIFICATE

Certified that final copy of MS thesis written by NOOR MUJDDED CHOUDARY Regn No 00000328645 of NUST- PNEC (College) has been vetted by undersigned, found complete in all respects as per NUST Status/Regulations, is free of plagiarism, errors, and mistakes and is accepted as partial fulfillment for award of MS degree. It is further certified that necessary amendments as pointed out by GEC members of the scholar have been incorporated in the said thesis.

Signature: __________

Name of Supervisor Dr. Bilal M Khan

Dated: 23-10-2023

Signature: HoD __________

AALIYA ALI
Lt Cdr Pakistan Navy
HOD CySD

Dated: 23-10-2023

Signature. (Dean/Principal): __________

DR NADEEM KURESHI
Commodore
DEAN MIS
PNS JAUHAR

Dated: __________

APPROVAL

It is certified that the contents and form of the thesis entitled " Lightweight framework to Secure the end to end communication of IoMT devices " submitted by "Noor Mujdded Choudary" have been found satisfactory for the requirement of the degree.

Advisor Dr Bilal M Khan
Signature: ___________
Date: 23-10-2023

Committee Member Dr. Rashida Ali Memon
Signatue: ___________
Date: 23-10-2023

Committee Member Dr. Lubna Moin
Signatue: ___________
Date: 23-10-2023

FORM TH-4

National University of Sciences and Technology

MASTER'S THESIS WORK

We hereby recommend that the dissertation prepared under our supervision by: (Student Name & Regn No.) NOOR MUJDDED CHOUDARY (00000328645) Titled: Lightweight framework to Secure the end to end communication of IoMT devices be accepted in partial fulfillment of the requirements for the award of Master's degree.

EXAMINATION COMMITTEE MEMBERS

1. Name: Dr. Rashida Ali Memon Signature: ____________

2. Name: Dr. Lubna Moin Signature: ____________

3. Name: ____________ Signature: ____________

Supervisor's name: Dr. Bilal M Khan Signature: ____________

Date: 23-10-2023

AALIYA ALI
Lt Cdr Pakistan Navy
HOD CySD

Head of Department

23-10-2023
Date

COUNTERSIGNED

Date: 23-10-2023

DR NADEEM KURESHI
Commodore
DEAN MIS
PNS JAUHAR

Dean / Principal

CERTIFICATE FOR PLAGIARISM

1. It is certified that PhD / M.Phil / **MS** Thesis Titled **"LIGHTWEIGHT FRAMEWORK TO SECURE END TO END COMMUNICATION OF IOMT DEVICES."** by **NOOR MUJDDED CHOUDARY (2020-NUST-MS Cyber Security (CyS Fall 20)** has been examined by us. We undertake the follows:

a. Thesis has significant new work / knowledge as compared already published or is under consideration to be published elsewhere. No sentence, equation, diagram, table, paragraph or section has been copied verbatim from previous work unless it is placed under quotation marks and duly referenced.

b. The work presented is original and own work of the author (i.e. there is no plagiarism). No ideas, processes, results or words of others have been presented as Author own work.

c. There is no fabrication of data or results which have been compiled / analyzed.

d. There is no falsification by manipulating research materials, equipment, or processes, or changing or omitting data or results such that the research is not accurately represented in the research record.

e. The thesis has been checked using TURNITIN (copy of originality report attached) and found within limits as per HEC Plagiarism Policy and instructions issued from time to time.

Name & Signature of Supervisor

DR. BILAL MUHAMMAD KHAN
Tenured Associate Professor
Director R&D
NUST-PNEC

CERTIFICATE OF ORIGINALITY

I certify that this research work titled "*Lightweight Framework to Secure the End-to-end Communication of IoMT Devices*" is my own work. The work has not been presented elsewhere for assessment. The material that has been used from other sources has been properly acknowledged / referred.

Signature of Student

Noor Mujdded Choudary
2020-MS-CYS 00000328645
PNEC

ACKNOWLEDGEMENTS

All praise and thanks belong to **THE ALMIGHTY ALLAH**, the Merciful, the sole Creator of the universe and the source of all knowledge and wisdom, WHO has bestowed upon me health, thoughts, talented instructors, and helpful companions. I offer my deepest gratitude to **The Holy Prophet Hazrat Muhammad** (Peace Be Upon Him), Whose moral and spiritual teachings enlightened my heart and intellect and inspired me to pursue lofty life goals.

I feel highly privileged to express my gratitude to my sincere and honorable **Dr. Bilal Muhammad Khan**, Associate Professor, Pakistan Navy Engineering College, NUST for his keen interest, unflagging guidance, and compassionate attitude throughout the research. Without his impassioned scholarly direction, this manuscript would not have been published in its current guise. Following, it was my loved ones' prayers that made it feasible to finish this research project. I would like to extend my heartfelt gratitude to my **Parents** and **Siblings** for their unwavering support during my research project. Their sincere assistance and encouragement throughout the entire process are deeply appreciated.

Dr. Rashida Ali Memon, Lt. Cdr Qurrat ul Ain TI(M) and Dr. Lubna Moin have provided me with consistent encouragement and cordial cooperation throughout the entirety of my research project, for which I am indebted and profoundly appreciative. I wish to express my gratitude for the valuable support and guidance provided by my brother, **Mr. Zahoor Mujdded Choudary**, who was instrumental in assisting me throughout this journey."

Dedicated to my exceptional parents and adored siblings whose tremendous support and cooperation led me to this wonderful accomplishment.

Abstract

The Internet of Medical Things (IoMT) is widely recognized as an essential tool of health informatics. Internet of Medical Things (IoMT) enabled devices are employed for monitoring patients inside hospital and home settings. These devices facilitate the collection and transmission of biological data, encompassing measurements such as blood pressure, electrocardiography (ECG), blood sugar levels, body temperature, as well as room temperature, among others. Wearable devices have been increasingly utilized in various healthcare applications. These devices produce data in real-time and communicate it to nearby gateways and remote servers for the purpose of processing and visualization. The utilization of the Internet of Medical Things (IoMT) is gaining significant traction within the realm of intelligent healthcare. These devices exhibit limitations in terms of available resources and are susceptible to security breaches. Due to inadequate security and privacy safeguards, the IoMT (Internet of Medical Things) is particularly vulnerable to malicious attacks, making it a prime target for adversaries within the healthcare network architecture. The creation of a streamlined architecture designed to enhance the security of communication inside an Internet of Medical Things (IoMT) sensor network was studied. The MQTT protocol, known for its lightweight nature, was employed for facilitating communication among the sensors utilized on the Internet of Medical Things (IoMT) system. An integrated security framework that incorporate lightweight cryptography is proposed. Risk assessment was performed to measure the effectiveness of the framework. The framework covered the components of the Internet of Medical Things (IoMT) and addressed their security aspects.

Key Words: *IoMT security framework, lightweight security, healthcare system security, Sensor network security.*

Table of Contents

List of Figures

List of Tables

CHAPTER 1: INTRODUCTION

The term "Internet of Medical Things" (IOMT) denotes a dynamic nexus between the healthcare industry and the Internet of Things (IoT) technology environment. It is essential to first understand the fundamentals of IoT in order to comprehend this idea. The aim behind the Internet of Things is to link commonplace items and products to the internet, creating a massive network that makes data and information sharing easier. Because of this interconnection, there are now smart homes, smart cities, and effective industrial processes, where decision-making is based on data-driven insights.

By seamlessly integrating medical equipment and healthcare systems with the IoT, IoMT expands on this concept in the field of healthcare. This implies that all aspects of the healthcare industry, including medical devices and equipment, are connected into this massive digital network. Let us contemplate a scenario whereby interconnected equipment are employed to perpetually observe an individual's cardiac rhythm, arterial pressure, and adherence to prescribed medications. Subsequently, individuals working in the medical domain are granted secure authorization to access this contemporaneous data, so affording them an accurate and up-to-date understanding of the patient's health status.

IoMT has the ability to completely transform how healthcare is delivered. Healthcare has often been defined by infrequent visits to medical institutions, restricted access to real-time data, and a reactive strategy for handling health conditions [1]. However, this paradigm is altered by the IoT technologies' interaction with medical equipment and systems. It has the potential to improve patient care outcomes by allowing medical practitioners to take proactive action. For instance, deviations from typical health metrics may result in alarms that direct healthcare professionals to act right away to avert possible health problems.

IoMT integration simplifies a number of medical procedures. Automated systems that connect fluidly with one another enhance administrative chores, patient monitoring, and even drug administration [2]. This lessens the workload on medical personnel, lowers mistakes, and boosts overall effectiveness. The fact that IOMT offers up new horizons for healthcare management is among its most exciting features.

Predictive models develop using the ongoing stream of real-time health data, providing insights into illness trends and population health patterns [3]. Using this data together with sophisticated analytics, public health efforts, resource allocation, and the creation of individualized treatment programs may be guided. Additionally, by integrating telemedicine and virtual care, IOMT makes healthcare available to those who live in distant locations or have mobility issues.

IOMT and IoT technology confluence marks a seismic change in the healthcare industry. In addition to empowering people to take an active role in their own health, it also equips healthcare professionals with data-driven tools to provide better, more individualized treatment [4]. However, it's essential to address issues with data privacy, security, and ethical considerations as with any technology innovation. Although there are enormous potential advantages, they must be carefully considered to guarantee that patient information is kept private, and that technology is used appropriately.

Medical wearables are a particularly significant component of the vast IoMT. These gadgets, which often take the shape of fitness trackers and smartwatches, serve as concrete illustrations of how cutting-edge technology is being effortlessly incorporated into our everyday lives to improve health and wellbeing [5].

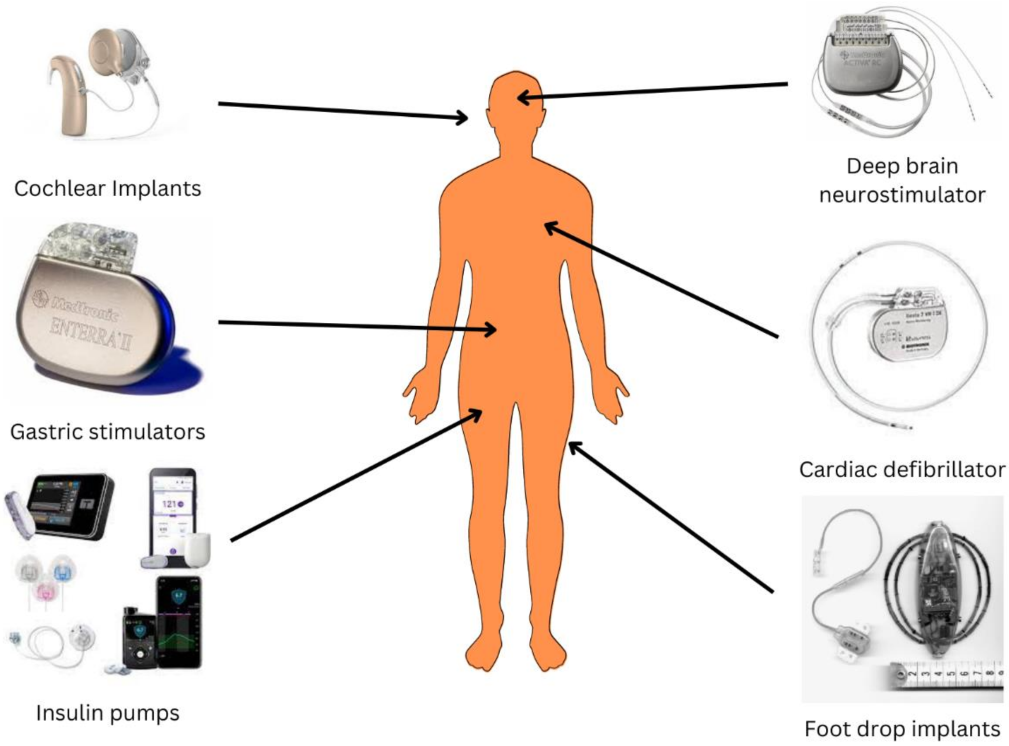


Figure 1.1: Wearable IoMT devices

These wearables stand out because of the sensors they include, which carefully record a range of crucial information about a person's physical condition and activities.

Take the common smartwatch, for instance. It can monitor a variety of physiological characteristics, including body temperature, blood pressure, and heart rate, in addition to indicating the time. various wearables provide a complete picture of the wearer's health that changes in real time by continuously monitoring various factors. Additionally, some cutting-edge devices include an electrocardiogram (ECG) capability that enables the identification of abnormal heart rhythms that may point to cardiac problems [6]. Medical wearables offer continuous, unobtrusive monitoring, which leads to the acquisition of a wealth of health-related data. This wealth of knowledge serves two functions. On the one hand, it gives people the ability to be more aware of their own health situation. Wearers monitor their fitness objectives' progress, see patterns in their heart rates during activity or rest, and learn how environmental elements like stress or the weather impact their well-being [7]. This self-awareness encourages a proactive attitude to health, which may inspire individuals to choose better lifestyle options and to act quickly in the event of abnormalities.

The healthcare industry benefits greatly from this real-time data stream. By using this constant flow of health measurements, doctors are given an unmatched perspective into the lives of their patients in between appointments. Data patterns reveal possible health problems before they become more serious, allowing for prompt and targeted treatments. For example, a healthcare professional becomes concerned about a steady increase in blood pressure, and makes the required changes to medication or lifestyle advice [8]. Similar to this, abnormal heart rhythms detected by the ECG feature could lead to a cardiologist's consultation and avert a possibly life-threatening cardiac episode [9].

The medical wearables' extensive usefulness has several facets. Anonymized data from wearables may help epidemiological studies and research into health patterns when gathered and examined on a bigger scale [10]. Insights that have relevance for public health policies and initiatives may be discovered by researchers through discovering connections between lifestyle decisions, environmental variables, and health outcomes [11]. On multiple data platforms, the data that has been acquired by the various IOMT devices is compiled and analyzed. These platforms offer extremely helpful insights into patient health trends, as well as management strategies for population health and individualized treatment regimens. On the other hand, despite the fact that IOMT has a lot of advantages, there are some worries about the privacy of patients and their data. As a result of the fact that these devices handle sensitive health information, stringent rules and security measures are required in order to guarantee patient safety and preserve the confidentiality of medical data in a healthcare environment that is becoming increasingly networked [12].

1.1. Global scenario of IOMT devices in relation to Pakistan

IoMT is an emerging sector that is seeing significant growth, with an estimated valuation of USD 187.60 billion by the year 2028. The IoMT market encompasses medical devices that are interconnected with the internet, enabling them to gather, transmit, and analyze data through a network. The COVID-19 pandemic has had an extraordinary and significant global influence, particularly in relation to the internet of medical things. This technology has had a notable increase in demand across all areas during this time. The IoMT market is anticipated to exhibit a compound annual growth rate (CAGR) of 29.5% for the period spanning from 2021-28 [13]. It is anticipated that North America will exhibit the highest market share in the IoMT market. Some of the prominent IoT software businesses in the industry are Microsoft, Amazon Web Services (AWS), Siemens, IBM, Cisco, Oracle, PTC, and MongoDB [14]. The market for the IoMT presents distinct legal, regulatory, technical, and privacy obstacles due to the extensive involvement of several parties within the IoMT ecosystem [15]. Healthcare organizations are confronted with an imperative to address the security concerns associated with the IoMT in a proactive manner.

IoMT is seeing significant global growth and holds great potential within the healthcare sector. IoMT devices have the capability to gather and share data through internet connectivity. These devices are employed to enable the monitoring of patients in real-time from remote locations, hence minimizing the need for in-person visits, reducing hospital stays and readmissions, and facilitating the provision of tailored care therapies [16]. The IoMT has been found to effectively mitigate the operational expenses of healthcare institutions, facilitate accurate disease diagnosis, minimize error rates, and enhance workflow efficiency for healthcare professionals and physicians. The market growth pattern suggests a substantial potential for the adoption of IoMT in

Pakistan's healthcare sector. The IoMT has the potential to improve healthcare delivery by offering healthcare professionals access to real-time data and valuable insights [17]. This phenomenon has the potential to result in enhanced diagnostic precision, individualized treatment strategies, and improved patient prognoses.

IoMT devices are associated with security threats, similar to other technological advancements. The security controls of connected medical equipment are still restricted, but the potential hazards to hospitals' cybersecurity extend beyond the IoMT itself [18]. Ensuring the safeguarding of patient information and maintaining the integrity of the IoMT ecosystem are imperative imperatives for healthcare organizations in Pakistan [19]. Therefore, it is of utmost importance for these organizations to priorities data security and adopt effective cybersecurity measures. Ensuring the implementation of necessary security protocols in IoMT enabled smart healthcare systems is of paramount importance. Artificial intelligence (AI) can also be utilized to enhance security measures in the IoMT. This can be achieved by the detection of network intrusion and intermediary security threats within IoMT systems, as well as the execution of web-based security assessments utilizing an IoMT-SAF device, among other methods [20].

The successful implementation of IoMT in Pakistan requires the development of a robust healthcare infrastructure, including reliable internet connectivity and interoperability between different IoMT devices and systems [21]. The Pakistani government possesses the potential to significantly contribute to the advancement of the IoMT by offering incentives, establishing rules, and formulating policies that facilitate its widespread use. Public-private partnerships also facilitate the integration of IoMT into the existing healthcare system [22]. The extent of the IoMT in Pakistan corresponds to the international scale, offering the possibility of enhancing healthcare provision, decreasing expenses, and improving patient results [20]. Nevertheless, it is imperative to acknowledge the security obstacles and allocate resources towards the enhancement of infrastructure to fully exploit the advantages of the IoMT within the nation.

1.2. Current Developments in IOMT Market

The market of the IoMT is rapidly evolving, with new trends and developments emerging regularly. Here are some current trends and developments in the field of IoMT devices:

1.2.1. IoMT Applications and Use Cases:

The Internet of Medical Things is an area of study that is seeing tremendous growth and development, with its range of applications and potential uses developing within the context of Pakistan [23]. The field of IoMT encompasses a range of wireless devices, interconnected systems, and ecosystems that are designed to gather, process, and analyze large volumes of data. These IoMT solutions aim to offer enhanced insights that can be readily translated into practical actions. In recent years, there has been significant progress in the field of Internet of Things (IoT) with regards to several aspects such as next-generation wireless technologies, sensor technologies, communication protocols, computational capabilities, big data and artificial intelligence (AI) approaches, as well as on-device, edge, and cloud processing [24]. The IoMT is a significant subset within the broader Internet of Things (IoT) framework, with a specific emphasis on healthcare as a basic component of human well-being [25]. It is anticipated that IoMT will generate substantial revenue, amounting to several billion dollars. The field of IoMT encompasses various dimensions, encompassing the utilization of medical sensors and equipment for the purpose of gathering biomedical data [26]. These devices range from advanced and specialized surgical tools to basic body sensors.

The implementation of IoMT devices within the healthcare sector in Pakistan has the potential to yield numerous advantages. The IoMT integrates patient data and enables real-time monitoring of a healthcare facility's whole infrastructure, encompassing both equipment and personnel [27]. The IoMT has the potential to contribute to the reduction of healthcare expenditures through the mitigation of in-person visits and hospital admissions. This can prove to be particularly advantageous for individuals residing in rural regions who may encounter restricted availability of healthcare facilities. The IoMT has the potential to improve healthcare delivery by offering healthcare professionals immediate access to real-time data and valuable insights. This phenomenon has the potential to result in enhanced diagnostic precision, individualized treatment strategies, and improved patient prognoses.

The utilization of IoMT devices in healthcare settings in Pakistan has inherent security threats. It is of utmost importance for healthcare organizations in the country to effectively tackle these difficulties in order to ensure the safety and integrity of patient data and overall system functionality. Due to the inherent sensitivity and stringent regulatory requirements associated with healthcare data, the implementation of the IoMT necessitates the establishment of a more robust and secure infrastructure [15]. The utilization of the IoMT introduces significant concerns regarding privacy and security. This is primarily due to the fact that IoMT data commonly traverses the public Internet, hence increasing its exposure to a greater number of security threats compared to a more protected environment such as a firewalled private network. Original Equipment Manufacturers (OEMs) are required to employ robust communication protocols and encryption mechanisms in order to safeguard sensitive data [28]. Healthcare information technology (IT) teams must possess the necessary readiness to effectively handle IoMT devices and safely manage the associated data. To achieve this, technology partners can offer valuable courses and specialized knowledge in IoMT security [29].

1.2.2. Smart E-healthcare

Smart hospitals are healthcare facilities that are constructed using intelligent, automated, and optimized modules, potentially incorporating artificial intelligence and machine learning, within the information and communication technology (ICT) infrastructure [30]. The primary objective of these hospitals is to enhance patient care procedures and introduce novel capabilities [31]. Smart hospitals encompass a range of uses, including telemedicine, telehealth, and remote robot surgery. Telemedicine refers to the provision of clinical care services from a remote location, whereas telehealth pertains to the delivery of non-clinical care services at a distance [32]. In the field of remote robot surgery, surgical procedures are executed by medical robots under the guidance and instructions of a remotely located surgeon [33].

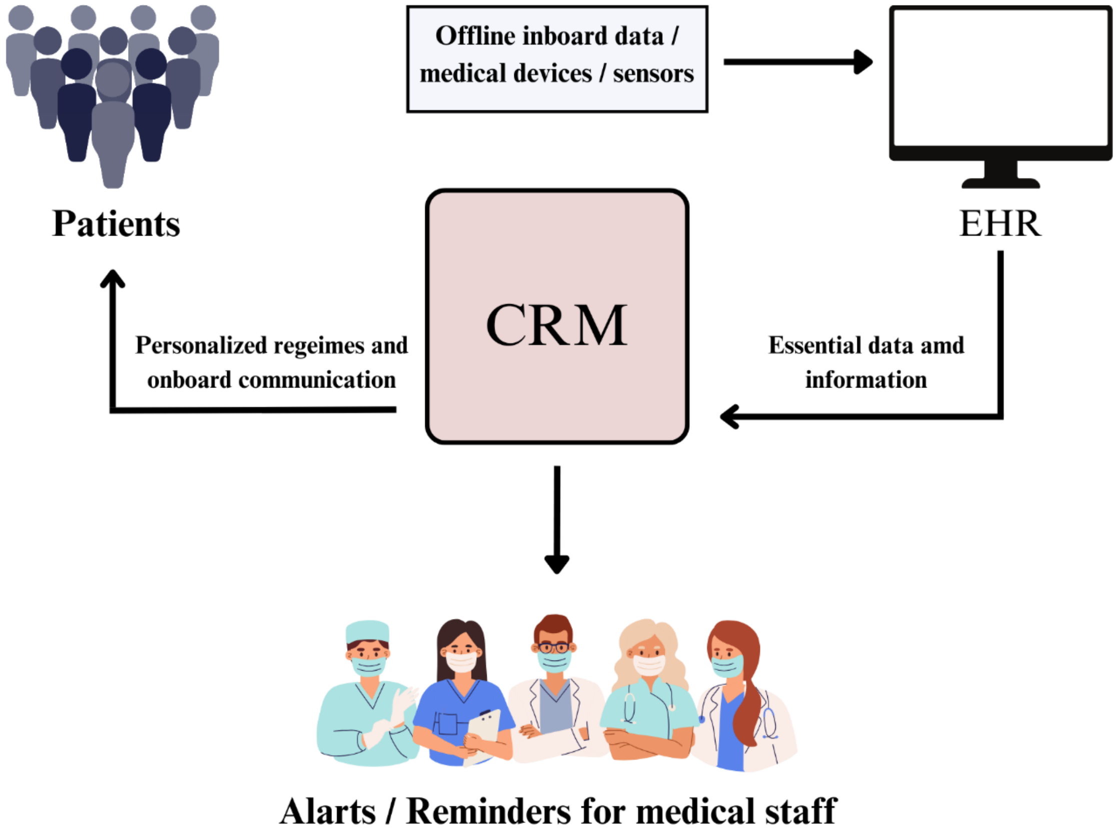

Offline inboard data / medical devices / sensors
Patients
EHR
CRM
Personalized regeimes and onboard communication
Essential data amd information
Alarts / Reminders for medical staff

Figure 1.2: Smart healthcare system

Figure 1.2 displays an example of a smart healthcare system in which the inbound data from various sources is first collected (e.g., by remote collecting or physical gathering) and transferred to EHR (Electronic Health records systems). Data could be considered as unstructured if it is collected offline on paper as medical notes by the professionals. If the data is collected in an organized way from the devices and sensors by using predefined data fields for users to fill, then it becomes straightforward to process in subsequent systems such as CRM (Customer Relational Management) System. The CRM brings to use the instruments for evaluating data and then assigning it to its predetermined goal in the ecosystem [34]. The vital data and information from EHR systems is given to the CRM system and it processes this patient data. The processed data elicits additional stimuli for both patients and medical workers within the ecosystem. Patients are provided with personalized health regimens through outbound communication from hospitals and health specialists. The healthcare professionals, including doctors and other medical personnel, receive notifications regarding reminders and alerts through the use of the same Customer Relationship Management (CRM) software within the ecosystem [35].

1.2.3. Real-time Patient Monitoring:

The field of the IoMT is experiencing tremendous growth, with its applications and use cases developing within the context of Pakistan. The IoMT facilitates continuous monitoring of individuals with chronic ailments, allowing for round-the-clock surveillance [36]. Moreover, it has the capability to notify healthcare professionals in the event that medical intervention becomes necessary. This entails the utilization of specialized sensor devices worn by patients, as well as the surgical implantation of subcutaneous sensors (e.g., for the purpose of monitoring glucose levels). The IoMT systems have the capability to function as virtual assistants, offering predictive capabilities for health issues, providing patient education regarding their symptoms, offering guidance on dietary and lifestyle choices, and notifying medical personnel in the event of emergency [37]. In certain instances, these devices may also incorporate a remotely operated first aid kit.

The implementation of IoMT devices within the healthcare sector in Pakistan has the potential to yield numerous advantages. The IoMT integrates patient data and enables real-time monitoring of a hospital's whole infrastructure, encompassing both equipment and personnel. The IoMT has the potential to mitigate healthcare expenditures by minimizing the necessity for in-person consultations and hospital admissions [38]. This can prove particularly advantageous for individuals residing in rural regions who may encounter restricted availability of healthcare establishments. The IoMT has the potential to improve healthcare delivery by offering healthcare professionals access to real-time data and valuable insights. This phenomenon has the potential to result in enhanced diagnostic precision, individualized treatment strategies, and improved patient prognoses.

The utilization of IoMT devices in healthcare settings in Pakistan presents inherent security vulnerabilities, necessitating the urgent attention and resolution of these issues by healthcare organizations [39]. Due to the inherent sensitivity and stringent regulatory requirements associated with healthcare data, the IoMT needs an infrastructure that is more robust and secure. The utilization of the IoMT introduces certain issues pertaining to privacy and security. This is primarily due to the fact that IoMT data often traverses the public Internet, hence increasing its vulnerability to a wider range of security threats compared to a more protected environment such as a firewalled private network.

1.2.4. Smart Hospitals:

It is projected that smart hospitals in Pakistan will witness a significant increase in the deployment of IoMT devices, surpassing 7 million by the year 2026 [40]. This figure represents a doubling of the recorded number of IoMT devices in 2021. The proliferation of IoMT devices is indicative of the rising acceptance and use of interconnected healthcare technologies across the nation [41]. These devices, both medical and non-medical, are interconnected and automatically feed patient data from monitoring devices, which is then recorded in electronic medical records.

The implementation of IoMT devices inside intelligent healthcare facilities yields numerous advantages. Real-time patient monitoring facilitates the remote monitoring of patients' health problems, enabling healthcare providers to intervene promptly when required [42]. This intervention has the potential to enhance patient outcomes and decrease the necessity for face-to-face consultations and hospital admissions. Furthermore, the utilization of IoMT devices facilitates the remote monitoring of patients who are deemed less critical and afflicted with infectious conditions, allowing them to remain in the familiar environment of their own residences [38]. This not only enhances the comfort experienced by patients but also alleviates the strain placed on healthcare institutions.

The predicted growth in IoMT devices in smart hospitals is significant, representing a 231% increase from the number deployed in 2021 [43]. This growth reflects the increasing recognition of the value and potential of IoMT in improving healthcare delivery and patient care. The deployment of a large number of IoMT devices in smart hospitals highlights the growing importance of connected healthcare technologies in Pakistan's healthcare industry.

1.3. Security Requirements and Solutions:

The implementation of the IoMT in Pakistan necessitates the establishment of a robust infrastructure to ensure its security. It is imperative to prioritize the implementation of necessary security measures in IoMT-based smart healthcare systems. Overview of the security needs and proposed solutions for the IoMT in the context of Pakistan are as under:

1.3.1. AI for Security:

Artificial intelligence (AI) has the potential to enhance security measures in the IoMT by effectively identifying network intrusions and mitigating intermediate security assaults within IoMT systems [44]. Additionally, AI contributes to web-based security assessments through the utilization of an IoMT-SAF device, among other applications. Artificial intelligence (AI) is also applied to tasks such as anomaly detection, threat intelligence, and predictive analytics in order to promptly identify and address security concerns [45].

1.3.2. Secure Communication Protocols:

Original Equipment Manufacturers (OEMs) are required to employ robust communication protocols and encryption mechanisms in order to safeguard sensitive data [46]. Secure communication protocols, such as Transport Layer Security (TLS) and Secure Sockets Layer (SSL), are utilized to apply encryption to data during its transmission [47].

1.3.3. Network Visibility:

The IT staff requires comprehensive awareness of the activities occurring within its network. Identifying the specific nature of the device can provide challenges due to the prevalence of unmanaged IoMT devices or those that do not conform to standard IT lifecycle activities [17]. The presence of unmanaged devices within a network poses an elevated risk due to their potential utilization of outdated operating systems. Network visibility is a crucial requirement for effectively identifying and securely managing IoMT devices and their associated data.

1.3.4. Robust Cybersecurity Measures:

It is imperative for healthcare information technology (IT) teams to possess the necessary readiness in effectively handling IoMT devices and safeguarding the associated data in a secure manner. The implementation of a comprehensive security policy that encompasses device management, continuous monitoring, and incident response is of utmost importance [48]. Technology partners have the capability to offer training and specialized knowledge in the domain of IoMT security.

1.4. Advantages of IOMT devices

The advent of the IoMT devices has ushered in a new era of healthcare innovation, replete with a multitude of advantages that extend across patient care, medical practice, and overall healthcare management [49]. These technologically advanced devices, seamlessly integrating the capabilities of medical equipment with the power of the Internet of Things (IoT), have the potential to reshape healthcare delivery, empower individuals, enhance clinical decision-making, and usher in a proactive approach to health management. From remote monitoring to personalized treatment plans, the advantages offered by IOMT devices are poised to drive improvements in patient outcomes, operational efficiency, and the way we approach modern healthcare [50].

The utilization of IoMT devices has both advantages and potential drawbacks. Several advantages can be observed, such as the utilization of real-time remote patient monitoring, leading to a decrease in the need for in-person visits. Additionally, there is a reduction in hospital stays and readmissions, along with the provision of more personalized care solutions. The IoMT facilitates continuous monitoring of individuals with chronic illnesses, allowing for round-the-clock surveillance [51]. Moreover, it has the capability to notify healthcare professionals in the event that medical intervention becomes necessary. The utilization of IoMT technology in healthcare settings has the potential to facilitate enhanced research capabilities and improve treatment outcomes [52]. The use of the IoMT has the potential to mitigate the financial burden on healthcare institutions by reducing the need for costly in-person visits. The utilization of remote video monitoring enables healthcare practitioners to remotely monitor patients post-discharge, ensuring medication adherence and promptly identifying warning signs for potential readmission [53]. Nevertheless, a significant limitation of the IoMT in the healthcare sector pertains to its inadequate security measures. The majority of IoMT devices were not originally developed with security considerations, rendering them particularly susceptible to potential breaches and compromises [54]. The exploitation of vulnerabilities in the IoMT empower hackers to engage in various nefarious activities, including gaining control over medical devices, illicitly acquiring sensitive patient health and personal data, and potentially jeopardizing patient well-being [55]. Consequently, healthcare organizations are confronted with a need to address the security concerns associated with the IoMT in a proactive manner.

1.4.1. Remote Monitoring and Management:

IOMT devices play a pivotal role in remote patient monitoring, allowing healthcare professionals to keep a constant watch over patients' health conditions from a distance [56]. This is especially valuable for individuals with chronic diseases such as diabetes, heart conditions, or respiratory issues. By collecting and transmitting data in real time, these devices enable timely intervention in case of any fluctuations or deviations from normal health parameters [57]. This proactive approach prevents complications, reduce the need for hospitalizations, and ultimately improve patient outcomes.

1.4.2. Proactive Healthcare:

Traditional healthcare often relies on periodic visits to healthcare facilities, which can result in delayed responses to health issues [58]. IOMT devices shift the paradigm by providing continuous data streams. This enables healthcare providers to identify trends and patterns in a patient's health over time. By recognizing subtle changes early on, healthcare professionals can take proactive steps to address potential problems before they escalate, resulting in more effective disease management and improved quality of life for patients [59].

1.4.3. Personalized Treatment:

IOMT devices generate a wealth of data that offers a comprehensive view of an individual's health. This data, combined with advanced analytics, empowers healthcare professionals to create highly personalized treatment plans [60]. For instance, a patient's response to medication, exercise routines, and dietary habits can be analyzed to tailor interventions that are specifically suited to their needs. This customization enhances treatment efficacy, reduces the risk of adverse effects, and increases patient satisfaction with their healthcare journey [61].

1.4.4. Patient Empowerment:

In the era of IOMT, patients are no longer passive recipients of medical care; they become active participants in their health management. The data collected by wearables and other IOMT devices is often accessible to patients, allowing them to track their health metrics, observe trends, and gain insights into how their lifestyle choices impact their well-being [62]. This empowerment encourages individuals to take ownership of their health, make informed decisions, and adopt healthier habits.

1.4.5. Reduced Healthcare Costs:

The integration of IOMT devices in healthcare has the potential to lead to significant cost savings. By facilitating remote monitoring and proactive interventions, these devices can help prevent complications that might lead to costly hospitalizations or emergency room visits [63]. Additionally, the ability to manage chronic conditions more effectively can reduce the need for frequent doctor's appointments and medical interventions, resulting in overall cost reductions for both patients and healthcare systems.

1.4.6. Telemedicine and Remote Consultations:

IOMT devices synergize with the concept of telemedicine, enabling remote consultations between patients and healthcare providers. This is particularly valuable for individuals who live in remote or underserved areas, as well as those with mobility limitations. Through video calls and data sharing, patients can receive expert medical advice without the need to travel long distances, enhancing access to healthcare and bridging geographical barriers [64].

1.4.7. Efficient Data Sharing:

IOMT devices facilitate seamless communication between patients, caregivers, and healthcare professionals. This efficient data sharing ensures that everyone involved in a patient's care journey is well-informed and up to date on the individual's health status [65]. This streamlined communication reduces the chances of misunderstandings, helps coordinate care plans, and improves the overall patient experience.

1.4.8. Early Diagnosis and Prevention:

The continuous monitoring capabilities of IOMT devices can significantly contribute to the early diagnosis and prevention of diseases. By closely tracking health metrics over time, these devices can detect subtle changes that might indicate the onset of a health issue [66]. This early detection allows for timely medical interventions, which can prevent the progression of illnesses and improve the chances of successful treatment outcomes.

1.4.9. Research and Public Health Insights:

Aggregated and anonymized data from IOMT devices have the potential to contribute to medical research and public health initiatives. By analyzing large datasets, researchers can uncover patterns and correlations that provide insights into population health trends, disease prevalence, and the impact of lifestyle factors [67]. This information can guide the development of targeted public health interventions and policies.

1.4.10. Improved Quality of Life:

For individuals managing chronic illnesses, IOMT devices offer the promise of an improved quality of life. By providing continuous monitoring and early intervention, these devices reduce the stress and uncertainty associated with managing health conditions. Patients gain confidence knowing that any changes in their health will be promptly addressed, allowing them to maintain their daily activities and live more fulfilling lives [68].

1.4.11. Data-Driven Decision Making:

Healthcare professionals can make well-informed decisions based on real-time data collected by IOMT devices. This data-driven approach enhances clinical decision-making accuracy [69]. For instance, a doctor can adjust medication dosages based on a patient's recent health trends, leading to more effective treatments and better health outcomes.

1.4.12 Access to Specialist Expertise:

IOMT devices facilitate the sharing of data with specialists for remote consultations. This enables patients and primary care providers to gain access to specialized medical expertise without the need for extensive travel. Patients receive timely guidance from specialists, leading to quicker diagnoses and appropriate treatment plans [70]. While these advantages illustrate the transformative potential of IOMT devices, it's important to acknowledge that challenges such as data security, privacy concerns, interoperability, and equitable access need to be carefully addressed to ensure that the benefits are realized while minimizing risks.

1.5. Challenges associated with IOMT devices:

1.5.1 Data Security and Privacy:

As IOMT devices collect and transmit sensitive health data, the security and privacy of this data become critical concerns. Medical information is highly confidential and must be safeguarded against unauthorized access, breaches, and cyberattacks [71]. The interconnected nature of IOMT devices increases the potential points of vulnerability, necessitating the implementation of robust encryption, authentication mechanisms, and stringent security protocols. Ensuring that patient data remains protected and compliant with data protection regulations is essential to maintain patient trust and prevent potential harm [72].

1.5.2. Interoperability:

IOMT devices are often developed by different manufacturers and may utilize various communication protocols and standards. Achieving seamless interoperability between these devices and existing healthcare systems is a considerable challenge. The ability of devices to exchange and share data effectively across diverse platforms is crucial for healthcare professionals to gain a comprehensive view of a patient's health. Without proper interoperability, the potential insights and benefits offered by IOMT devices may be limited.

1.5.3. Reliability and Accuracy:

The accuracy and reliability of data collected by IOMT devices are of paramount importance, especially when medical decisions are based on this information. Even minor inaccuracies or errors in measurements can lead to incorrect diagnoses, treatment plans, and patient harm [73]. Ensuring that IOMT devices are properly calibrated, validated, and adhere to stringent quality assurance measures is crucial to maintain the trustworthiness of the data they produce.

1.5.4. Regulatory Compliance:

IOMT devices fall under the purview of medical device regulations and data protection laws, which can vary significantly between regions and countries. Adhering to these complex regulatory frameworks poses a challenge for device manufacturers and healthcare providers [74]. Ensuring compliance with multiple sets of regulations while maintaining the safety and effectiveness of the devices requires careful navigation of legal and regulatory landscapes.

1.5.5. Ethical Considerations:

The continuous monitoring capabilities of IOMT devices raise ethical questions about patient consent, data ownership, and potential misuse of sensitive health information. Balancing the benefits of data-driven healthcare with individuals' rights to privacy and autonomy is a complex ethical challenge [75]. Ensuring that patients are fully informed about data usage, providing them with control over their data, and addressing concerns about data security are essential to maintain ethical integrity.

1.5.6. Data Overload and Analysis:

IOMT devices generate vast amounts of data, which can quickly overwhelm healthcare systems. Healthcare professionals may struggle to analyze and interpret this data effectively, leading to data overload. Implementing advanced data analysis techniques, such as machine learning and artificial intelligence, is necessary to sift through the data deluge and extract meaningful insights that can inform clinical decision-making [76].

1.5.7. User Adoption and Training:

The successful integration of IOMT devices relies on users' ability to understand and effectively use these technologies. Both patients and healthcare professionals may require training to navigate the interfaces of these devices, interpret the data they provide, and integrate that information into clinical decision-making [77] . Failing to provide adequate training can lead to mismanagement of health data or misinterpretation of device readings, potentially compromising patient care.

1.5.8. Integration into Clinical Workflows:

Seamlessly integrating IOMT devices into existing clinical workflows presents a challenge. Healthcare professionals effectively incorporate data from these devices into electronic health records (EHRs) and utilize the information to make informed decisions [78]. Ensuring that device data aligns with established protocols and is easily accessed within existing healthcare systems is essential to prevent disruption to clinical routines and to maximize the benefits of IOMT devices.

1.5.9. Lack of Standardization:

The absence of standardized protocols for data transmission, security measures, and interoperability poses a barrier to the widespread adoption of IOMT devices. The lack of uniformity can lead to compatibility issues between different devices and systems, hindering the seamless exchange of data and hindering the potential for widespread implementation [79]. Developing and adhering to industry standards will be essential to ensure consistent and compatible integration of IOMT devices across healthcare environments.

1.5.10. Technical Challenges:

IOMT devices, like any technology, can encounter technical issues such as connectivity problems, sensor malfunctions, and software glitches. These technical challenges can impact the reliability and functionality of the devices, potentially leading to inaccurate data or disruptions in patient monitoring. Addressing technical issues promptly and effectively is crucial to maintain the credibility and utility of IOMT devices in healthcare settings.

Addressing these challenges requires collaborative efforts among healthcare providers, device manufacturers, regulatory bodies, and policymakers. By proactively addressing these issues, the healthcare industry can maximize the benefits of IOMT devices while minimizing risks and ensuring the highest standards of patient care, data security, and ethical integrity.

1.6. Objectives of the study

The objective of the study was to develop the lightweight and secure framework for communication of IoMT devices. The comparative analysis of lightweight ECC framework with other security protocols and cryptographic algorithms was administered. Also, the comparative analysis for communication protocols were conducted and lightweight protocols were considered.

CHAPTER 2: REVIEW OF LITERATURE

Existing lightweight frameworks for protecting IoMT devices have been specifically built to handle the unique problems that come with securing resource-constrained medical devices while still guaranteeing patient data privacy and system integrity. These frameworks have been designed and developed in this manner for the purpose of securing IoMT devices. These frameworks were developed with the express purpose of providing a solution to these difficulties. These frameworks offer a wide variety of cryptographic approaches for securing the end-to-end communication of IoMT devices, making it possible to prevent unauthorized access. Lightweight encryption algorithms and Elliptic Curve Cryptography (also known as ECC) are two examples of these types of methods [80]. In addition to this, they use authentication technologies such as digital signatures and secure key exchange protocols to develop trust between the devices and the servers. This is being done to make the devices and servers more reliable. Message authentication codes, sometimes known as MACs, and hash functions are both utilized by particular frameworks for the aim of preventing tampering and unauthorized access respectively. In addition to this, they make use of protected modes of communication, such as Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS), to maintain the secrecy of the data even when it is being sent from one network to another [81]. In addition, these lightweight frameworks usually place an emphasis on both energy efficiency and computational simplicity as a means of minimizing the impact that the utilization of these frameworks has on the limited resources of IoMT devices. This is done to ensure that the use of these frameworks has as little of an adverse effect as possible. Even though these already-existing solutions have demonstrated some degree of success in improving the safety of IoMT devices, additional investigation and development are required to continuously adjust to newly emerging dangers and guarantee the highest possible level of security for sensitive healthcare information.

2.1. End-to-End Communication in IoMT

The end-to-end principle is a conceptual paradigm in the field of computer networking, which emphasizes that application-specific functionalities, such as dependability and security, should be implemented and managed at the endpoints of the network communication [82]. In networks that adhere to this principle, intermediary nodes, such as gateways and routers, serve the purpose of establishing the network and may employ these mechanisms to enhance efficiency. However, they are unable to ensure end-to-end correctness. The notion was first introduced by Paul Baran in the 1960s, as he sought to address the need for network reliability in situations where the individual components are intrinsically unstable [83]. The underlying concept of the end-to-end principle posits that when two processes engage in communication through a given medium, it is not reasonable to assume that the reliability provided by that medium would precisely match the reliability needs of the processes involved. The end-to-end principle is a design approach that aims to enhance the efficiency and reliability of communication networks. It achieves this by eliminating essential components from intermediary nodes, thereby expanding routing possibilities, optimizing data transmission speeds, and ensuring that application failures occur only in the event of end-point failures [84]. The notion was formulated in response to the necessity for dependable communication in intrinsically volatile settings and has been widely utilized in the majority of networking frameworks for a considerable period of time.

Relevance and Implications for IoMT Devices

The end-to-end communication concept is highly relevant to IoMT devices, which are medical devices that communicate autonomously over a network. Here are some implications of the end-to-end communication concept for IoMT devices:

- IoMT devices can derive advantages from the use of the end-to-end concept, wherein certain elements pertaining to dependability and security are situated within the communicating end nodes of the network [85]. This approach has the potential to enhance network efficiency and establish a safeguard wherein application failures are contingent upon end point failures.
- The use of the end-to-end concept in the context of IoMT networks facilitates the effective management of routing and communication between gateways and end devices [86]. This principle has the potential to enhance data delivery rates and establish a mechanism where intermediary nodes cannot guarantee end-to-end correctness.
- The design of 5G-IoMT is characterized by end-to-end coordination and incorporates automated and intelligent operations at every stage. This integration has the potential to enhance the reliability and efficiency of IoMT networks [17].

- The perception layer of IoMT networks, which comprises data sources like smart objects, health monitoring devices, and mobile apps that are integrated with sensors, can benefit from the end-to-end principle by having intermediary nodes removed to increase routing options and improve data delivery rates [87].
- The application of the end-to-end principle can effectively safeguard the reliability and security of patient data obtained and sent through IoMT devices [88]. This is of utmost importance in ensuring the safety and privacy of patients.

Overall, the end-to-end communication concept is highly relevant to IoMT devices and can help improve the reliability, efficiency, and security of IoMT networks.

2.2. Key Security Requirements for IoMT Devices

The establishment of robust security measures for IoMT devices is of utmost importance in ensuring the safety and privacy of patients. IoMT devices frequently exhibit inadequate security authentication mechanisms, hence rendering them susceptible to cyberattacks. Hence, it is imperative to ascertain that all IoMT devices provide robust security measures in order to safeguard the confidentiality and integrity of data [89]. Healthcare organizations must acquire reliable visibility and categorization of all IoMT devices throughout various components of their infrastructure, including hospital networks, data centers, endpoints, remote clinics, mobile assets, and cloud environments. This technique can assist healthcare IT teams in adopting a prevention-oriented strategy to safeguard medical devices from potential security risks. The establishment of standardized laws and the implementation of a secure system are crucial for ensuring the security of IoMT devices [90]. This measure can effectively guarantee that all IoMT devices adhere to the prescribed minimum-security standards and are adequately safeguarded against any cyber threats. The management of communication between gateways and end devices in IoMT networks is crucial to ensure that intermediary nodes cannot provide a guarantee of end-to-end correctness [91]. This measure can enhance the security of IoMT networks and guarantee the reliability and security of patient data.

Healthcare security executives are required to formulate and execute effective security policies for the IoMT in order to guarantee the security of IoMT equipment. This can encompass the implementation of security measures, including encryption, access control, and intrusion detection and prevention systems. Ensuring the security of IoMT devices is of utmost importance in safeguarding the well-being and confidentiality of patients [92]. In order to safeguard the security of IoMT devices, it is imperative to establish robust authentication controls, acquire reliable visibility and classification of all IoMT devices, establish standardized regulations, oversee communication between gateways and end devices, and formulate and execute effective security strategies for IoMT.

Factors Influencing Lightweight Security Framework Design

Designing a lightweight security framework for IoMT devices requires considering several factors, including:

- **Computational and storage requirements:** The lightweight encryption algorithm used in the security framework should be designed to minimize the computational and storage requirements of IoMT devices [93]. This can help ensure that the security framework does not significantly impact the performance of IoMT devices.
- **Security requirements:** The security architecture must align with the security requisites of IoMT devices, encompassing the preservation of data confidentiality, integrity, and availability [70]. In addition, it is imperative for the framework to provide safeguards against cyberattacks that specifically aim to compromise the security of interconnected medical devices.
- **Detection and classification methods:** Machine learning (ML) has the potential to be utilized in the development of detection and classification techniques for the purpose of identifying cyberattacks that specifically target IoMT equipment within the healthcare industry [94]. Hence, it is imperative to integrate machine learning-based detection and classification techniques into the security framework in order to enhance the security of IoMT devices.
- **Lightweight and secure communication protocol:** The security framework should include a lightweight and secure communication protocol that can be used to transmit data between IoMT devices and other network components [95]. The communication protocol should be designed to minimize the computational and storage requirements of IoMT devices while ensuring the security of data transmission.
- **Authentication and key agreement protocol:** The security framework should incorporate an authentication and key agreement mechanism that is both lightweight and secure [96]. This protocol will serve the purpose of authenticating IoMT devices and establishing secure communication channels. The design of the protocol should prioritize the minimization of computational and storage demands on IoMT devices,

while simultaneously guaranteeing the security of authentication and key agreement processes.

2.3. Current Technologies and Protocols of IoMT

Protocols are crucial to the realization of intelligent operations inside the Internet of Things (IoT) framework. In order to ensure their security, it is important to possess a comprehensive understanding of the protocols. The integration of security measures is an essential aspect of the IoMT, thus necessitating a comprehensive comprehension of the vulnerabilities associated with technologies like Wi-Fi [97]. Numerous protocols have been devised within the realm of the IoMT to effectively execute certain functions and fulfil distinct requirements. RFID technology serves as the forerunner to contemporary IoMT standards. RFID entails the attachment of a chip to an object, which contains pertinent information about said object [98]. Subsequently, an RFID reader transmits a query signal to retrieve this information. The tag is capable of receiving the signal and subsequently transmitting a reflection signal to the reader. The transmission exchange is transmitted to a database for the purpose of verifying the identity of the object. RFID technology is widely regarded as the pioneering machine-to-machine (M2M) communication technology. Although RFIDs continue to serve a purpose, the field of IoMT has made significant advancements by adopting more sophisticated protocols [99].

The constrained application protocol (CoAP) is considered to be a pivotal protocol inside the application layer of Internet of Things (IoT) architecture. The CoAP protocol leverages the principles of Representational State Transfer (REST) to offer its functionality over the Hyper-Text Transfer Protocol (HTTP) [100]. The Representational State Transfer (REST) architecture facilitates streamlined communication between a client and server through the use of HTTP methods such as GET, POST, PUT, and DELETE [101]. The HTTP protocol is a commonly utilized request/response communication protocol on the internet. However, it is typically considered to be excessively thick and power-consuming for IoMT applications. The Constrained Application Protocol (CoAP) was developed by the Constrained RESTful Environments group (CoRE) of the Internet Engineering Task Force (IETF) with the aim of offering a more lightweight alternative to HTTP [102]. The Constrained Application Protocol (CoAP) leverages the request/response mechanism of the

Hypertext Transfer Protocol (HTTP) and transforms it into a more suitable format for the IoMT.

One fundamental distinction between HTTP and CoAP lies in their underlying transport protocols. Specifically, HTTP operates on the Transmission Control Protocol (TCP), whereas CoAP operates on the User Datagram Protocol (UDP). The Transmission Control Protocol (TCP) is a dependable communication protocol that facilitates the establishment of links between a transmitting entity and a receiving entity [103]. The User Datagram Protocol (UDP) is characterized by its lack of reliability and absence of connection establishment between the sender and recipient [104]. In the Transmission Control Protocol (TCP), both the sender and receiver engage in a process of connection establishment before data transmission, and subsequently terminate the connection to ensure the reliable transmission and reception of data.

In the User Datagram Protocol (UDP), the establishment of a connection does not precede the transmission of data, resulting in the possibility of a packet failing to reach its intended destination [105]. Due to this factor, User Datagram Protocol (UDP) is commonly seen as lacking reliability. While TCP is generally considered to be a reliable protocol, it is often seen as slower compared to UDP due to the additional overhead introduced by the handshake procedure. This process involves the confirmation and subsequent closure of the connection. UDP also facilitates the implementation of broadcast and multicast capabilities. Broadcast communication is a method of transmitting messages to all devices inside a network, while multicast communication is a technique that distributes signals to a select group of numerous receivers [106]. There are several reasons why UDP is considered beneficial in the context of the IoMT, mostly due to its simpler and lightweight communication capabilities. Therefore, the Constrained Application Protocol (CoAP) is a highly efficient communication protocol that incorporates the methods of Hypertext Transfer Protocol (HTTP) and minimizes the size of HTTP.

The Advanced Message Queuing Protocol (AMQP) is a message-based application layer protocol that supports both the request/response and publish/subscribe formats. The Advanced Message Queuing Protocol (AMQP) operates on the Transmission Control Protocol (TCP). AMQP is commonly employed in corporate settings due to its robust security measures and dependable performance [107]. The Advanced Message Queuing Protocol (AMQP) utilizes a messaging queue as a means of facilitating effective communication. One further noteworthy protocol at the application layer is the Extensible Messaging and Presence Protocol (XMPP) is a communication protocol that enables the exchange of messages and presence information between entities on a network [107].

The Extensible Messaging and Presence Protocol (XMPP) is characterized by its decentralized nature, allowing it to be compatible with various operating systems. The Extensible Messaging and Presence Protocol (XMPP) facilitates several telecommunication modalities, including multi-party communication, voice communication, and video communication [108]. XMPP distinguishes itself from other protocols by its lack of support for machine-to-machine communication. Instead, numerous XMPP servers establish mutual recognition inside a shared network.

The 6LowPAN protocol is a widely accepted standard for implementing IPv6 over low power wireless personal area networks, specifically designed for networks with limited power and high packet loss, such as wireless sensor networks (WSNs) [109]. IPv6 is a network communication protocol that can be utilized in the context of the IoMT. The utilization of 6LowPAN enables the establishment of wireless sensor networks that are fully based on the Internet Protocol (IP). Each sensor or node inside an IoMT network possesses its unique IPv6 address, enabling direct connectivity to the internet and facilitating the establishment of a mesh network [110]. The system employs AES-128 encryption at the connection layer, while it depends on upper layers to provide end-to-end encryption.

Message queue telemetry transport (MQTT) is an additional significant protocol utilized in the context of the IoMT. MQTT is a protocol that shares similarities with CoAP, however, a notable distinction lies in its utilization of the TCP/IP protocol instead of UDP [111]. CoAP operates on a sender/receiver basis, whereas MQTT utilizes a publisher/subscriber mechanism. The publisher/subscriber paradigm can be distinguished from the sender/receiver model based on the absence of a direct connection between the publisher and the subscriber. The handshake mechanism is absent in this context. Instead, communication between the publisher and subscriber occurs via a broker, who disseminates messages to all potential subscribers of a given topic. The MQTT protocol is a lightweight communication protocol designed for the IoMT that guarantees dependability by implementing quality-of-service levels [112].

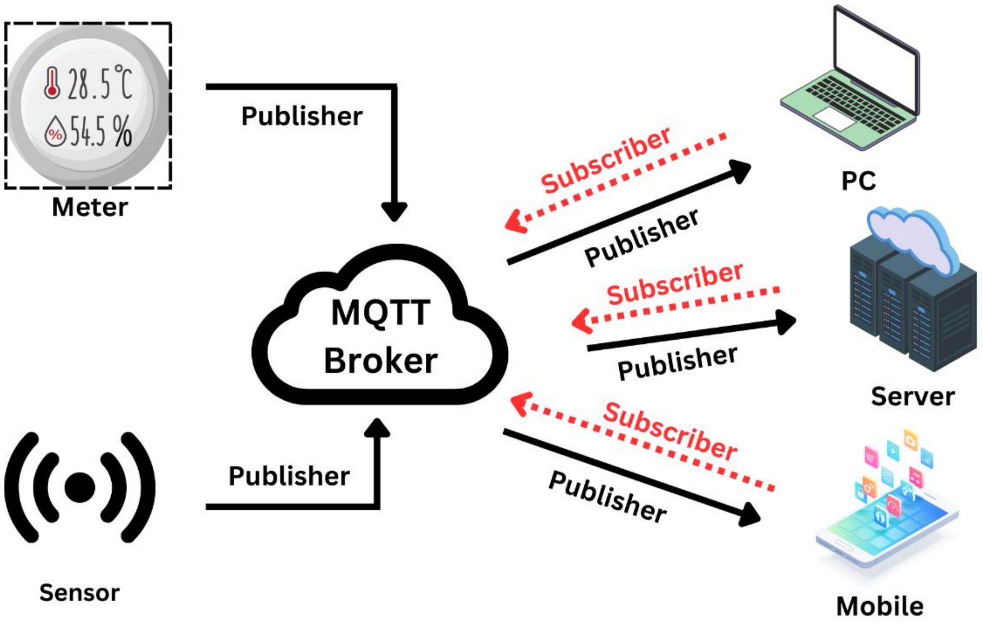


Figure 2.1: Publisher-Subscriber relationship

The aforementioned protocols and standards play a crucial role as fundamental technologies in the realm of the IoMT. The components are essential for enabling the IoMT to operate as an intelligent communication system. Gaining a comprehensive comprehension of the protocols and their operational mechanisms facilitates the strategic planning of many facets within the IoMT ecosystem, with a particular emphasis on ensuring robust security measures [113]. There exist numerous additional protocols and standards within the realm of the IoMT; nevertheless, the aforementioned protocols and standards are widely recognized and significant. The presence of diverse protocols poses a challenge to achieving interoperability within an IoMT ecosystem. The establishment of secure communication between devices is of utmost importance. Bluetooth technology is widely recognized and used as a standard for the IoMT. Bluetooth technology is an integral component of nearly all smart devices, facilitating efficient and seamless communication [114]. Bluetooth plays a crucial role in the IoMT due to its extensive utilization in various applications.

2.4. Comparative Analysis of Lightweight Frameworks

These protocols serve different purposes and have their own strengths and weaknesses, so the choice between them depends on the specific requirements of your application [115].

Table 2.1: Comparative analysis of lightweight framework protocols

Criteria	**MQTT**	**CoAP**	**AMQP**	**XMPP**
Purpose	Publish-Subscribe	Resource-Oriented	Message Queuing	Instant Messaging
Transport Layer	TCP	UDP/DTLS	TCP	XML over TCP/HTTP
Message Format	Binary	Binary/Text	Binary	XML
Message Size	Small to Medium	Small to Medium	Small to Large	Small to Large
QoS Support	0, 1, 2	0, 1, 2	Guaranteed	Presence
Security	SSL/TLS	DTLS	SSL/TLS	TLS
Scalability	High	Medium-High	High	Medium
Complexity	Low	Low	Medium-High	Medium-High
Pub-Sub Model	Yes	No	No	Yes
Request-Response	No	Yes	Yes	Yes
Resource Discovery	No	Yes	No	Yes
Use Cases	IoT, Real-time	IoT, RESTful APIs	Enterprise Messaging, IoT	Instant Messaging
Standardization	OASIS	IETF	OASIS	XMPP Standards Foundation
Open Source	Yes	Yes	Yes	Yes

2.5. Strengths and Limitations of Each Framework

Power Consumption:

- **MQTT:** MQTT is designed to be lightweight, and its power consumption is relatively low, making it suitable for resource-constrained devices in IoT applications.
- **CoAP:** CoAP is also designed to be efficient and consumes low to moderate power, making it suitable for IoT devices with limited power resources.
- **AMQP:** AMQP can be more power-intensive due to its advanced features and larger message payloads, making it less suitable for extremely power-constrained devices.
- **XMPP:** XMPP's power consumption is moderate and generally acceptable for devices with reasonable power resources, but it may not be the best choice for extremely low-power devices.

Speed:

- **MQTT:** MQTT is known for its speed and efficiency, making it suitable for real-time communication and IoT applications where low latency is crucial.
- **CoAP:** CoAP is designed for speed and is well-suited for IoT applications that require fast communication between devices and servers.
- **AMQP:** AMQP offers moderate to fast speed, making it suitable for enterprise messaging systems but may not be as fast as MQTT or CoAP for certain real-time use cases.
- **XMPP:** XMPP offers moderate speed and is well-suited for instant messaging and presence management, but it may not be as fast as MQTT or CoAP for other types of applications.

Lightweight:

- **MQTT:** MQTT is considered lightweight and is suitable for resource-constrained devices and environments.
- **CoAP:** CoAP is designed to be lightweight and efficient, making it suitable for IoT applications.

- **AMQP:** AMQP is a heavyweight compared to MQTT and CoAP, which may limit its suitability for resource-constrained environments.
- **XMPP:** XMPP is moderate in terms of lightweightness and may be more suitable for devices with reasonable computing resources.

Security:

- **MQTT:** MQTT offers good security features, including SSL/TLS encryption and authentication mechanisms, making it suitable for secure IoT and messaging applications.
- **CoAP:** CoAP also provides good security features, including DTLS for secure communication, making it suitable for secure IoT deployments.
- **AMQP:** AMQP offers excellent security features, including strong authentication and access control, making it well-suited for secure enterprise messaging.
- **XMPP:** XMPP provides good security features, including TLS encryption, but may require additional extensions for specific security requirements.

Table 2.2 Comparison of lightweight protocols over power consumption, speed and security

Criteria	**MQTT**	**CoAP**	**AMQP**	**XMPP**
Power Consumption	Low to Moderate	Low to Moderate	Moderate to High	Moderate
Speed	Fast	Fast	Moderate to Fast	Moderate
Lightweight	Lightweight	Lightweight	Moderate	Moderate
Security	Good	Good	Excellent	Good

2.6. IoMT Systems Architecture

Most of the current IoMT systems are typically divided into four layers, as shown in Fig. [95]. These layers include all data stages starting from the individual’s biometric collection stage and ending in data storage and subsequent visualization by a physician for analysis. Moreover, the patient can also visualize their overall health status from the cloud. With the current advances in IMDS, IoWDs, and IMDs mostly share the same architecture, given that IMDs can communicate with the gateways, as exemplified by Medtronic peacemaker.

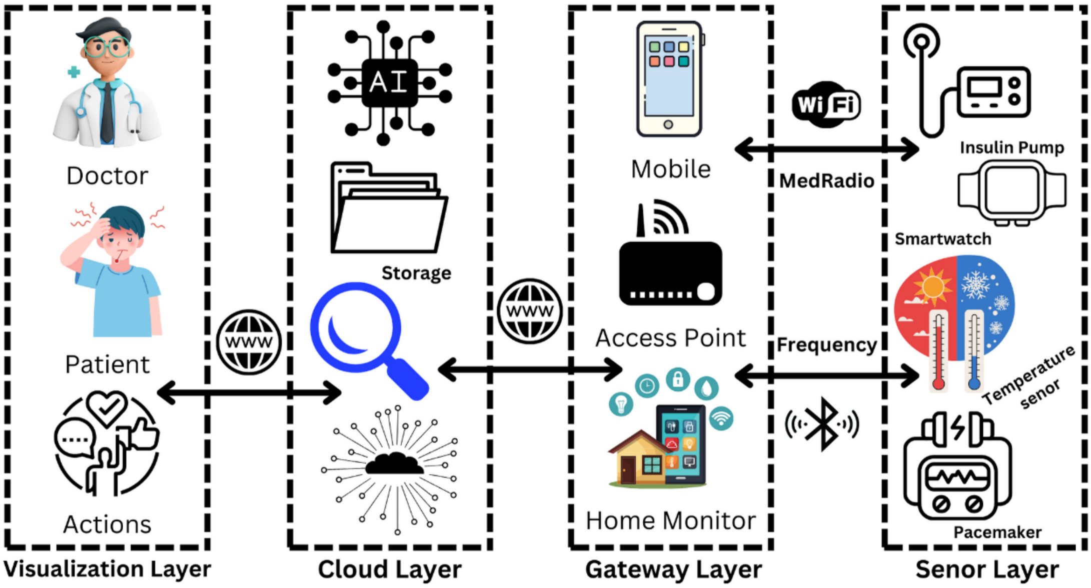


Figure 2.2: Architecture of IoMT system

1) **Sensor Layer**: This layer consists of a set of small implanted or worn sensors that collect the patient's biometrics. The data are transmitted to the second layer over wireless protocols, such as Wi-Fi, Bluetooth, or over MedRadio frequency (RF) spectrum reserved for IMDs [116].

2) **Gateway Layer**: Due to the processing and storage limitations of IoMT sensors, the data are transferred without processing to the second layer, i.e., the gateway layer [67]. The devices in this layer can be the patient's smartphone or a dedicated access point (AP), which are generally more powerful than sensors. They can perform some preprocessing operations, such as validation, short-term data storage, and simple AI-based analysis. In addition, they send the sensor data to the cloud over the Internet.

3) **Cloud Layer**: The cloud layer is responsible for getting the data from the gateway for storage, analysis, and secure access. The analysis includes data processing to find any changes in the patient's health and presenting them to the physicians or patients for further actions [117]. The key generation server (KGS) is responsible for generating IDs and keys for various system nodes. The access to the sensors can be remotely managed and controlled from this layer.

4) **Visualization/Action Layer**: In this layer, the data are presented to the physicians and the patients to track their health. This layer also includes the actions recommended by the physician based on the patient's health conditions. Examples of actions include prescribing or adjusting the dosage for various medicines.

Lack of visibility into endpoints and architecture is a major gap for lightweight framework devices. With respect to security, each component of the lightweight framework was focused to make effective end to end communication. Esp32 microcontroller was used to collect the IoMT sensor's data and publish it to MQTT broker, Grafana was used for data visualization. Prometheus was used for data scrapping and alerting.

CHAPTER 3: METHODOLOGY AND COMPONENTS

This chapter presents an overview of the approach utilized and the software tools selected to achieve the desired solution in this thesis. The methodology describes the systematic technique used to achieve the research objectives, while the selection of software tools was thorough and based on their compatibility with the chosen methodology.

At the core of every lightweight framework device lies a carefully selected set of components that collectively enable its functionality and resource-efficient operation [118]. These components must be meticulously chosen and optimized to meet the specific demands of the device's intended application, often involving considerations such as processing power, memory, energy efficiency, and connectivity. The major components used in the research work are described as follows:

3.1. Microcontroller (MCU):

A microcontroller is a foundational component within the Internet of Things (IoT) ecosystem, enabling the functionality of numerous IoT devices. These devices are designed to acquire data from their surroundings, process that data, and often communicate it to other devices or cloud-based platforms for further analysis and action. Microcontrollers play a pivotal role in facilitating these essential functions. They are equipped with a variety of sensors and interfaces, allowing them to gather data, such as environmental conditions or device status.

The data acquisition procedure involved the utilization of the ESP32 microcontroller to gather information from the DHT11 sensor. Subsequently, the collected data was subjected to processing, enabling its transmission to a user interface. This interface facilitated the monitoring and analysis of the acquired data. This technology facilitates the ability to monitor devices and systems in real-time, control them remotely, and automate their operations. As a result, it plays a crucial role in the development of intelligent and interconnected solutions within the Internet of Things (IoT) framework. Once collected, microcontrollers employ their built-in processing capabilities, which include a central processing unit (CPU), memory, and input/output (I/O) ports, to analyze and make decisions based on the data. Furthermore, they possess communication modules, such as Wi-Fi, Bluetooth, or cellular connectivity, to securely transmit data to other devices or the cloud. Microcontrollers are also designed with energy efficiency in mind, ensuring IoT devices can conserve power and operate on limited energy sources. Additionally, they incorporate security features, enabling data protection and device integrity. Their real-time operation capability suits IoT applications requiring instantaneous responses, and their cost-effectiveness makes them ideal for mass-produced IoT devices. Customizable through firmware, microcontrollers offer flexibility, and they empower edge computing by enabling local data processing, reducing latency, and enhancing system responsiveness. In summary, microcontrollers are the essential workhorses at the core of IoT devices, bringing data collection, processing, and communication together to enable diverse IoT applications across industries. The following are the ESP32's key security features:

- All IEEE 802.11 standard security features, including WPA, WPA2, WPA3 (depending on version), and WLAN Authentication and Privacy Infrastructure (WAPI), are supported.
- Safe boot
- Encryption of flash memory
- AES, SHA-2, RSA, elliptic curve cryptography (ECC), random number generator (RNG) cryptographic hardware acceleration

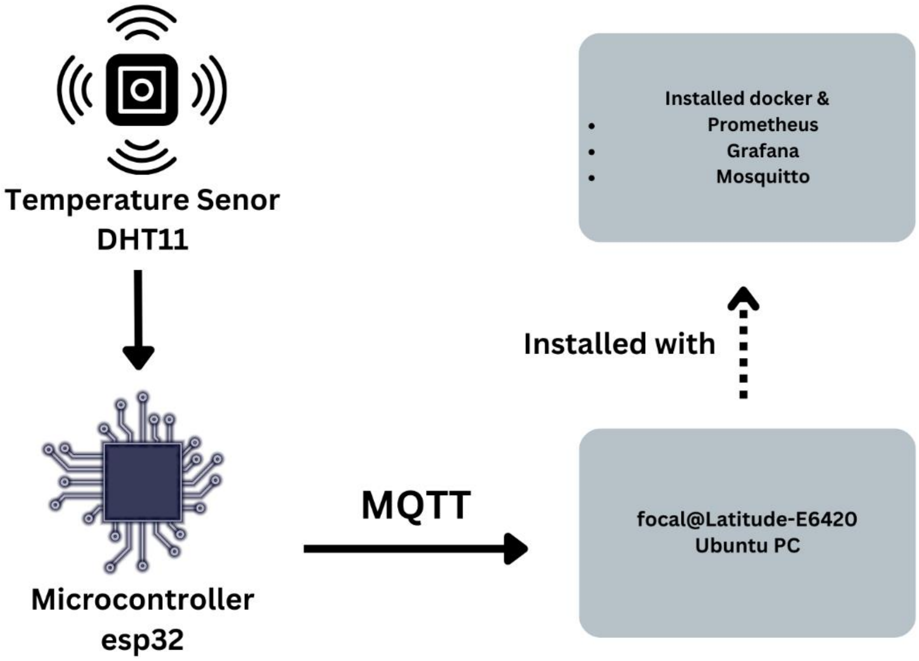


Figure 3.1: Methodology for lightweight framework in IoMT

3.2 Sensors:

Sensors play a pivotal role as a core component of lightweight framework devices, contributing to the device's ability to interact with and gather data from the physical world. These devices are often deployed in various applications, such as IoT (Internet of Things), where they need to monitor, measure, and respond to changes in their environment [115]. Sensors serve as the primary data acquisition units for lightweight framework devices, detecting and measuring physical phenomena such as temperature, humidity, light levels, motion, pressure, proximity, and more, depending on the device's purpose. They are transducers that convert physical properties into electrical signals, allowing the device to interpret and process this information. Sensors provide lightweight framework devices with a degree of environmental awareness, enabling them to respond to changing conditions, make informed decisions, or trigger actions [119]. The monitoring and control of temperature and humidity are crucial components in ensuring the maintenance of a safe and healthy environment within hospital facilities. The efficacy of medical equipment, medication, and patient recovery rates can be compromised by variations in temperature and humidity levels. Temperature and humidity sensors play a crucial role in this context. The DHT11 sensor, which measures temperature and humidity, was employed in this study to fulfil the research objectives.

3.3 MQTT Broker:

An MQTT broker is a crucial component in MQTT-based messaging systems, facilitating communication between MQTT clients in a publish-subscribe model. One of the prominent MQTT brokers in this domain is the Mosquitto broker that was used in the current study. Mosquitto is known for its efficiency and lightweight design, making it ideal for resource constrained IoT devices and applications. Mosquitto offers robust security features like authentication, TLS/SSL encryption, and access control lists (ACLs) to safeguard MQTT communications. It manages Quality of Service (QoS) levels, ensuring message delivery reliability, and offers session persistence to store retained messages and maintain client state.

3.4 User Interface (UI)

Grafana was used following multiple steps. Subsequently, efforts were made to integrate data sources with Grafana, encompassing medical sensors. Designing the dashboard was a pivotal step, with a focus on creating clear and intuitive displays of sensor data, utilizing Grafana's drag-and-drop interface for visualizations like charts, graphs, and gauges. The interface's design prioritized effective data representation for healthcare professionals or patients. Alerting rules were established to provide timely notifications to relevant parties when specific medical conditions or thresholds were met. Security measures were implemented to control access to patient data through user authentication and authorization features within Grafana. Usability testing was conducted to gather feedback from healthcare professionals or patients, leading to adjustments to enhance functionality and user experience. Ensuring scalability and performance optimization was crucial to handle growing data volumes and concurrent users while maintaining real-time updates and responsiveness. Extensive documentation and training materials were created to aid users and administrators in effectively utilizing the Grafana-based interface. Continuous monitoring and maintenance activities were carried out to address issues, update data sources, and incorporate new features or improvements, ensuring the interface's reliability and relevance in supporting healthcare outcomes and patient care.

3.5 Lightweight Protocol

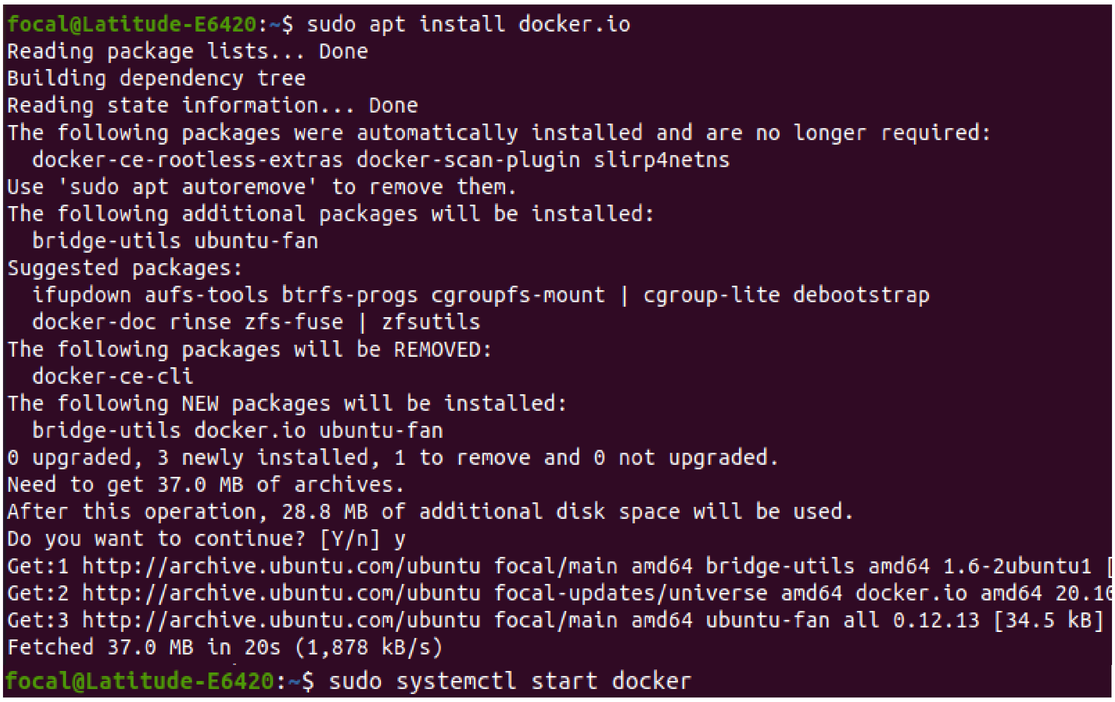
Networking and Protocols stand as indispensable components within lightweight framework devices, enabling these devices to establish vital connections and communicate with other devices, networks, or cloud services. Their role in IoMT and embedded systems is pivotal, as they facilitate the seamless exchange of data, remote control, and connectivity across a wide array of applications. These components permit lightweight devices to transmit sensor readings, status updates, control commands, and other critical information to other devices or systems, often via wireless technologies like Wi-Fi, Bluetooth, Zigbee, LoRa, or cellular connections. MQTT protocol was employed to format and transmit data over the network, ensuring compatibility and interoperability within the IoMT ecosystem [120].

CHAPTER 4: FRAMEWORK IMPLEMENTATION AND RESULTS

4.1. System Readiness:

4.1.1. Installation of dockers and containers

For the system readiness there was necessary to install the docker because docker is used for containerization. Containers are lightweight, portable and isolated environments that can run consistently on different computing environments. Containers provide a level of isolation, enhancing security by containing applications and preventing interference between them.

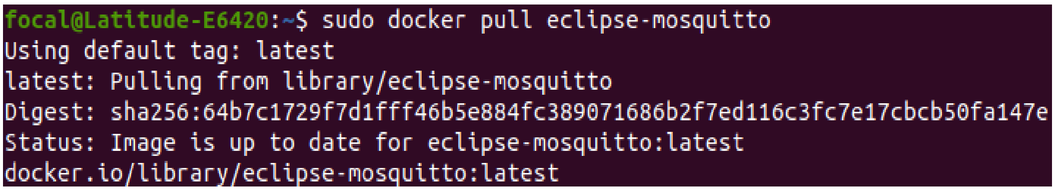


```
focal@Latitude-E6420:~$ sudo apt install docker.io
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  docker-ce-rootless-extras docker-scan-plugin slirp4netns
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  bridge-utils ubuntu-fan
Suggested packages:
  ifupdown aufs-tools btrfs-progs cgroupfs-mount | cgroup-lite debootstrap
  docker-doc rinse zfs-fuse | zfsutils
The following packages will be REMOVED:
  docker-ce-cli
The following NEW packages will be installed:
  bridge-utils docker.io ubuntu-fan
0 upgraded, 3 newly installed, 1 to remove and 0 not upgraded.
Need to get 37.0 MB of archives.
After this operation, 28.8 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://archive.ubuntu.com/ubuntu focal/main amd64 bridge-utils amd64 1.6-2ubuntu1 [
Get:2 http://archive.ubuntu.com/ubuntu focal-updates/universe amd64 docker.io amd64 20.10
Get:3 http://archive.ubuntu.com/ubuntu focal/main amd64 ubuntu-fan all 0.12.13 [34.5 kB]
Fetched 37.0 MB in 20s (1,878 kB/s)
focal@Latitude-E6420:~$ sudo systemctl start docker
```

Figure 4.1: The installation of dockers in Linux platform

4.1.2. Installation of MQTT broker

To exchange messages between MQTT clients (publishers and subscribers) a MQTT broker was installed. The MQTT broker receives published messages from clients and then distributes them to the appropriate subscribers based on topic subscriptions.



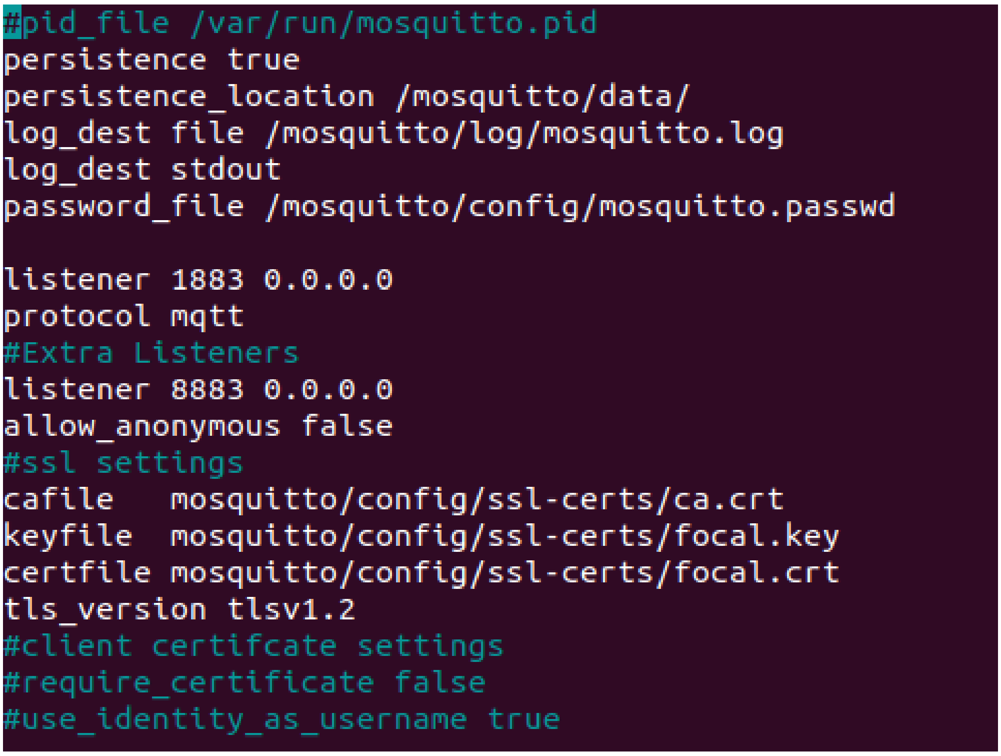
```
focal@Latitude-E6420:~$ sudo docker pull eclipse-mosquitto
Using default tag: latest
latest: Pulling from library/eclipse-mosquitto
Digest: sha256:64b7c1729f7d1fff46b5e884fc389071686b2f7ed116c3fc7e17cbcb50fa147e
Status: Image is up to date for eclipse-mosquitto:latest
docker.io/library/eclipse-mosquitto:latest
```

Figure 4.2: Shows that how to pull latest image to eclipse mosquitto

Creations of directory was must to share the information between host and docker container. To serve this purpose the following commands were used to make the directories for configuration, data, and logs.

- mkdir mosquitto
- mkdir mosquitto\config
- mkdir mosquitto\data
- mkdir mosquitto\log

After creation of mosquitto.config file the configuration setting needs to be changed. for setting up the listener port at 1883 which is default port of mqqt. To implement security port 8883 was configured in the file. Port 8883 is also used for mqtt communication with encryption enabled. When an MQTT broker is configured to listen on port 8883, it means that it is set up to accept incoming MQTT client connections that are secured using TLS/SSL.



```
#pid_file /var/run/mosquitto.pid
persistence true
persistence_location /mosquitto/data/
log_dest file /mosquitto/log/mosquitto.log
log_dest stdout
password_file /mosquitto/config/mosquitto.passwd

listener 1883 0.0.0.0
protocol mqtt
#Extra Listeners
listener 8883 0.0.0.0
allow_anonymous false
#ssl settings
cafile   mosquitto/config/ssl-certs/ca.crt
keyfile  mosquitto/config/ssl-certs/focal.key
certfile mosquitto/config/ssl-certs/focal.crt
tls_version tlsv1.2
#client certifcate settings
#require_certificate false
#use_identity_as_username true
```

Figure 4.3 Shows the configuration settings of mqtt broker

- Change permission of the mosquitto configuration folder by typing the following command in the terminal.

  sudo chown -R 1883:1883 ~/mosquitto

- Create the Mosquitto container.

  sudo docker run -it -p 1883:1883 --name mosquitto -v ~/mosquitoconfig -v ~ /mosquitto/data:/mosquitto/data -v ~/mosquitto/log:/mosquitto/log eclipsemosquitto

- Set up a username and password for Mosquitto. Access the shell of the container using the following command:

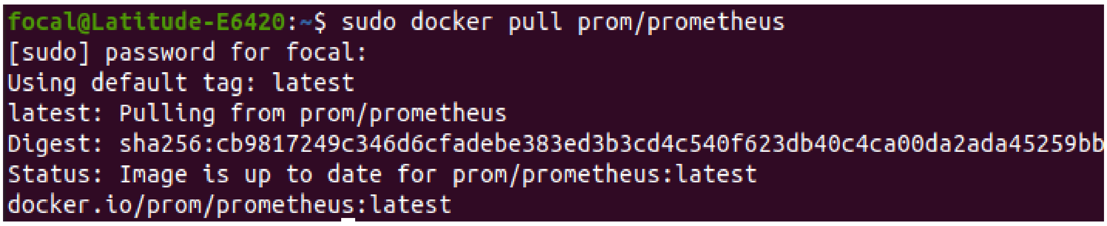
  sudo docker exec -it 933fcf359ae9 sh

- The following command was used to set the username and password.

  mosquitto_passwd -c /mosquitto/config/mosquitto.passwd admin

4.1.3. Prometheus container

To scrap metrics and data from sensor Prometheus was pulled. It has the capability to collect and stores time series data. Prometheus was used for alerting mechanism that was configured to send notification when predefined conditions are met, or thresholds are exceed.



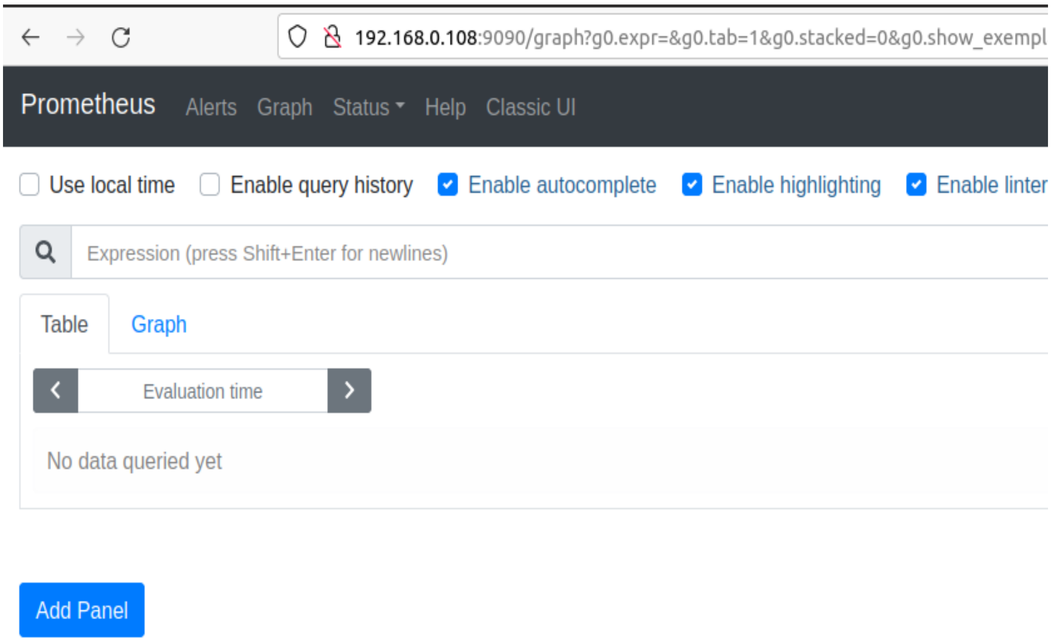
```
focal@Latitude-E6420:~$ sudo docker pull prom/prometheus
[sudo] password for focal:
Using default tag: latest
latest: Pulling from prom/prometheus
Digest: sha256:cb9817249c346d6cfadebe383ed3b3cd4c540f623db40c4ca00da2ada45259bb
Status: Image is up to date for prom/prometheus:latest
docker.io/prom/prometheus:latest
```

Figure 4.4: Shows that how to pull Prometheus with its digest

Following command was run in the terminal:

sudo docker run -itd -v "$(pwd)/:/etc/prometheus -p 9090:9090 prom/prometheus

The purpose of this Docker command was to run a Prometheus container with the specified options, ensuring that it has access to a local directory for configuration (-v), and exposes port 9090 for accessing the Prometheus UI (-p).



192.168.0.108:9090/graph?g0.expr=&g0.tab=1&g0.stacked=0&g0.show_exempl

□ Use local time □ Enable query history ☑ Enable autocomplete ☑ Enable highlighting ☑ Enable linter

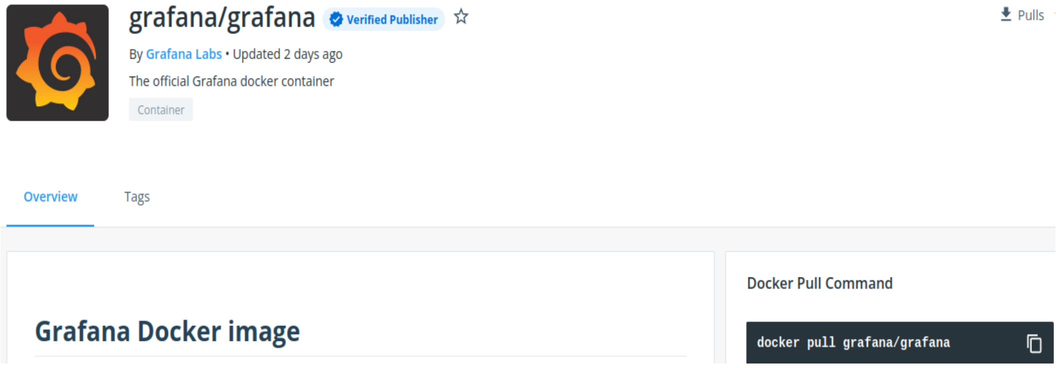
Expression (press Shift+Enter for newlines)

No data queried yet

Figure 4.5: Shows that the access of Prometheus interface

4.1.4. Grafana Container (Analytics and interactive visualization web application)

To visualize the data from our sensor to the dashboard in the form of graphs for better visualization, Grafana was used. Which is open-source platform run using containerization.



grafana/grafana

By Grafana Labs • Updated 2 days ago

The official Grafana docker container

Grafana Docker image

Docker Pull Command

```
docker pull grafana/grafana
```

Figure 4.6: The official image of grafana

Following commands were used to make the directory of grafana and change the permission for a specific group.

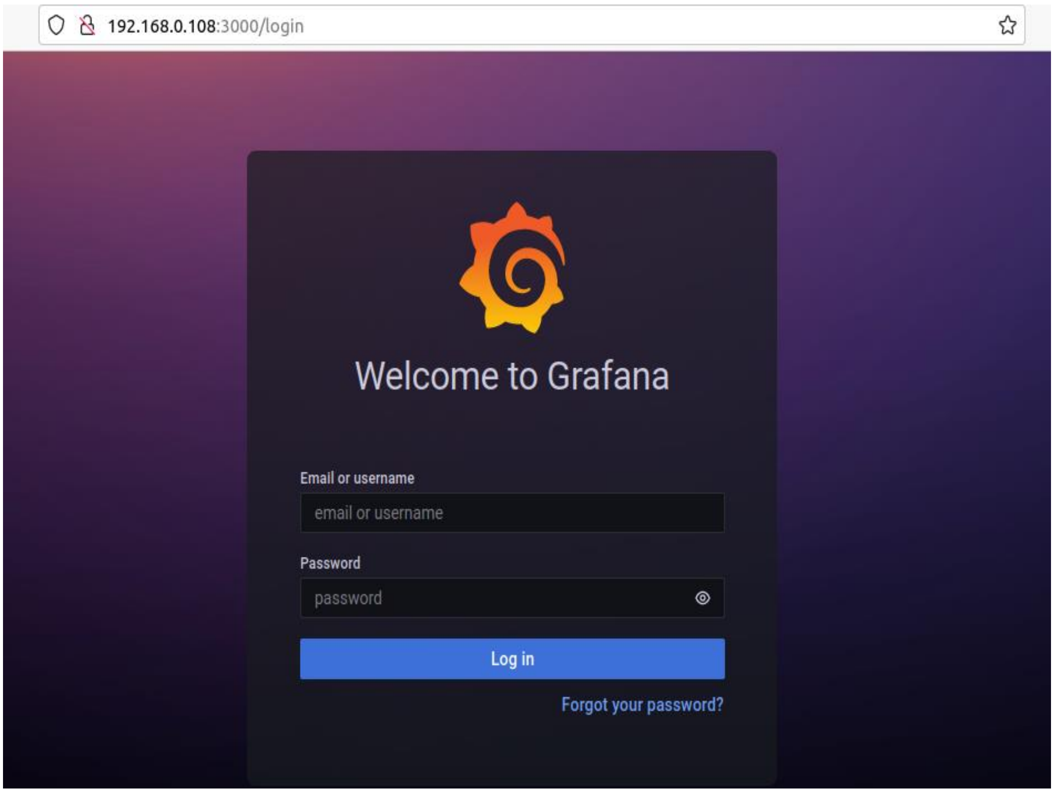
mkdir /var/lib/grafana -p

chown -R 472:472 /var/lib/grafana

Create the Grafana container by typing the following command in the terminal.

```
sudo docker run -d -p 3000:3000 -v /var/lib/grafana:/var/lib/grafana
 -e                                   "GF_SECURITY_ADMIN_PASSWORD
=Bekh@m0987"grafana/Grafana
```

To open Grafana type http://192.168.0.108:3000 (change your Ip as per configuration)



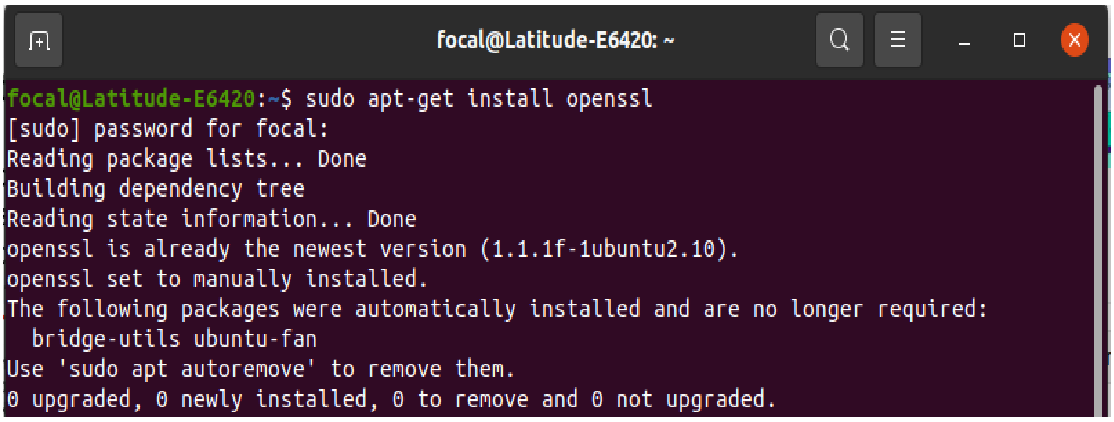
Welcome to Grafana

Figure 4.7: Grafana login page

4.2. Securing the communication with lightweight cryptographic algorithm:

4.2.1. Integration TLS/SSL certificates with mosquitto MQTT broker and ESp32

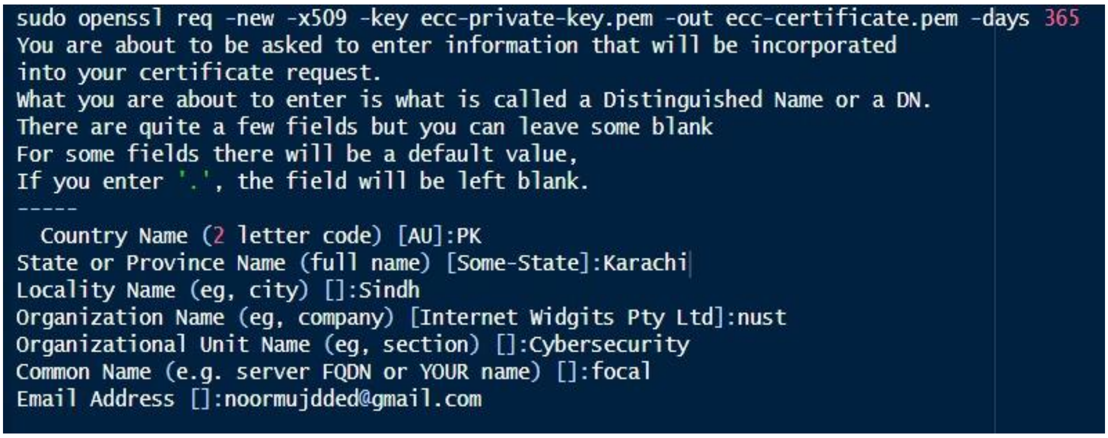
Integrating TLS/SSL certificates with the Mosquitto MQTT broker and ESP32 (a popular microcontroller for IoT projects) involves configuring both the MQTT broker and the ESP32 to use secure connections. TLS/SSL provides encryption and authentication to secure MQTT communication.



```
focal@Latitude-E6420:~$ sudo apt-get install openssl
[sudo] password for focal:
Reading package lists... Done
Building dependency tree
Reading state information... Done
openssl is already the newest version (1.1.1f-1ubuntu2.10).
openssl set to manually installed.
The following packages were automatically installed and are no longer required:
  bridge-utils ubuntu-fan
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
```

Figure 4.8: Shows the installation process of openssl

Now to secure the communication the procedure of Installation and configuration of server.crt for server certificate and key was performed.



```
sudo openssl req -new -x509 -key ecc-private-key.pem -out ecc-certificate.pem -days 365
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
  Country Name (2 letter code) [AU]:PK
State or Province Name (full name) [Some-State]:Karachi
Locality Name (eg, city) []:Sindh
Organization Name (eg, company) [Internet Widgits Pty Ltd]:nust
Organizational Unit Name (eg, section) []:Cybersecurity
Common Name (e.g. server FQDN or YOUR name) []:focal
Email Address []:noormujdded@gmail.com
```

Figure 4.9: Configuration of TLS/SSL certificate using ECC

After Installation and configuration on server side the client-side configuration was performed (client.crt) for client certificate and key for connect with external publishers like (mqtt explorer)

Sudo openssl req -new -out client.csr -key client.key

You are about to be asked to enter information that will be incorporated into your certificate request. What you are about to enter is what is called a Distinguished Name or a DN. There are quite a few fields but you can leave some blank For some fields there will be a default value, If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:PK

State or Province Name (full name) [Some-State]:Karachi

Locality Name (eg, city) []:Sindh

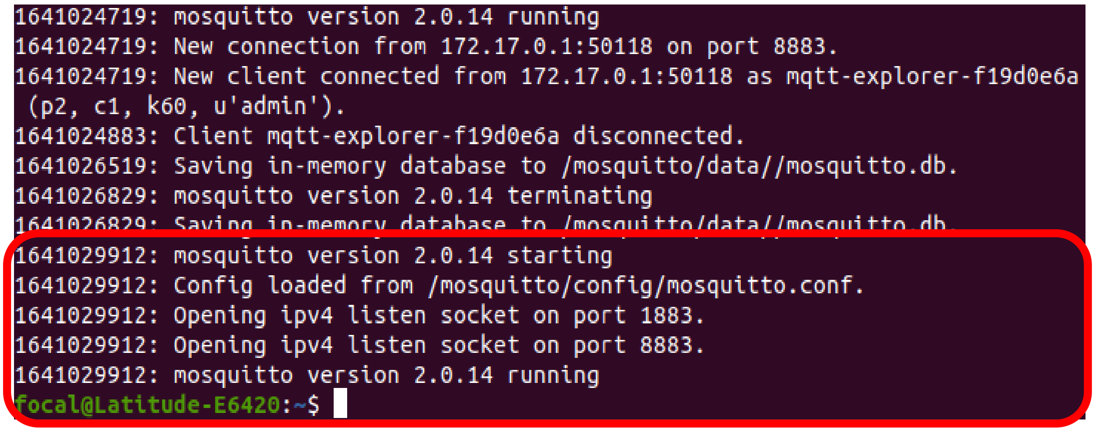
Organization Name (eg, company) [Internet Widgits Pty Ltd]:nust

Organizational Unit Name (eg, section) []:Cybersecurity

Common Name (e.g. server FQDN or YOUR name) []:focal

Email Address []:noormujdded@gmail.com

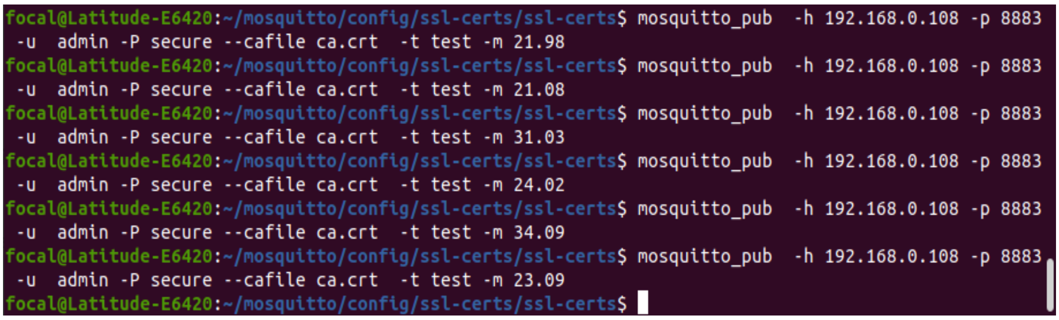
A. Then, we checked all configured certificates in mosquitto directory.



```
1641024719: mosquitto version 2.0.14 running
1641024719: New connection from 172.17.0.1:50118 on port 8883.
1641024719: New client connected from 172.17.0.1:50118 as mqtt-explorer-f19d0e6a
 (p2, c1, k60, u'admin').
1641024883: Client mqtt-explorer-f19d0e6a disconnected.
1641026519: Saving in-memory database to /mosquitto/data//mosquitto.db.
1641026829: mosquitto version 2.0.14 terminating
1641026829: Saving in-memory database to /mosquitto/data//mosquitto.db.
1641029912: mosquitto version 2.0.14 starting
1641029912: Config loaded from /mosquitto/config/mosquitto.conf.
1641029912: Opening ipv4 listen socket on port 1883.
1641029912: Opening ipv4 listen socket on port 8883.
1641029912: mosquitto version 2.0.14 running
focal@Latitude-E6420:~$
```

Figure 4.10: The docker logs of TLS configuration on port 8883

MQTT (Message Queuing Telemetry Transport) operates on a publish/subscribe (pub/sub) messaging model. This messaging model is a communication pattern used in distributed systems and IoT (Internet of Things) applications.



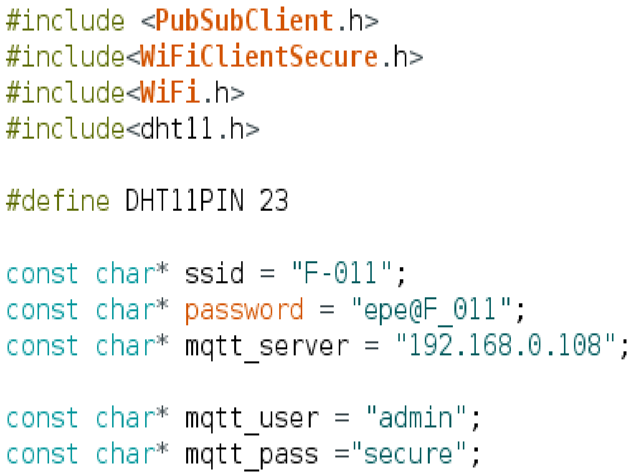
```
focal@Latitude-E6420:~/mosquitto/config/ssl-certs/ssl-certs$ mosquitto_pub  -h 192.168.0.108 -p 8883
 -u  admin -P secure --cafile ca.crt  -t test -m 21.98
focal@Latitude-E6420:~/mosquitto/config/ssl-certs/ssl-certs$ mosquitto_pub  -h 192.168.0.108 -p 8883
 -u  admin -P secure --cafile ca.crt  -t test -m 21.08
focal@Latitude-E6420:~/mosquitto/config/ssl-certs/ssl-certs$ mosquitto_pub  -h 192.168.0.108 -p 8883
 -u  admin -P secure --cafile ca.crt  -t test -m 31.03
focal@Latitude-E6420:~/mosquitto/config/ssl-certs/ssl-certs$ mosquitto_pub  -h 192.168.0.108 -p 8883
 -u  admin -P secure --cafile ca.crt  -t test -m 24.02
focal@Latitude-E6420:~/mosquitto/config/ssl-certs/ssl-certs$ mosquitto_pub  -h 192.168.0.108 -p 8883
 -u  admin -P secure --cafile ca.crt  -t test -m 34.09
focal@Latitude-E6420:~/mosquitto/config/ssl-certs/ssl-certs$ mosquitto_pub  -h 192.168.0.108 -p 8883
 -u  admin -P secure --cafile ca.crt  -t test -m 23.09
focal@Latitude-E6420:~/mosquitto/config/ssl-certs/ssl-certs$
```

Figure 4.11: The publishing and subscribing of Topics & messages on TLS/SSL encryption ports

4.2.2. Setting up TLS/SSL setting on ESP32 Arduino IDE

To establish secure connections to MQTT brokers using SSL/TLS encryption The PubSubClient and WiFiClientSecure libraries were used in Arduino for ESP32.These libraries simplify the process of connecting to MQTT brokers that require secure communication and the use of SSL certificates.

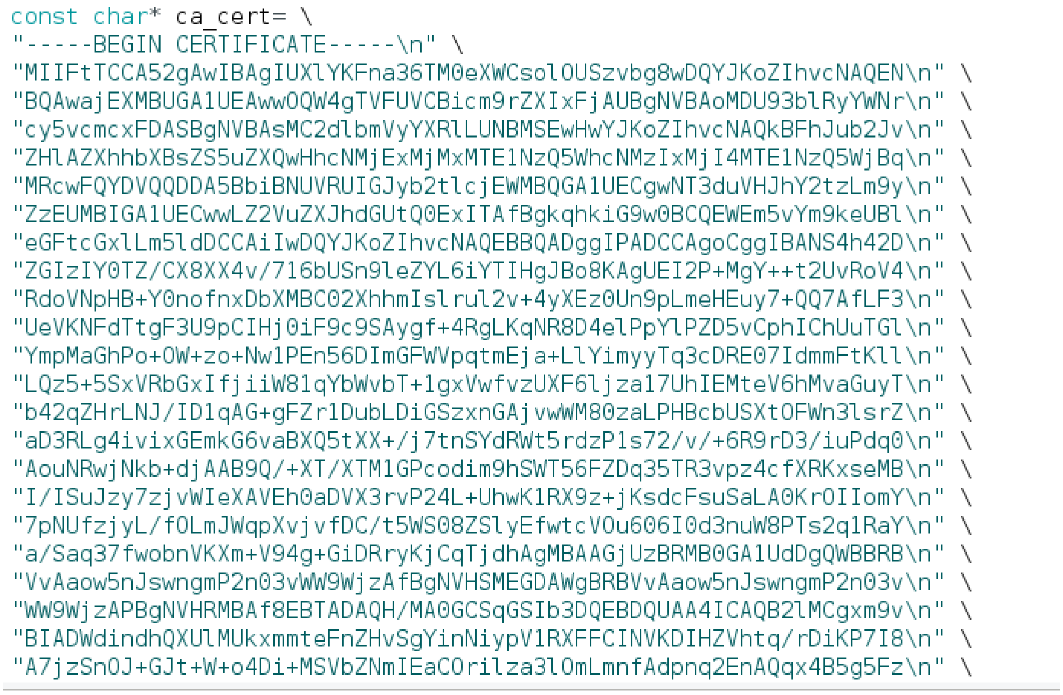
Figure (4.12) shows that the use of pubsub client and wificlientsecure library for mqtt broker and ssl certificate.



```
#include <PubSubClient.h>
#include<WiFiClientSecure.h>
#include<WiFi.h>
#include<dht11.h>

#define DHT11PIN 23

const char* ssid = "F-011";
const char* password = "epe@F_011";
const char* mqtt_server = "192.168.0.108";

const char* mqtt_user = "admin";
const char* mqtt_pass ="secure";
```

Figure 4.12: Adding libraries in microcontroller

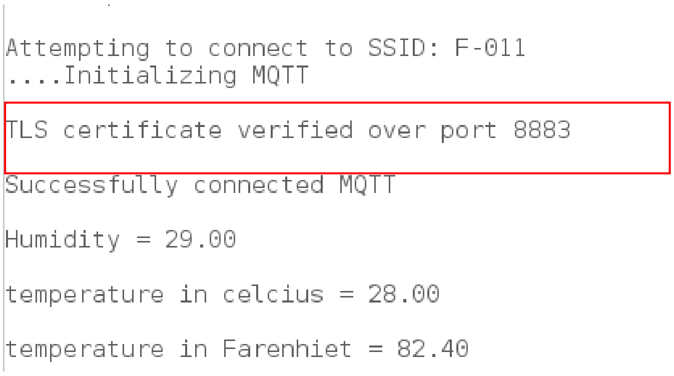
Set certificates in ca_cert declaration.



```
const char* ca_cert= \
"-----BEGIN CERTIFICATE-----\n" \
"MIIFtTCCA52gAwIBAgIUXlYKFna36TM0eXWCsolOUSzvbg8wDQYJKoZIhvcNAQEN\n" \
"BQAwajEXMBUGA1UEAwwOQW4gTVFUVCBicm9rZXIxFjAUBgNVBAoMDU93blRyYWNr\n" \
"cy5vcmcxFDASBgNVBAsMC2dlbmVyYXRlLUNBMSEwHwYJKoZIhvcNAQkBFhJub2Jv\n" \
"ZHlAZXhhbXBsZS5uZXQwHhcNMjExMjMxMTE1NzQ5WhcNMzIxMjI4MTE1NzQ5WjBq\n" \
"MRcwFQYDVQQDDA5BbiBNUVRUIGJyb2tlcjEWMBQGA1UECgwNT3duVHJhY2tzLm9y\n" \
"ZzEUMBIGA1UECwwLZ2VuZXJhdGUtQ0ExITAfBgkqhkiG9w0BCQEWEm5vYm9keUBl\n" \
"eGFtcGxlLm5ldDCCAiIwDQYJKoZIhvcNAQEBBQADggIPADCCAgoCggIBANS4h42D\n" \
"ZGIzIY0TZ/CX8XX4v/716bUSn9leZYL6iYTIHgJBo8KAgUEI2P+MgY++t2UvRoV4\n" \
"RdoVNpHB+Y0nofnxDbXMBC02XhhmIslrul2v+4yXEz0Un9pLmeHEuy7+QQ7AfLF3\n" \
"UeVKNFdTtgF3U9pCIHj0iF9c9SAygf+4RgLKqNR8D4elPpYlPZD5vCphIChUuTGl\n" \
"YmpMaGhPo+OW+zo+Nw1PEn56DImGFWVpqtmEja+LlYimyyTq3cDRE07IdmmFtKll\n" \
"LQz5+5SxVRbGxIfjiiW81qYbWvbT+1gxVwfvzUXF6ljza17UhIEMteV6hMvaGuyT\n" \
"b42qZHrLNJ/ID1qAG+gFZr1DubLDiGSzxnGAjvwWM80zaLPHBcbUSXtOFWn3lsrZ\n" \
"aD3RLg4ivixGEmkG6vaBXQ5tXX+/j7tnSYdRWt5rdzP1s72/v/+6R9rD3/iuPdq0\n" \
"AouNRwjNkb+djAAB9Q/+XT/XTM1GPcodim9hSWT56FZDq35TR3vpz4cfXRKxseMB\n" \
"I/ISuJzy7zjvWIeXAVEh0aDVX3rvP24L+UhwK1RX9z+jKsdcFsuSaLA0KrOIIomY\n" \
"7pNUfzjyL/fOLmJWqpXvjvfDC/t5WS08ZSlyEfwtcVOu606I0d3nuW8PTs2q1RaY\n" \
"a/Saq37fwobnVKXm+V94g+GiDRryKjCqTjdhAgMBAAGjUzBRMB0GA1UdDgQWBBRB\n" \
"VvAaow5nJswngmP2n03vWW9WjzAfBgNVHSMEGDAWgBRBVvAaow5nJswngmP2n03v\n" \
"WW9WjzAPBgNVHRMBAf8EBTADAQH/MA0GCSqGSIb3DQEBDQUAA4ICAQB2lMCgxm9v\n" \
"BIADWdindhQXUlMUkxmmteFnZHvSgYinNiypV1RXFFCINVKDIHZVhtq/rDiKP7I8\n" \
"A7jzSnOJ+GJt+W+o4Di+MSVbZNmIEaCOrilza3lOmLmnfAdpnq2EnAQqx4B5g5Fz\n" \
```

Figure 4.13: Certificate declaration

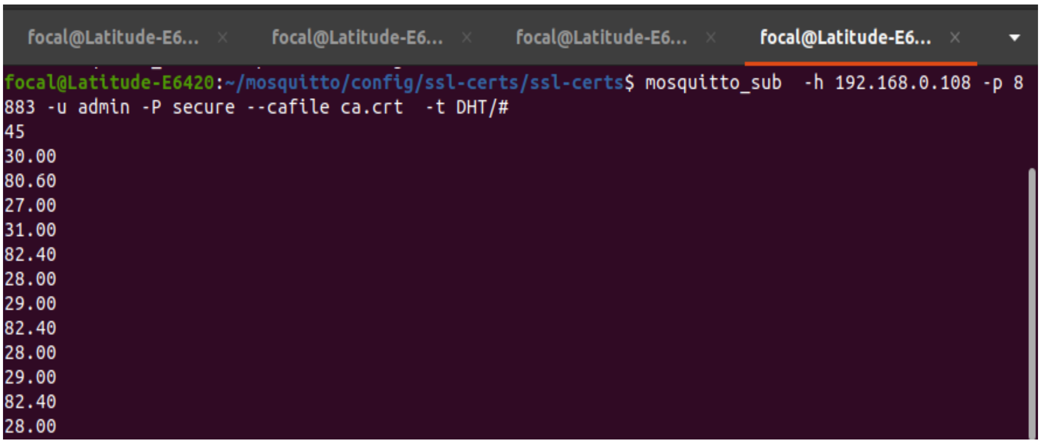
After successful implementation and configuration of certificate a clear output can be seen that TLS certificate verified over port 8883 which is secure port used for encrypted sensor traffic.



```
Attempting to connect to SSID: F-011
....Initializing MQTT
TLS certificate verified over port 8883
Successfully connected MQTT
Humidity = 29.00
temperature in celcius = 28.00
temperature in Farenhiet = 82.40
```

Figure 4.14: Security successful implemented for sensor communication

By subscribing on topic DHT/# for, temperature in Celsius, Fahrenheit and humidity with ca certificate a clear out can be seen after subscribing the topics.



```
focal@Latitude-E6420:~/mosquitto/config/ssl-certs/ssl-certs$ mosquitto_sub  -h 192.168.0.108 -p 8
883 -u admin -P secure --cafile ca.crt  -t DHT/#
45
30.00
80.60
27.00
31.00
82.40
28.00
29.00
82.40
28.00
29.00
82.40
28.00
```

Figure 4.15: Successful output after subscribing topics

4.3. Data visualization steps

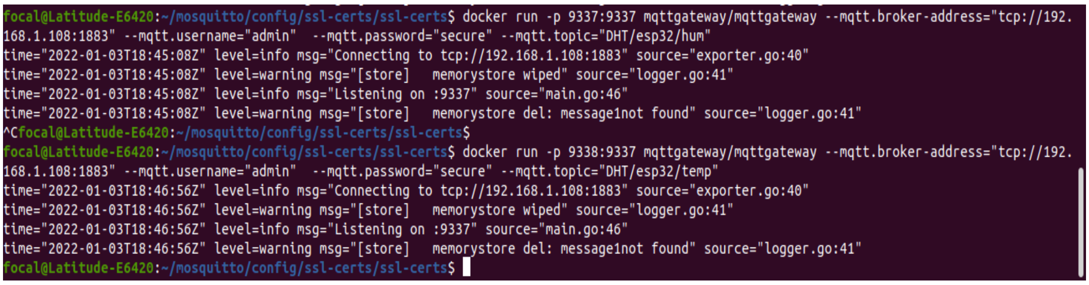
4.3.1. Integration of Prometheus with Mosquitto MQTT broker

Running mqtt-gateway dockers to integrate with Prometheus.

- Run 2 mqtt-gateway dockers to subscribe to 2 topics published by node MCU, “DHT/esp32/temp” and “DHT/esp32/hum” respectively, by using the following command.

a. docker run -p 9337:9337 mqttgateway/mqttgateway --mqtt.broker-address="tcp://192.168.1.108:1883" --mqtt.username="admin" --mqtt.password="secure" --mqtt.topic="DHT/esp32/hum"

b. docker run -p 9338:9337 mqttgateway/mqttgateway --mqtt.broker-address="tcp://192.168.1.108:1883" --mqtt.username="admin" --mqtt.password="secure" --mqtt.topic="DHT/esp32/temp"

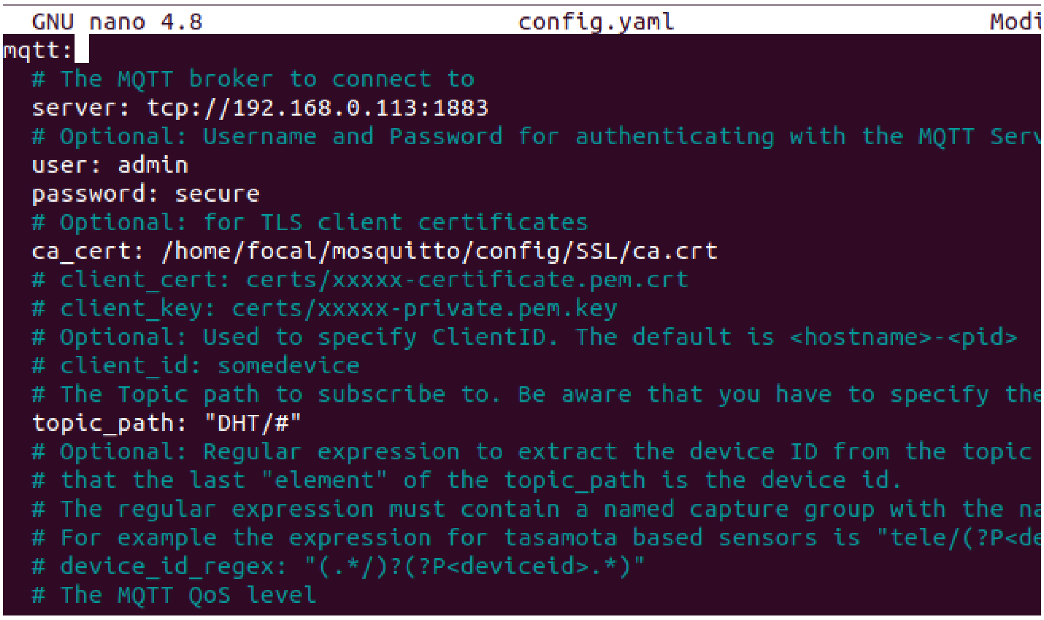


```
focal@Latitude-E6420:~/mosquitto/config/ssl-certs/ssl-certs$ docker run -p 9337:9337 mqttgateway/mqttgateway --mqtt.broker-address="tcp://192.
168.1.108:1883" --mqtt.username="admin"  --mqtt.password="secure" --mqtt.topic="DHT/esp32/hum"
time="2022-01-03T18:45:08Z" level=info msg="Connecting to tcp://192.168.1.108:1883" source="exporter.go:40"
time="2022-01-03T18:45:08Z" level=warning msg="[store]   memorystore wiped" source="logger.go:41"
time="2022-01-03T18:45:08Z" level=info msg="Listening on :9337" source="main.go:46"
time="2022-01-03T18:45:08Z" level=warning msg="[store]   memorystore del: message1not found" source="logger.go:41"
^Cfocal@Latitude-E6420:~/mosquitto/config/ssl-certs/ssl-certs$
focal@Latitude-E6420:~/mosquitto/config/ssl-certs/ssl-certs$ docker run -p 9338:9337 mqttgateway/mqttgateway --mqtt.broker-address="tcp://192.
168.1.108:1883" --mqtt.username="admin"  --mqtt.password="secure" --mqtt.topic="DHT/esp32/temp"
time="2022-01-03T18:46:56Z" level=info msg="Connecting to tcp://192.168.1.108:1883" source="exporter.go:40"
time="2022-01-03T18:46:56Z" level=warning msg="[store]   memorystore wiped" source="logger.go:41"
time="2022-01-03T18:46:56Z" level=info msg="Listening on :9337" source="main.go:46"
time="2022-01-03T18:46:56Z" level=warning msg="[store]   memorystore del: message1not found" source="logger.go:41"
focal@Latitude-E6420:~/mosquitto/config/ssl-certs/ssl-certs$
```

Figure 4.16: Integration of Prometheus with Mosquitto MQTT broker

B. go in the folder mqtt_exporter, by typing /home/focal/mqtt_exporter and configure config.yaml file.

Sudo nano config.yaml



```
  GNU nano 4.8                          config.yaml                          Modi
mqtt:
  # The MQTT broker to connect to
  server: tcp://192.168.0.113:1883
  # Optional: Username and Password for authenticating with the MQTT Serv
  user: admin
  password: secure
  # Optional: for TLS client certificates
  ca_cert: /home/focal/mosquitto/config/SSL/ca.crt
  # client_cert: certs/xxxxx-certificate.pem.crt
  # client_key: certs/xxxxx-private.pem.key
  # Optional: Used to specify ClientID. The default is <hostname>-<pid>
  # client_id: somedevice
  # The Topic path to subscribe to. Be aware that you have to specify the
  topic_path: "DHT/#"
  # Optional: Regular expression to extract the device ID from the topic
  # that the last "element" of the topic_path is the device id.
  # The regular expression must contain a named capture group with the na
  # For example the expression for tasamota based sensors is "tele/(?P<de
  # device_id_regex: "(.*/)?(?P<deviceid>.*)"
  # The MQTT QoS level
```

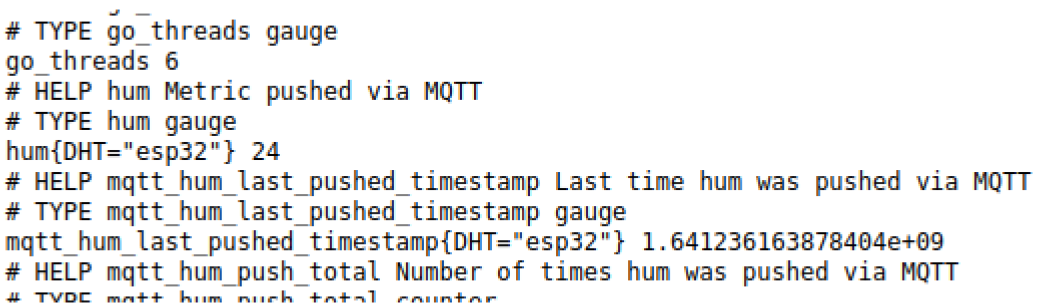
Figure 4.17: Configuration file for Prometheus with MQTT

C. open browser and check metrics for temperature and humidity.

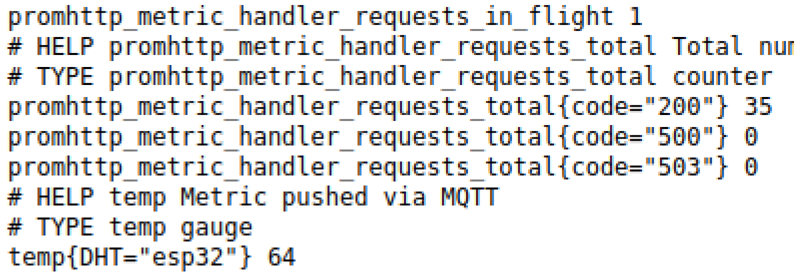
a. http://localhost:9337/metrics

b. http://localhost:9338/metrics

It will display published messages of Node MCU on broker.



```
# TYPE go_threads gauge
go_threads 6
# HELP hum Metric pushed via MQTT
# TYPE hum gauge
hum{DHT="esp32"} 24
# HELP mqtt_hum_last_pushed_timestamp Last time hum was pushed via MQTT
# TYPE mqtt_hum_last_pushed_timestamp gauge
mqtt_hum_last_pushed_timestamp{DHT="esp32"} 1.641236163878404e+09
# HELP mqtt_hum_push_total Number of times hum was pushed via MQTT
# TYPE mqtt_hum_push_total counter
```



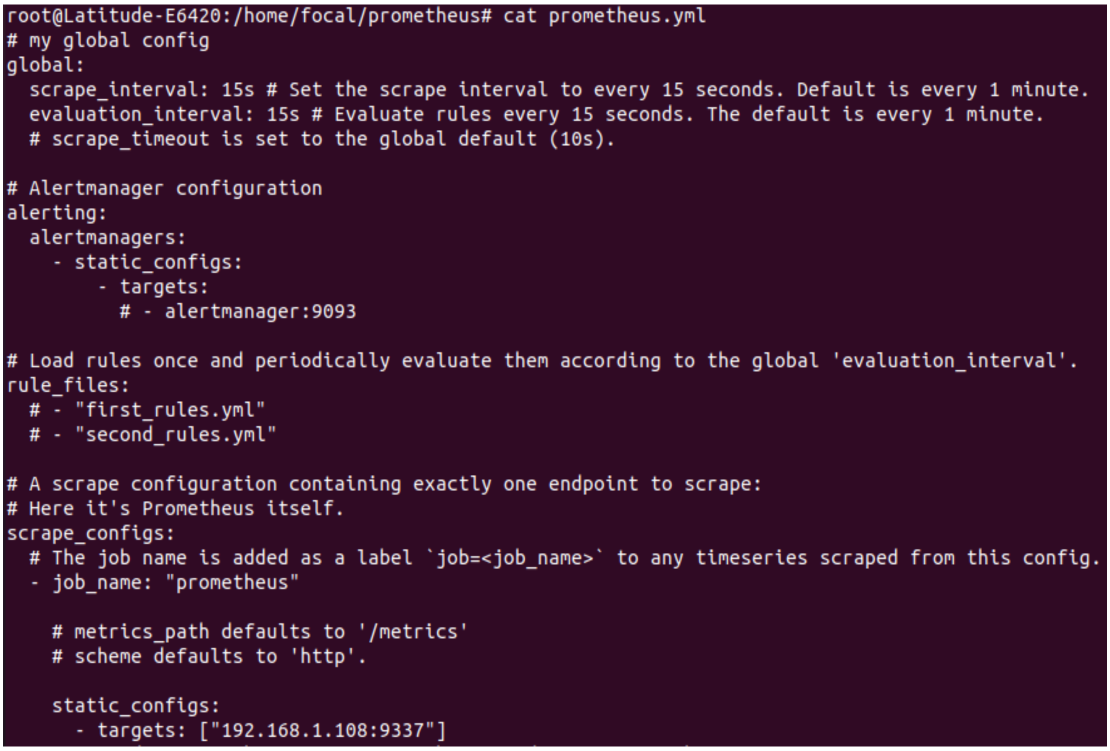
```
promhttp_metric_handler_requests_in_flight 1
# HELP promhttp_metric_handler_requests_total Total nur
# TYPE promhttp_metric_handler_requests_total counter
promhttp_metric_handler_requests_total{code="200"} 35
promhttp_metric_handler_requests_total{code="500"} 0
promhttp_metric_handler_requests_total{code="503"} 0
# HELP temp Metric pushed via MQTT
# TYPE temp gauge
temp{DHT="esp32"} 64
```

4.3.2. Configuring Prometheus to read and store data from mqtt-gateway

A. Open prometheus.yml file for setting mqtt-gateway configuration.

B. Create and open file prometheus.yml by typing sudo nano prometheus.yml, and paste content from github

C. Check file prometheus.yml by typing : cat prometheus.yml



```
root@Latitude-E6420:/home/focal/prometheus# cat prometheus.yml
# my global config
global:
  scrape_interval: 15s # Set the scrape interval to every 15 seconds. Default is every 1 minute.
  evaluation_interval: 15s # Evaluate rules every 15 seconds. The default is every 1 minute.
  # scrape_timeout is set to the global default (10s).

# Alertmanager configuration
alerting:
  alertmanagers:
    - static_configs:
        - targets:
          # - alertmanager:9093

# Load rules once and periodically evaluate them according to the global 'evaluation_interval'.
rule_files:
  # - "first_rules.yml"
  # - "second_rules.yml"

# A scrape configuration containing exactly one endpoint to scrape:
# Here it's Prometheus itself.
scrape_configs:
  # The job name is added as a label `job=<job_name>` to any timeseries scraped from this config.
  - job_name: "prometheus"

    # metrics_path defaults to '/metrics'
    # scheme defaults to 'http'.

    static_configs:
      - targets: ["192.168.1.108:9337"]
```

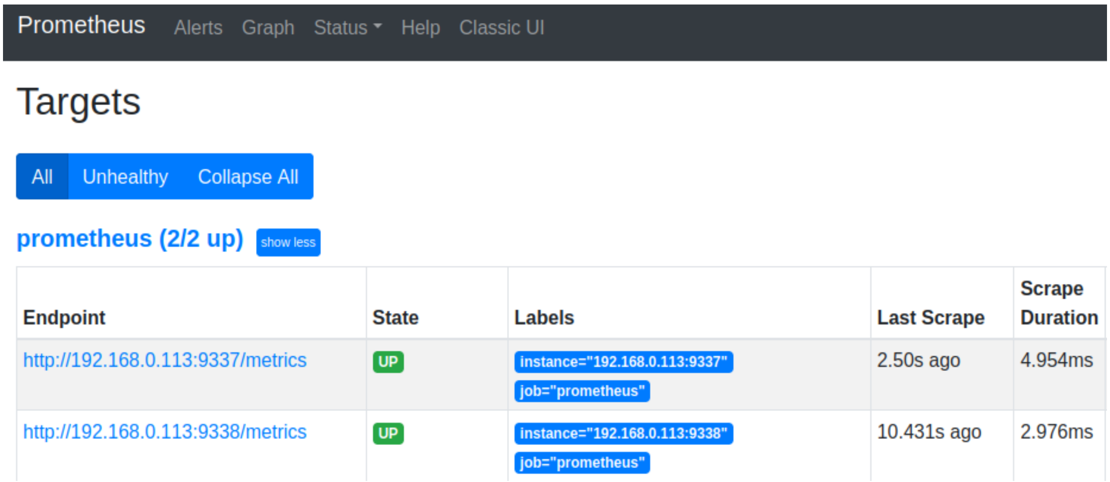
Figure 4.18: Configuration of Prometheus

A. Set targets for both published topics.

B. Now open prometheus by clicking the following link:
   http://192.168.1.108:9090

C. Click on status and then targets.

D. It will display both metrics.



Targets

prometheus (2/2 up)

Endpoint	State	Labels	Last Scrape	Scrape Duration

http://192.168.0.113:9337/metrics	UP	instance="192.168.0.113:9337"
		job="prometheus"
			2.50s ago	4.954ms

http://192.168.0.113:9338/metrics	UP	instance="192.168.0.113:9338"
		job="prometheus"
			10.431s ago	2.976ms

Figure 4.19: The end point on Prometheus

E. Click on the graph tab and enter predefined topics “hum” and “temp”.

F. Open Grafana and click the explore icon and select Prometheus.

G. Enter topic name, hum and temp and click run query.

H. Temperature and humidity Graph will be displayed.

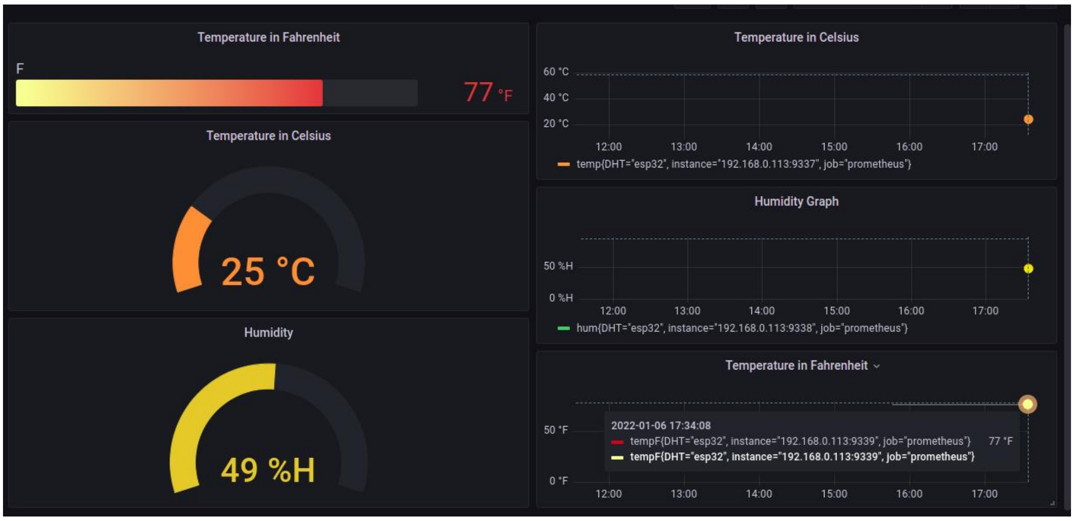
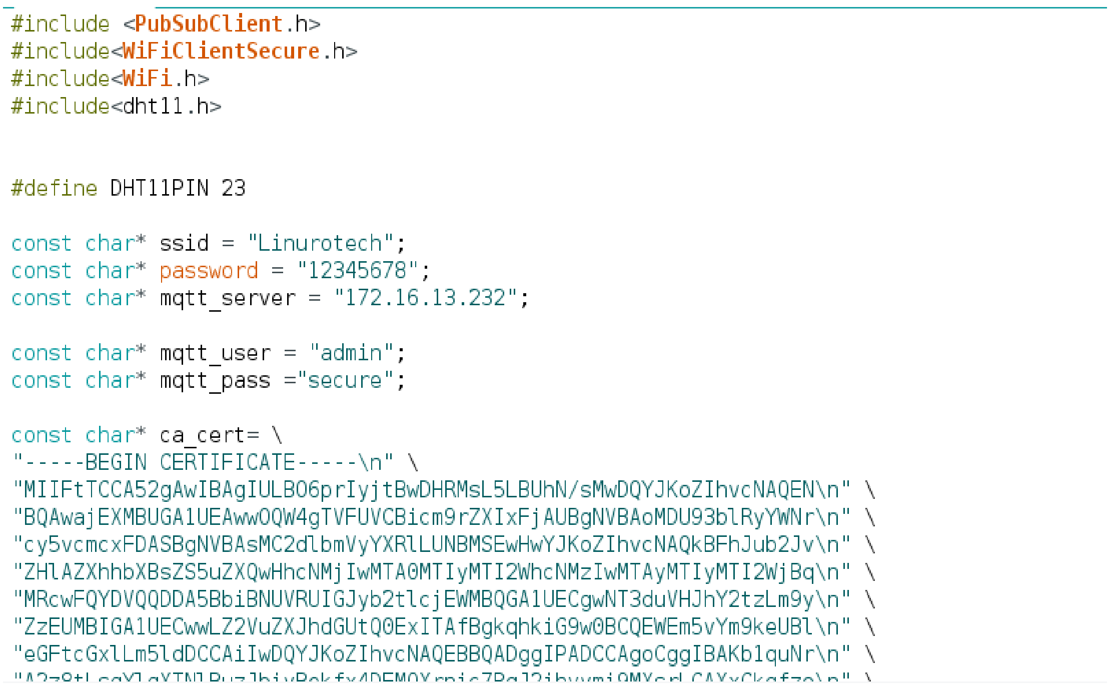

Temperature in Fahrenheit
F
77 °F
Temperature in Celsius
25 °C
Humidity
49 %H
Temperature in Celsius
60 °C
40 °C
20 °C
12:00
13:00
14:00
15:00
16:00
17:00
temp{DHT="esp32", instance="192.168.0.113:9337", job="prometheus"}
Humidity Graph
50 %H
0 %H
hum{DHT="esp32", instance="192.168.0.113:9338", job="prometheus"}
Temperature in Fahrenheit
50 °F
0 °F
2022-01-06 17:34:08
tempF{DHT="esp32", instance="192.168.0.113:9339", job="prometheus"} 77 °F
tempF{DHT="esp32", instance="192.168.0.113:9339", job="prometheus"}

Figure 4.20: Grafana dashboard visuals

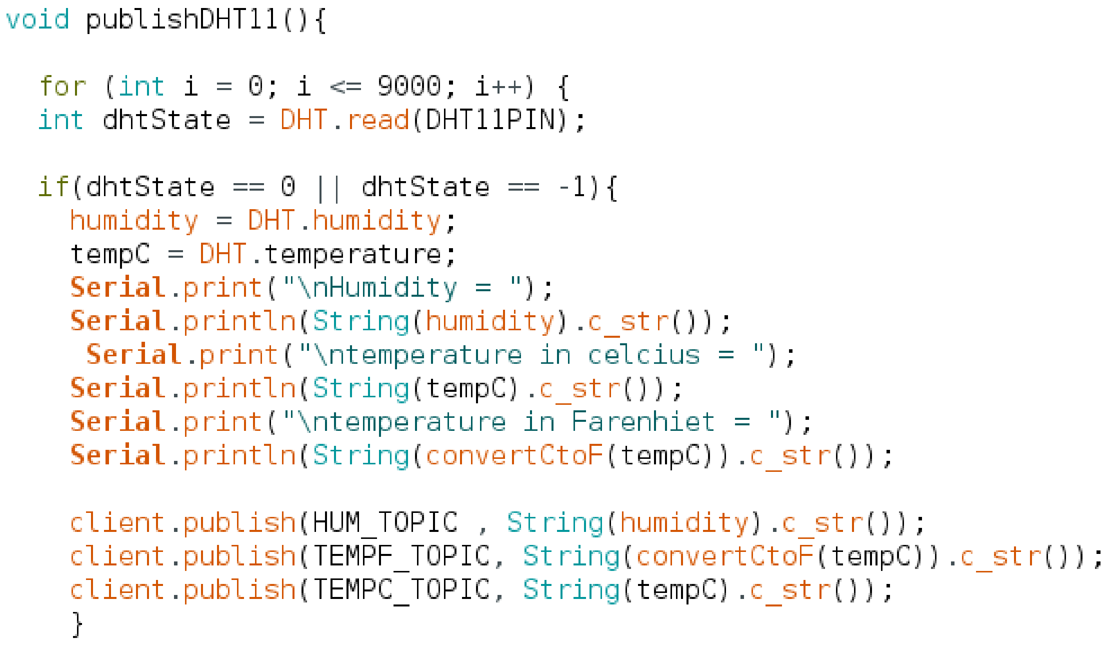
4.3.3. ESp32 and Arduino with DHT11 sensor



```
#include <PubSubClient.h>
#include<WiFiClientSecure.h>
#include<WiFi.h>
#include<dht11.h>


#define DHT11PIN 23

const char* ssid = "Linurotech";
const char* password = "12345678";
const char* mqtt_server = "172.16.13.232";

const char* mqtt_user = "admin";
const char* mqtt_pass ="secure";

const char* ca_cert= \
"-----BEGIN CERTIFICATE-----\n" \
"MIIFtTCCA52gAwIBAgIULBO6prIyjtBwDHRMsL5LBUhN/sMwDQYJKoZIhvcNAQEN\n" \
"BQAwajEXMBUGA1UEAww0QW4gTVFUVCBicm9rZXIxFjAUBgNVBAoMDU93blRyYWNr\n" \
"cy5vcmcxFDASBgNVBAsMC2dlbmVyYXRlLUNBMSEwHwYJKoZIhvcNAQkBFhJub2Jv\n" \
"ZHlAZXhhbXBsZS5uZXQwHhcNMjIwMTA0MTIyMTI2WhcNMzIwMTAyMTIyMTI2WjBq\n" \
"MRcwFQYDVQQDDA5BbiBNUVRUIGJyb2tlcjEWMBQGA1UECgwNT3duVHJhY2tzLm9y\n" \
"ZzEUMBIGA1UECwwLZ2VuZXJhdGUtQ0ExITAfBgkqhkiG9w0BCQEWEm5vYm9keUBl\n" \
"eGFtcGxlLm5ldDCCAiIwDQYJKoZIhvcNAQEBBQADggIPADCCAgoCggIBAKb1quNr\n" \
```

Figure 4.21: Configuration of Arduino with DHT11 sensor

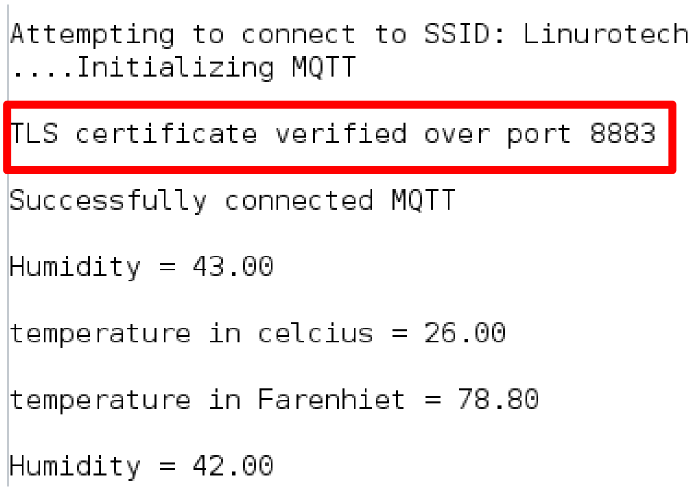
4.3.4. ESP32 actively publishing data on secure wifi network



```
void publishDHT11(){

  for (int i = 0; i <= 9000; i++) {
  int dhtState = DHT.read(DHT11PIN);

  if(dhtState == 0 || dhtState == -1){
    humidity = DHT.humidity;
    tempC = DHT.temperature;
    Serial.print("\nHumidity = ");
    Serial.println(String(humidity).c_str());
     Serial.print("\ntemperature in celcius = ");
    Serial.println(String(tempC).c_str());
    Serial.print("\ntemperature in Farenhiet = ");
    Serial.println(String(convertCtoF(tempC)).c_str());

    client.publish(HUM_TOPIC , String(humidity).c_str());
    client.publish(TEMPF_TOPIC, String(convertCtoF(tempC)).c_str());
    client.publish(TEMPC_TOPIC, String(tempC).c_str());
    }
```

Figure 4.22: Configuration of Esp32 for temperature sensor

4.3.5. ESp32 actively publishing data on secure wifi network.

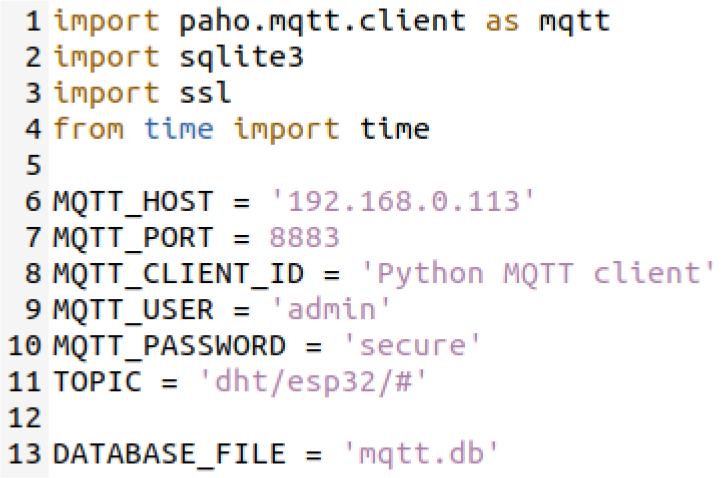


```
Attempting to connect to SSID: Linurotech
....Initializing MQTT

TLS certificate verified over port 8883

Successfully connected MQTT

Humidity = 43.00

temperature in celcius = 26.00

temperature in Farenhiet = 78.80

Humidity = 42.00
```

Figure 4.23: Actively publishing the data on secure wifi

.

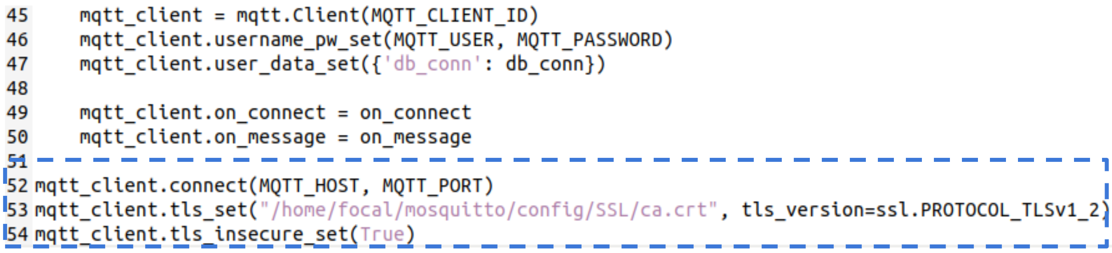
4.3.6. Storing MQTT data into SQLite database using TLS certificate



```python
import paho.mqtt.client as mqtt
import sqlite3
import ssl
from time import time

MQTT_HOST = '192.168.0.113'
MQTT_PORT = 8883
MQTT_CLIENT_ID = 'Python MQTT client'
MQTT_USER = 'admin'
MQTT_PASSWORD = 'secure'
TOPIC = 'dht/esp32/#'

DATABASE_FILE = 'mqtt.db'
```

Figure 4.24: Data storage on Sqlite database

4.3.7. Setting TLS certificate in python script:



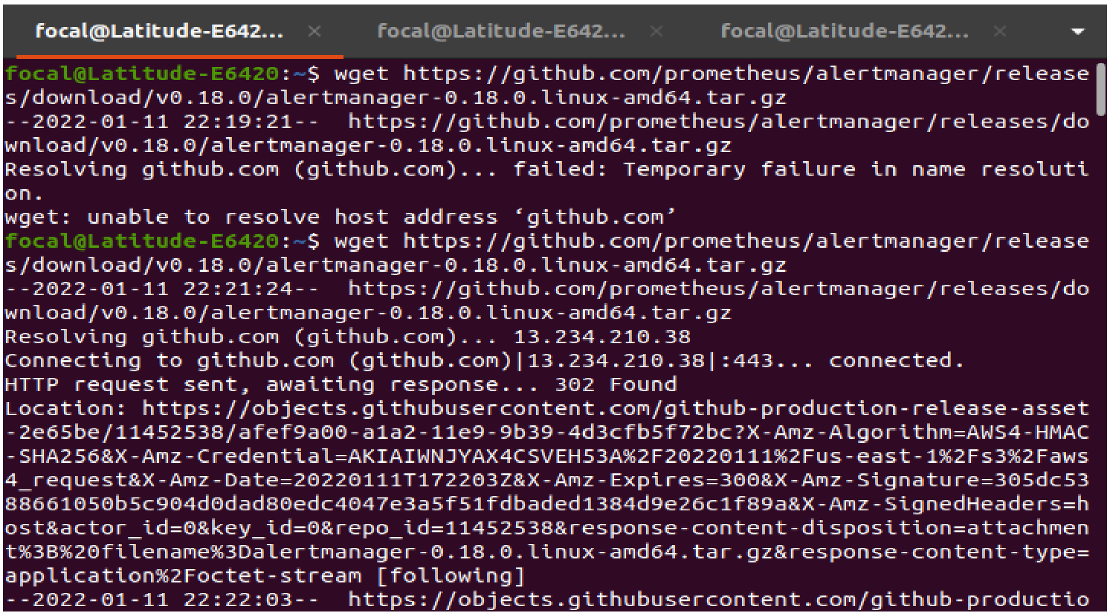
```python
    mqtt_client = mqtt.Client(MQTT_CLIENT_ID)
    mqtt_client.username_pw_set(MQTT_USER, MQTT_PASSWORD)
    mqtt_client.user_data_set({'db_conn': db_conn})

    mqtt_client.on_connect = on_connect
    mqtt_client.on_message = on_message

mqtt_client.connect(MQTT_HOST, MQTT_PORT)
mqtt_client.tls_set("/home/focal/mosquitto/config/SSL/ca.crt", tls_version=ssl.PROTOCOL_TLSv1_2)
mqtt_client.tls_insecure_set(True)
```

Figure 4.25: Setting TLS for MQTT client

4.3.8. Alert manager configuration on prometheus

1. Download from the following link.

wgethttps://github.com/prometheus/alertmanager/releases/download/v0.18.0/alertmanager-0.18.0.linux-amd64.tar.gz

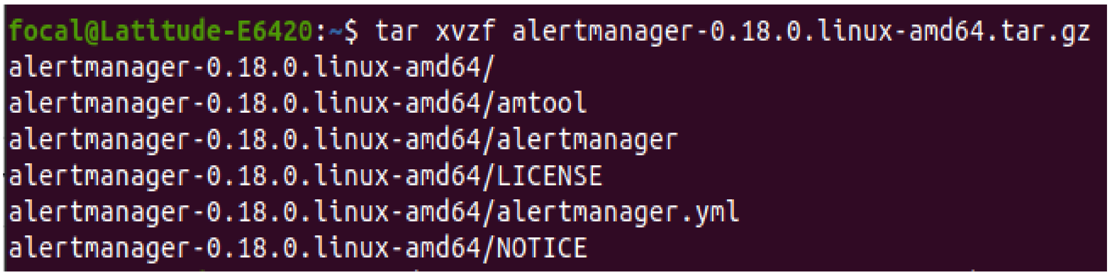


```
focal@Latitude-E6420:~$ wget https://github.com/prometheus/alertmanager/release
s/download/v0.18.0/alertmanager-0.18.0.linux-amd64.tar.gz
--2022-01-11 22:19:21--  https://github.com/prometheus/alertmanager/releases/do
wnload/v0.18.0/alertmanager-0.18.0.linux-amd64.tar.gz
Resolving github.com (github.com)... failed: Temporary failure in name resoluti
on.
wget: unable to resolve host address ‘github.com’
focal@Latitude-E6420:~$ wget https://github.com/prometheus/alertmanager/release
s/download/v0.18.0/alertmanager-0.18.0.linux-amd64.tar.gz
--2022-01-11 22:21:24--  https://github.com/prometheus/alertmanager/releases/do
wnload/v0.18.0/alertmanager-0.18.0.linux-amd64.tar.gz
Resolving github.com (github.com)... 13.234.210.38
Connecting to github.com (github.com)|13.234.210.38|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset
-2e65be/11452538/afef9a00-a1a2-11e9-9b39-4d3cfb5f72bc?X-Amz-Algorithm=AWS4-HMAC
-SHA256&X-Amz-Credential=AKIAIWNJYAX4CSVEH53A%2F20220111%2Fus-east-1%2Fs3%2Faws
4_request&X-Amz-Date=20220111T172203Z&X-Amz-Expires=300&X-Amz-Signature=305dc53
88661050b5c904d0dad80edc4047e3a5f51fdbaded1384d9e26c1f89a&X-Amz-SignedHeaders=h
ost&actor_id=0&key_id=0&repo_id=11452538&response-content-disposition=attachmen
t%3B%20filename%3Dalertmanager-0.18.0.linux-amd64.tar.gz&response-content-type=
application%2Foctet-stream [following]
--2022-01-11 22:22:03--  https://objects.githubusercontent.com/github-productio
```

Figure 4.26: Alert manager configuration for Prometheus

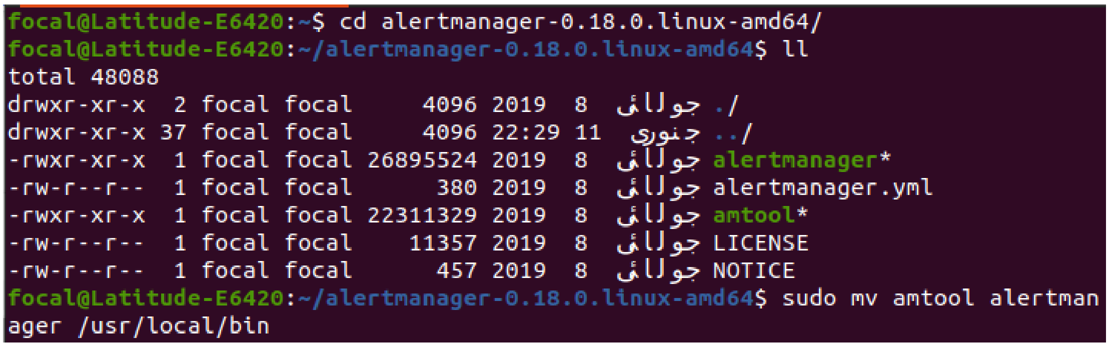
2. Extract the files from the archive.

tar xvzf alertmanager-0.18.0.linux-amd64.tar.gz



```
focal@Latitude-E6420:~$ tar xvzf alertmanager-0.18.0.linux-amd64.tar.gz
alertmanager-0.18.0.linux-amd64/
alertmanager-0.18.0.linux-amd64/amtool
alertmanager-0.18.0.linux-amd64/alertmanager
alertmanager-0.18.0.linux-amd64/LICENSE
alertmanager-0.18.0.linux-amd64/alertmanager.yml
alertmanager-0.18.0.linux-amd64/NOTICE
```

3. Move the executables to the /usr/local/bin folder.

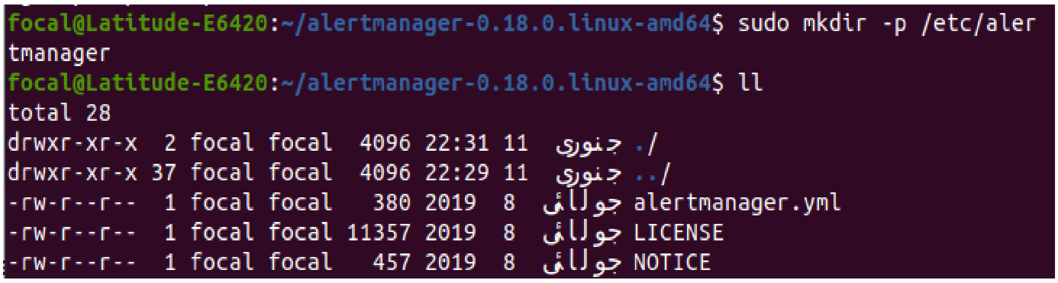


```
focal@Latitude-E6420:~$ cd alertmanager-0.18.0.linux-amd64/
focal@Latitude-E6420:~/alertmanager-0.18.0.linux-amd64$ ll
total 48088
drwxr-xr-x  2 focal focal     4096 2019  8  جوالى ./
drwxr-xr-x 37 focal focal     4096 22:29 11  جنوري ../
-rwxr-xr-x  1 focal focal 26895524 2019  8  جوالى alertmanager*
-rw-r--r--  1 focal focal      380 2019  8  جوالى alertmanager.yml
-rwxr-xr-x  1 focal focal 22311329 2019  8  جوالى amtool*
-rw-r--r--  1 focal focal    11357 2019  8  جوالى LICENSE
-rw-r--r--  1 focal focal      457 2019  8  جوالى NOTICE
focal@Latitude-E6420:~/alertmanager-0.18.0.linux-amd64$ sudo mv amtool alertman
ager /usr/local/bin
```

4. For the configuration files, create a new folder in /etc called alertmanager.

sudo mkdir -p /etc/alertmanager

sudo mv alertmanager.yml /etc/alertmanager



```
focal@Latitude-E6420:~/alertmanager-0.18.0.linux-amd64$ sudo mkdir -p /etc/aler
tmanager
focal@Latitude-E6420:~/alertmanager-0.18.0.linux-amd64$ ll
total 28
drwxr-xr-x  2 focal focal  4096 22:31 11  جنوري ./
drwxr-xr-x 37 focal focal  4096 22:29 11  جنوري ../
-rw-r--r--  1 focal focal   380 2019  8  جوالئ alertmanager.yml
-rw-r--r--  1 focal focal 11357 2019  8  جوالئ LICENSE
-rw-r--r--  1 focal focal   457 2019  8  جوالئ NOTICE
```

5. Create a data folder at the root directory, with a prometheus folder inside.

sudo mkdir -p /data/alertmanager

6. Next, create a user for your upcoming service.

sudo useradd -rs /bin/false alertmanager

7. Give permissions to your newly created user for the AlertManager binaries.

sudo chown alertmanager:alertmanager /usr/local/bin/amtool /usr/local/bin/alertmanager

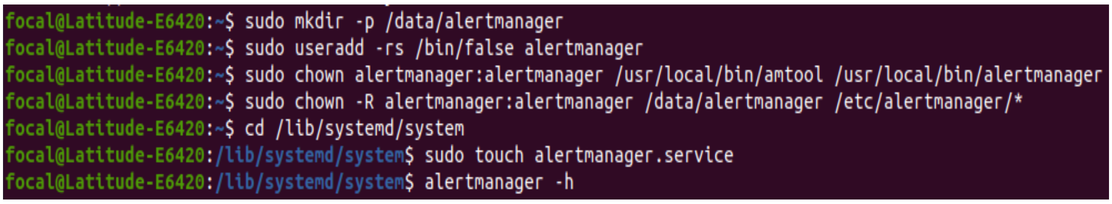
8. Give the correct permissions to those folders recursively.

sudo chown -R alertmanager:alertmanager /data/alertmanager /etc/alertmanager/

9. To create a Linux service (using systemd), head over to the */lib/systemd/system* folder and create a service named alertmanager.service

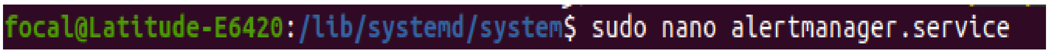
cd /lib/systemd/system

sudo touch alertmanager.service

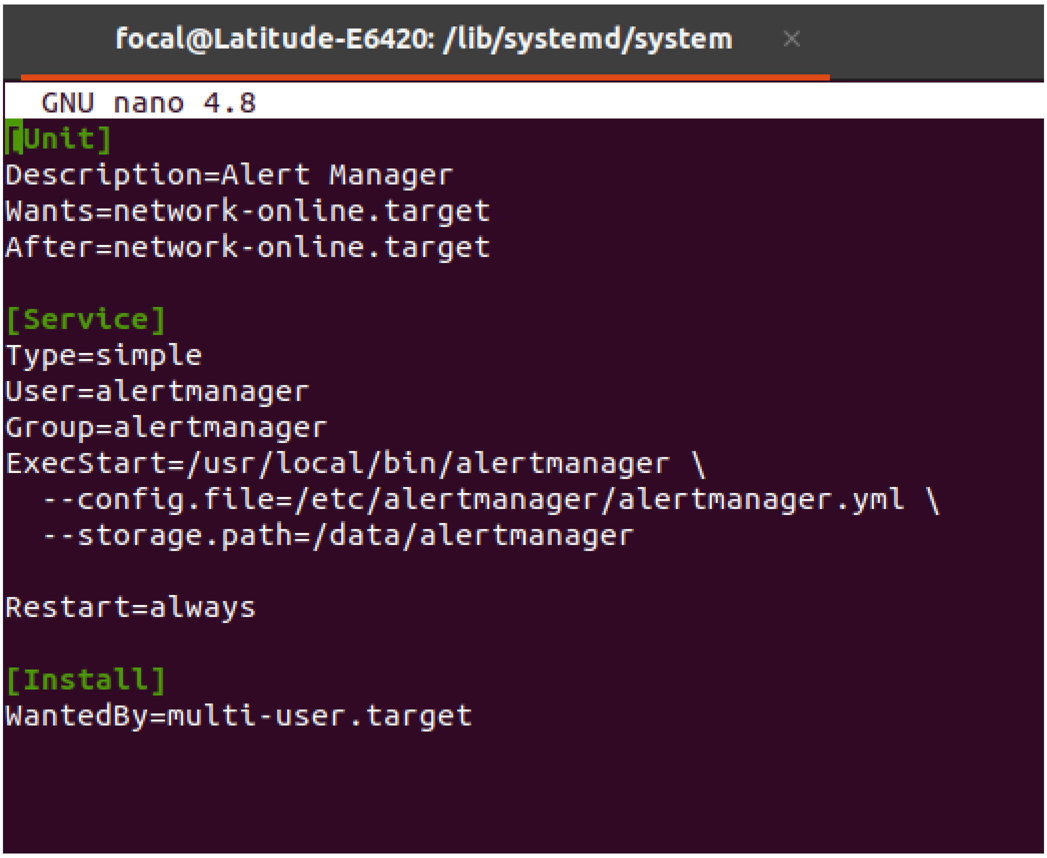


```
focal@Latitude-E6420:~$ sudo mkdir -p /data/alertmanager
focal@Latitude-E6420:~$ sudo useradd -rs /bin/false alertmanager
focal@Latitude-E6420:~$ sudo chown alertmanager:alertmanager /usr/local/bin/amtool /usr/local/bin/alertmanager
focal@Latitude-E6420:~$ sudo chown -R alertmanager:alertmanager /data/alertmanager /etc/alertmanager/*
focal@Latitude-E6420:~$ cd /lib/systemd/system
focal@Latitude-E6420:/lib/systemd/system$ sudo touch alertmanager.service
focal@Latitude-E6420:/lib/systemd/system$ alertmanager -h
```

10. Edit Service file of alertmanager.

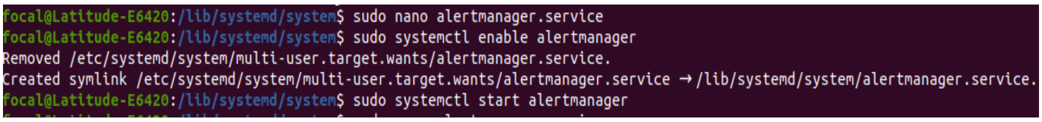


```
focal@Latitude-E6420:/lib/systemd/system$ sudo nano alertmanager.service
```



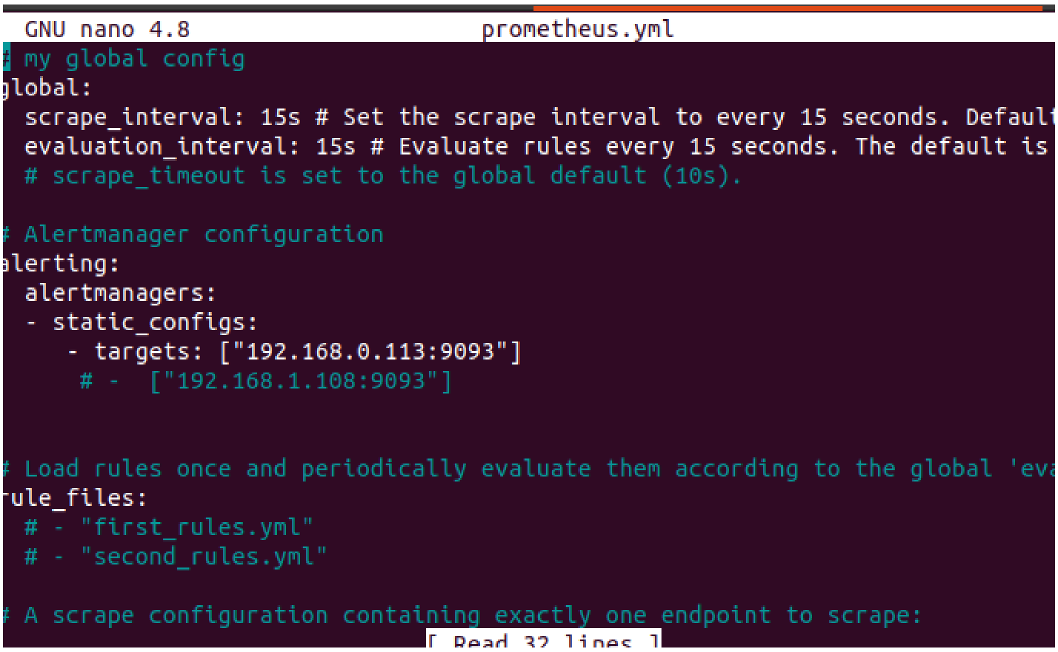
```
  GNU nano 4.8
[Unit]
Description=Alert Manager
Wants=network-online.target
After=network-online.target

[Service]
Type=simple
User=alertmanager
Group=alertmanager
ExecStart=/usr/local/bin/alertmanager \
  --config.file=/etc/alertmanager/alertmanager.yml \
  --storage.path=/data/alertmanager

Restart=always

[Install]
WantedBy=multi-user.target
```

11. Save the file, enable the service and start it.



```
focal@Latitude-E6420:/lib/systemd/system$ sudo nano alertmanager.service
focal@Latitude-E6420:/lib/systemd/system$ sudo systemctl enable alertmanager
Removed /etc/systemd/system/multi-user.target.wants/alertmanager.service.
Created symlink /etc/systemd/system/multi-user.target.wants/alertmanager.service → /lib/systemd/system/alertmanager.service.
focal@Latitude-E6420:/lib/systemd/system$ sudo systemctl start alertmanager
```

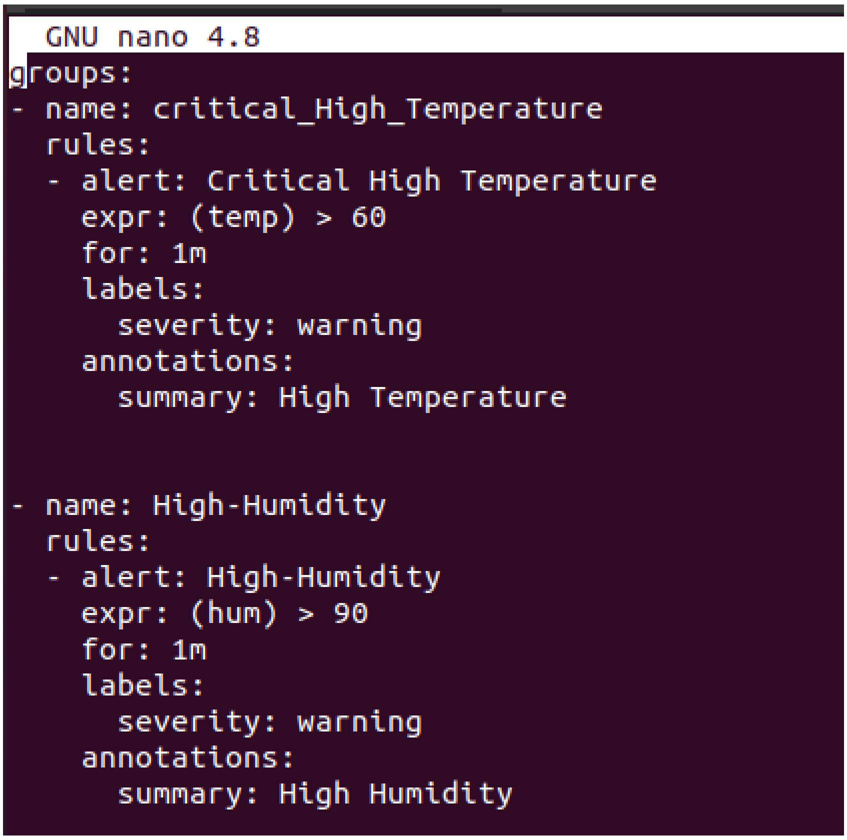
Figure 4.27: Integrating Prometheus with alert message



```
  GNU nano 4.8                              prometheus.yml
# my global config
global:
  scrape_interval: 15s # Set the scrape interval to every 15 seconds. Default
  evaluation_interval: 15s # Evaluate rules every 15 seconds. The default is
  # scrape_timeout is set to the global default (10s).

# Alertmanager configuration
alerting:
  alertmanagers:
  - static_configs:
      - targets: ["192.168.0.113:9093"]
       # -  ["192.168.1.108:9093"]


# Load rules once and periodically evaluate them according to the global 'eva
rule_files:
  # - "first_rules.yml"
  # - "second_rules.yml"

# A scrape configuration containing exactly one endpoint to scrape:
                                [ Read 32 lines ]
```

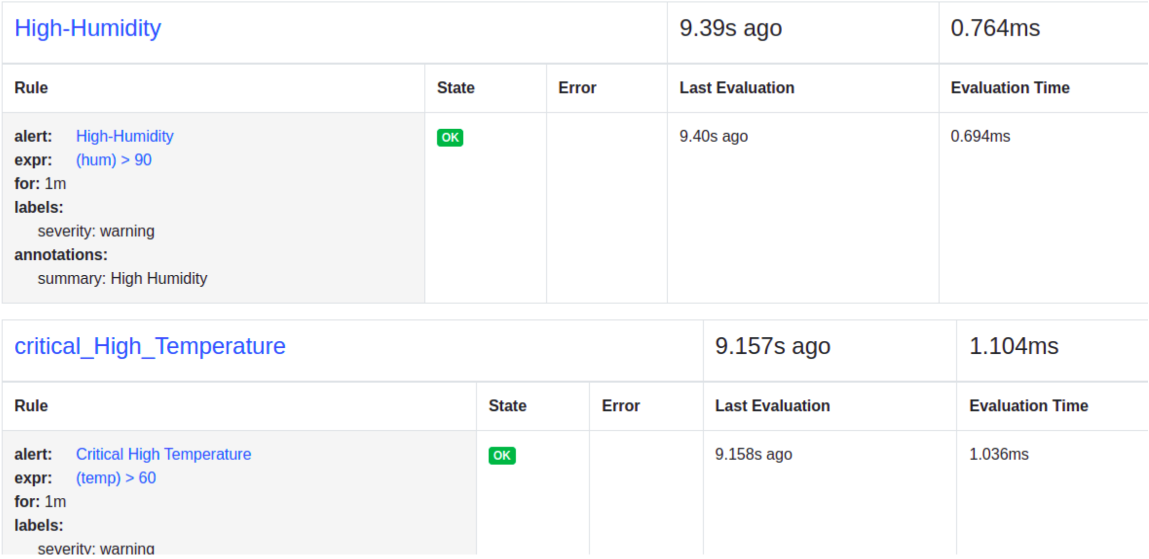
Figure 4.28: Configuration of prometheus.yml

12. Adding alerts file Rule.Yml



```
  GNU nano 4.8
groups:
- name: critical_High_Temperature
  rules:
  - alert: Critical High Temperature
    expr: (temp) > 60
    for: 1m
    labels:
      severity: warning
    annotations:
      summary: High Temperature


- name: High-Humidity
  rules:
  - alert: High-Humidity
    expr: (hum) > 90
    for: 1m
    labels:
      severity: warning
    annotations:
      summary: High Humidity
```

Figure 4.29: The setting up rules for alert in yml file



High-Humidity			9.39s ago	0.764ms
Rule	**State**	**Error**	**Last Evaluation**	**Evaluation Time**
alert: High-Humidity **expr:** (hum) > 90 **for:** 1m **labels:** severity: warning **annotations:** summary: High Humidity	OK		9.40s ago	0.694ms

critical_High_Temperature			9.157s ago	1.104ms
Rule	**State**	**Error**	**Last Evaluation**	**Evaluation Time**
alert: Critical High Temperature **expr:** (temp) > 60 **for:** 1m **labels:** severity: warning	OK		9.158s ago	1.036ms

Figure 4.30: The rules for alerts on Prometheus

4.3.9. Results of key size comparison between RSA and ECC

In terms of memory requirements and memory utilization, ECC outperforms RSA. The ECC algorithm provided a comparable level of security to RSA while utilizing a smaller amount of memory resources.

Table 4.1: Comparative analysis RSA and ECC in terms of key size comparison

Operation	**Security level (bits)**	**RSA**		**ECC**	
		Time (seconds)	**Data Memory (bytes)**	**Time (seconds)**	**Data Memory (bytes)**
Public Key operation	80	0.47s	539	0.84s	282
Private Key operation	80	10.78s	919	0.83s	287
Public Key operation	112	1.81s	1272	2.23s	426
Private Key operation	112	13.58s	1780	2.25s	422

4.3.10. Results of performance, security, and space requirements comparison of RSA and ECC

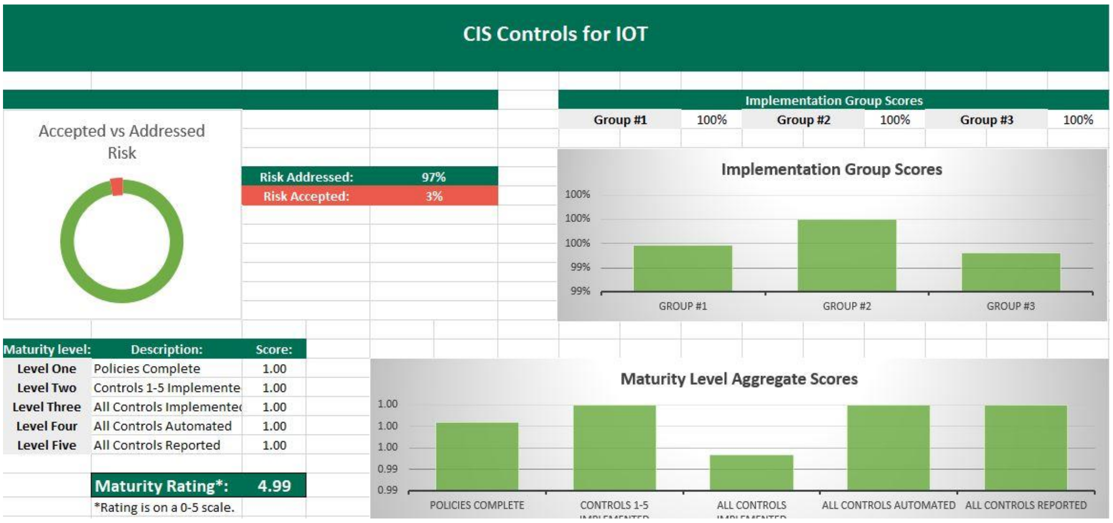
ECC tends to outperform RSA in terms of performance and space requirements while providing comparable or even enhanced security.

Table 4.2: Performance, security and space requirements comparison of RSA and ECC

	Key generation time (ms)	**Memory requirement (bytes)**	**Encrypt/Decrypt Time (ms)**
ECC (210 bits)	117	139	13
RSA (2048 bits)	18386	619	1866
ECC (106 bits)	56	106	10
RSA (512 bits)	381	151	16
ECC (160 bits)	110	127	17
RSA (1024)	2611	315	341
ECC (132 bits)	99	118	18
RSA (768 bits)	891	237	161

4.3.11. Risk Assessment Results

Risk assessment was performed on the basis of CIS's top 18 controls for IoT. The results were surprising as our framework addressed 97% of risks and the maturity rating was 4.99. Any organization can identify and mitigate the risk for their IOT/IoMT systems by conducting the risk assessment in compliance with these CIS controls. After conducting risk assessment the threats and vulnerabilities can be identified. This helps a lot to mitigate and make strategies to overcome the risks.



CIS Controls for IOT

Accepted vs Addressed Risk

Risk Addressed:	97%
Risk Accepted:	3%

Implementation Group Scores

Group #1	100%	Group #2	100%	Group #3	100%

Maturity level:	Description:	Score:
Level One	Policies Complete	1.00
Level Two	Controls 1-5 Implemente	1.00
Level Three	All Controls Implementec	1.00
Level Four	All Controls Automated	1.00
Level Five	All Controls Reported	1.00

Maturity Rating*: 4.99

*Rating is on a 0-5 scale.

Figure 4.31: Risk assessment based on CIS critical controls for IOT

CONCLUSION

The upholding of security is an essential element in the transmission of healthcare data inside the Internet of Medical Things (IoMT) network, owing to the sensitive and secret nature of the information involved. Nevertheless, sensors used on the Internet of Medical Things (IoMT) sometimes possess limited resources, and certain implanted sensors necessitate the use of external devices for their security. The validity of our method was confirmed through experimentation conducted in a laboratory environment. Specifically, we constructed a system and executed dockers, which included containers, within this setup. RSA and Lightweight cryptography were employed to ensure the security of the sensor communication. Comparison was made and the framework showed the better results by using lightweight ECC. The test results demonstrate encouraging outcomes for an end-to-end security solution aimed at safeguarding the transfer of healthcare data on the Internet of Medical Things (IoMT). Risk assessment was performed on the basis of CIS's top 18 controls for IoT. The results were surprising as our framework addressed 97% of risks and the maturity rating was 4.99 out of 5. This study also made a valuable contribution by expanding the scope for subsequent research on security measures driven by Internet of Medical Things (IoMT) sensors.

REFERENCES

1. Lee, J.Y., et al., *Managing High-Cost Healthcare Users: The International Search for Effective Evidence-Supported Strategies.* Journal of the American Geriatrics Society, 2018. **66**(5): p. 1002-1008.
2. Atkinson, M., et al., *Scientific workflows: Past, present and future*. 2017, Elsevier. p. 216-227.
3. Heesterbeek, H., et al., *Modeling infectious disease dynamics in the complex landscape of global health.* Science, 2015. **347**(6227): p. aaa4339.
4. Mathew, P.S., A.S. Pillai, and V. Palade, *Applications of IoT in healthcare.* Cognitive Computing for Big Data Systems Over IoT: Frameworks, Tools and Applications, 2018: p. 263-288.
5. Dao, N.-N., *Internet of wearable things: Advancements and benefits from 6G technologies.* Future Generation Computer Systems, 2023. **138**: p. 172-184.
6. Elul, Y., et al., *Meeting the unmet needs of clinicians from AI systems showcased for cardiology with deep-learning–based ECG analysis.* Proceedings of the National Academy of Sciences, 2021. **118**(24): p. e2020620118.
7. Farrokhi, A., et al., *Application of Internet of Things and artificial intelligence for smart fitness: A survey.* Computer Networks, 2021. **189**: p. 107859.
8. Jaarsma, T., et al., *Self-care of heart failure patients: practical management recommendations from the Heart Failure Association of the European Society of Cardiology.* European journal of heart failure, 2021. **23**(1): p. 157-174.
9. Heidbuchel, H., et al., *Recommendations for participation in leisure-time physical activity and competitive sports in patients with arrhythmias and potentially arrhythmogenic conditions: Part 1: Supraventricular arrhythmias. A position statement of the Section of Sports Cardiology and Exercise from the European Association of Preventive Cardiology (EAPC) and the European Heart Rhythm Association (EHRA), both associations of the European Society of Cardiology.* European journal of preventive cardiology, 2021. **28**(14): p. 1539-1551.
10. Kim, D.W., et al., *Efficient assessment of real-world dynamics of circadian rhythms in heart rate and body temperature from wearable data.* Journal of the Royal Society Interface, 2023. **20**(205): p. 20230030.

11. Pastorino, R., et al., *Benefits and challenges of Big Data in healthcare: an overview of the European initiatives.* European journal of public health, 2019. **29**(Supplement_3): p. 23-27.

12. SRAIDI, N., *STAKEHOLDERS'PERSPECTIVES ON WEARABLE INTERNET OF MEDICAL THINGS PRIVACY AND SECURITY.* International Journal of Computations, Information and Manufacturing (IJCIM), 2022. **2**(2).

13. Sindhuja, R. *A Survey of Internet of Medical Things (IoMT) Applications, Architectures and Challenges in Smart Healthcare Systems*. in *ITM Web of Conferences*. 2023. EDP Sciences.

14. Hill, S., *Scalable iot platforms*. 2019.

15. Rasool, R.U., et al., *Security and privacy of internet of medical things: A contemporary review in the age of surveillance, botnets, and adversarial ML.* Journal of Network and Computer Applications, 2022. **201**: p. 103332.

16. Farid, F., et al., *The Roles of AI Technologies in Reducing Hospital Readmission for Chronic Diseases: A Comprehensive Analysis.* 2023.

17. Dwivedi, R., D. Mehrotra, and S. Chandra, *Potential of Internet of Medical Things (IoMT) applications in building a smart healthcare system: A systematic review.* Journal of oral biology and craniofacial research, 2022. **12**(2): p. 302-318.

18. Rawat, R., *a Systematic Review of Blockchain Technology Use in E-Supply Chain in Internet of Medical Things (Iomt).* International Journal of Computations, Information and Manufacturing (IJCIM), 2022. **2**(2).

19. Tulchinsky, T.H., *Ethical issues in public health.* Case Studies in Public Health, 2018: p. 277.

20. Awotunde, J.B., et al., *AiIoMT: IoMT-based system-enabled artificial intelligence for enhanced smart healthcare systems.* Machine Learning for Critical Internet of Medical Things: Applications and Use Cases, 2022: p. 229-254.

21. Ashfaq, Z., et al., *A review of enabling technologies for Internet of Medical Things (IoMT) Ecosystem.* Ain Shams Engineering Journal, 2022. **13**(4): p. 101660.

22. Anders, V., *For a public-private partnership to achieve migrant health equality in Morocco: A Cross-Analysis of Integration Policies and Migrant Peer Educator Programs.* 2016.

23. Inam, A., et al., *Using causal loop diagrams for the initialization of stakeholder engagement in soil salinity management in agricultural watersheds in developing countries: A case study in the Rechna Doab watershed, Pakistan.* Journal of environmental management, 2015. **152**: p. 251-267.

24. Gill, S.S., et al., *Transformative effects of IoT, Blockchain and Artificial Intelligence on cloud computing: Evolution, vision, trends and open challenges.* Internet of Things, 2019. **8**: p. 100118.

25. Aminizadeh, S., et al., *The applications of machine learning techniques in medical data processing based on distributed computing and the Internet of Things.* Computer Methods and Programs in Biomedicine, 2023: p. 107745.

26. Monteiro, A.C.B., et al., *An overview of medical Internet of Things, artificial intelligence, and cloud computing employed in health care from a modern panorama.* The Fusion of Internet of Things, Artificial Intelligence, and Cloud Computing in Health Care, 2021: p. 3-23.

27. Alam, S., et al., *Blockchain-based solutions supporting reliable healthcare for fog computing and internet of medical things (IoMT) integration.* Sustainability, 2022. **14**(22): p. 15312.

28. Steger, M., et al., *An efficient and secure automotive wireless software update framework.* IEEE Transactions on Industrial Informatics, 2017. **14**(5): p. 2181-2193.

29. Abbas, A., et al., *Blockchain-assisted secured data management framework for health information analysis based on Internet of Medical Things.* Personal and ubiquitous computing, 2021: p. 1-14.

30. Tian, S., et al., *Smart healthcare: making medical care more intelligent.* Global Health Journal, 2019. **3**(3): p. 62-65.

31. Potter, P.A., et al., *Fundamentals of nursing-e-book*. 2021: Elsevier health sciences.

32. Kulkarni, R., *Use of telehealth in the delivery of comprehensive care for patients with haemophilia and other inherited bleeding disorders.* Haemophilia, 2018. **24**(1): p. 33-42.

33. Evans, C.R., M.G. Medina, and A.M. Dwyer, *Telemedicine and telerobotics: from science fiction to reality*. Updates in surgery, 2018. **70**: p. 357-362.

34. Hassan, H.O., S. Azizi, and M. Shojafar, *Priority, network and energy-aware placement of IoT-based application services in fog-cloud environments.* IET communications, 2020. **14**(13): p. 2117-2129.

35. Sadiq, M., I. Singh, and M. Ahmad, *Internet of Medical Things in curbing pandemics*, in *Deep Learning in Personalized Healthcare and Decision Support*. 2023, Elsevier. p. 357-371.

36. Devi, D.H., et al., *5g technology in healthcare and wearable devices: A review.* Sensors, 2023. **23**(5): p. 2519.

37. Dogheim, G.M. and A. Hussain, *Patient Care through AI-driven Remote Monitoring: Analyzing the Role of Predictive Models and Intelligent Alerts in Preventive Medicine.* Journal of Contemporary Healthcare Analytics, 2023. **7**(1): p. 94-110.

38. Hick, J.L., et al., *Duty to plan: health care, crisis standards of care, and novel coronavirus SARS-CoV-2.* Nam Perspectives, 2020. **2020**.

39. Dias, F.M., et al., *Risk management focusing on the best practices of data security systems for healthcare.* International Journal of Innovation, 2021. **9**(1): p. 45-78.

40. ESCAP, U., *Delivering on the sustainable development goals through solutions at the energy, food and finance nexus.* 2023.

41. Pateraki, M., et al., *Biosensors and Internet of Things in smart healthcare applications: Challenges and opportunities.* Wearable and Implantable Medical Devices, 2020: p. 25-53.

42. Ullah, M., et al., *Smart Technologies used as Smart Tools in the Management of Cardiovascular Disease and their Future Perspective.* Current Problems in Cardiology, 2023. **48**(11): p. 101922.

43. Hussien, H.M., et al., *Blockchain technology in the healthcare industry: Trends and opportunities.* Journal of Industrial Information Integration, 2021. **22**: p. 100217.

44. Rahman, M. and H. Jahankhani, *Security vulnerabilities in existing security mechanisms for iomt and potential solutions for mitigating cyber-attacks.* Information security technologies for controlling pandemics, 2021: p. 307-334.

45. Montasari, R., et al., *Application of artificial intelligence and machine learning in producing actionable cyber threat intelligence.* Digital Forensic Investigation of Internet of Things (IoT) Devices, 2021: p. 47-64.

46. Anguraj, D.K., *Advanced encryption standard based secure iot data transfer model for cloud analytics applications.* Journal of Information Technology and Digital World, 2022. **4**(2): p. 114-124.

47. Charjan, D.M., P.M. Bochare, and Y.R. Bhuyar, *An overview of secure sockets layer.* Int. J. Comput. Sci. Appl, 2013. **6**(2): p. 388-393.

48. Clinton, B., *A national security strategy for a new century*. 1998: White House.

49. Council, N.R., *Reaping the benefits of genomic and proteomic research: Intellectual property rights, innovation, and public health*. 2006: National Academies Press.

50. Meinert, E., et al., *The internet of things in health care in oxford: protocol for proof-of-concept projects.* JMIR research protocols, 2018. **7**(12): p. e12077.

51. Tikkha, R. and S. Sharma. *Cryptographic Measures in IoMT: Security Threats and Measurement*. in *2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT)*. 2022. IEEE.

52. Ortiz, E. and C.M. Clancy, *Use of information technology to improve the quality of health care in the United States.* Health Services Research, 2003. **38**(2): p. xi.

53. Krzesiński, P., *Digital health technologies for post-discharge care after heart failure hospitalisation to relieve symptoms and improve clinical outcomes.* Journal of Clinical Medicine, 2023. **12**(6): p. 2373.

54. Thomasian, N.M. and E.Y. Adashi, *Cybersecurity in the internet of medical things.* Health Policy and Technology, 2021. **10**(3): p. 100549.

55. Chaudhary, S., et al., *A taxonomy on smart healthcare technologies: Security framework, case study, and future directions.* Journal of Sensors, 2022. **2022**.

56. Kakhi, K., et al., *The internet of medical things and artificial intelligence: trends, challenges, and opportunities.* Biocybernetics and Biomedical Engineering, 2022. **42**(3): p. 749-771.

57. Anikwe, C.V., et al., *Mobile and wearable sensors for data-driven health monitoring system: State-of-the-art and future prospect.* Expert Systems with Applications, 2022. **202**: p. 117362.

58. Li, G., et al., *EdgeLaaS: Edge learning as a service for knowledge-centric connected healthcare.* IEEE Network, 2019. **33**(6): p. 37-43.

59. Bardhan, I., H. Chen, and E. Karahanna, *Connecting systems, data, and people: A multidisciplinary research roadmap for chronic disease management.* MIS Quarterly, 2020. **44**(1): p. 185-200.

60. Lonsdale, H., et al., *The perioperative human digital twin.* Anesthesia & Analgesia, 2022. **134**(4): p. 885-892.

61. Chen, S.-H., et al., *Advantage in privacy protection by using synchronous video observed treatment enhances treatment adherence among patients with latent tuberculosis infection.* Journal of Infection and Public Health, 2020. **13**(9): p. 1354-1359.

62. Nanayakkara, S., X. Zhou, and H. Spallek, *Impact of big data on oral health outcomes.* Oral Diseases, 2019. **25**(5): p. 1245-1252.

63. Chaaben, A.B., *A Survey of Different IoMT Protocols for Healthcare Applications.* ResearchBerg Review of Science and Technology, 2022. **2**(1): p. 41-57.

64. Irving, M., et al., *Using teledentistry in clinical practice as an enabler to improve access to clinical care: A qualitative systematic review.* Journal of telemedicine and telecare, 2018. **24**(3): p. 129-146.

65. Aslam, B., et al., *Blockchain and ANFIS empowered IoMT application for privacy preserved contact tracing in COVID-19 pandemic.* Personal and ubiquitous computing, 2021: p. 1-17.

66. Abugabah, A., N. Nizamuddin, and A.A. Alzubi, *Decentralized telemedicine framework for a smart healthcare ecosystem.* IEEE Access, 2020. **8**: p. 166575-166588.

67. S. Rubí, J.N. and P.R. L. Gondim, *IoMT platform for pervasive healthcare data aggregation, processing, and sharing based on OneM2M and OpenEHR.* Sensors, 2019. **19**(19): p. 4283.

68. Wilson, D. *An overview of the application of wearable technology to nursing practice*. in *Nursing forum*. 2017. Wiley Online Library.

69. Singh, S., A. Singh, and S. Limkar, *Prediction of Heart Disease Using Deep Learning and Internet of Medical Things.* International Journal of Intelligent Systems and Applications in Engineering, 2024. **12**(1s): p. 512-525.

70. Paul, M., et al., *Digitization of healthcare sector: A study on privacy and security concerns.* ICT Express, 2023.

71. Bhosale, K.S., M. Nenova, and G. Iliev. *A study of cyber attacks: In the healthcare sector*. in *2021 Sixth Junior Conference on Lighting (Lighting)*. 2021. IEEE.

72. Koutras, D., et al., *Security in IoMT communications: A survey.* Sensors, 2020. **20**(17): p. 4828.

73. Yaacoub, J.-P.A., et al., *Securing internet of medical things systems: Limitations, issues and recommendations.* Future Generation Computer Systems, 2020. **105**: p. 581-606.

74. Kumar, P., S. Chauhan, and L.K. Awasthi, *Artificial intelligence in healthcare: review, ethics, trust challenges & future research directions.* Engineering Applications of Artificial Intelligence, 2023. **120**: p. 105894.

75. Karam, A.A., *INVESTIGATING THE IMPORTANCE OF ETHICS AND SECURITY ON INTERNET OF MEDICAL THINGS (IoMT).* International Journal of Computations, Information and Manufacturing (IJCIM), 2022. **2**(2).

76. Sanchez-Martinez, S., et al., *Machine learning for clinical decision-making: challenges and opportunities in cardiovascular imaging.* Frontiers in Cardiovascular Medicine, 2022. **8**: p. 765693.

77. Fiske, A., A. Buyx, and B. Prainsack, *Health information counselors: a new profession for the age of big data.* Academic Medicine, 2019. **94**(1): p. 37.

78. Kim, S., et al., *Analysis of the factors influencing healthcare professionals' adoption of mobile electronic medical record (EMR) using the unified theory of acceptance and use of technology (UTAUT) in a tertiary hospital.* BMC medical informatics and decision making, 2015. **16**(1): p. 1-12.

79. Abdel-Basset, M., et al., *Internet of things, preliminaries and foundations.* Deep learning techniques for IoT security and privacy, 2022: p. 37-65.

80. Kumar, M., et al., *ANAF-IoMT: A novel architectural framework for IoMT-enabled smart healthcare system by enhancing security based on RECC-VC.* IEEE Transactions on Industrial Informatics, 2022. **18**(12): p. 8936-8943.

81. Tschofenig, H. and T. Fossati, *Transport layer security (tls)/datagram transport layer security (dtls) profiles for the internet of things*. 2016.

82. Priyadarsini, M. and P. Bera, *Software defined networking architecture, traffic management, security, and placement: A survey.* Computer Networks, 2021. **192**: p. 108047.

83. Malcomson, S., *Splinternet: How geopolitics and commerce are fragmenting the World Wide Web*. 2016: OR books.

84. Shao, J. and J. Zhang. *Bottlenet++: An end-to-end approach for feature compression in device-edge co-inference systems*. in *2020 IEEE International Conference on Communications Workshops (ICC Workshops)*. 2020. IEEE.

85. Sarkar, S., et al., *Security of zero trust networks in cloud computing: A comparative review.* Sustainability, 2022. **14**(18): p. 11213.

86. Shaikh, J.A., et al., *A UAV-Assisted Stackelberg Game Model for Securing loMT Healthcare Networks.* Drones, 2023. **7**(7): p. 415.

87. Refaee, E., et al., *Secure and scalable healthcare data transmission in IoT based on optimized routing protocols for mobile computing applications.* Wireless Communications and Mobile Computing, 2022. **2022**: p. 1-12.

88. Vaiyapuri, T., A. Binbusayyis, and V. Varadarajan, *Security, privacy and trust in IoMT enabled smart healthcare system: A systematic review of current and future trends.* International Journal of Advanced Computer Science and Applications, 2021. **12**(2).

89. Kumar, B., et al., *The Role of IOT and Cyber Warfare in Developing the Health Care Devices.* Mathematical Statistician and Engineering Applications, 2022. **71**(3s): p. 1185-1194.

90. Talaminos-Barroso, A., J. Reina-Tosina, and L.M. Roa, *Adaptation and application of the IEEE 2413-2019 standard security mechanisms to IoMT systems.* Measurement: Sensors, 2022. **22**: p. 100375.

91. Jan, S.U., et al., *Secure patient authentication framework in the healthcare system using wireless medical sensor networks.* Journal of Healthcare Engineering, 2021. **2021**.

92. Awotunde, J.B., et al., *Privacy and security concerns in IoT-based healthcare systems*, in *The Fusion of Internet of Things, Artificial Intelligence, and Cloud Computing in Health Care*. 2021, Springer. p. 105-134.

93. Arul, R., et al., *Multi-modal secure healthcare data dissemination framework using blockchain in IoMT.* Personal and Ubiquitous Computing, 2021: p. 1-13.

94. Rbah, Y., et al. *Machine learning and deep learning methods for intrusion detection systems in iomt: A survey*. in *2022 2nd International Conference on Innovative Research in Applied Science, Engineering and Technology (IRASET)*. 2022. IEEE.

95. Ghubaish, A., et al., *Recent advances in the internet-of-medical-things (IoMT) systems security*. IEEE Internet of Things Journal, 2020. **8**(11): p. 8707-8718.

96. Sahu, A.K., S. Sharma, and D. Puthal, *Lightweight multi-party authentication and key agreement protocol in iot-based e-healthcare service.* ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 2021. **17**(2s): p. 1-20.

97. Ksibi, S., F. Jaidi, and A. Bouhoula, *A Comprehensive Study of Security and Cyber-Security Risk Management within e-Health Systems: Synthesis, Analysis and a Novel Quantified Approach.* Mobile Networks and Applications, 2022: p. 1-21.

98. Sadiku, M.N., et al., *Emerging Technologies in Healthcare*. 2021: AuthorHouse.

99. Agrawal, P., et al., *Trust-Based Communication Systems for Internet of Things Applications*. 2022: John Wiley & Sons.

100. Alyahya, S., et al., *Cyber secure framework for smart agriculture: Robust and tamper-resistant authentication scheme for IoT devices.* Electronics, 2022. **11**(6): p. 963.

101. Subramanian, H. and P. Raj, *Hands-On RESTful API Design Patterns and Best Practices: Design, develop, and deploy highly adaptable, scalable, and secure RESTful web APIs*. 2019: Packt Publishing Ltd.

102. Rathod, D. and S. Patil, *Security analysis of constrained application protocol (CoAP): IoT protocol.* International Journal of Advanced Studies in Computers, Science and Engineering, 2017. **6**(8): p. 37.

103. Ghirardello, K., et al. *Cyber security of smart homes: Development of a reference architecture for attack surface analysis*. in *Living in the Internet of Things: Cybersecurity of the IoT-2018*. 2018. IET.

104. Huh, J.-H., *Reliable user datagram protocol as a solution to latencies in network games.* Electronics, 2018. **7**(11): p. 295.

105. Morais, D.H. and D.H. Morais, *5G Transport Payload: Ethernet-Based Packet-Switched Data.* 5G and Beyond Wireless Transport Technologies: Enabling Backhaul, Midhaul, and Fronthaul, 2021: p. 19-31.

106. Elnourani, M., S. Deshmukh, and B. Beferull-Lozano, *Distributed resource allocation in underlay multicast D2D communications.* IEEE Transactions on Communications, 2021. **69**(5): p. 3409-3422.

107. Bayılmış, C., et al., *A survey on communication protocols and performance evaluations for Internet of Things.* Digital Communications and Networks, 2022. **8**(6): p. 1094-1104.

108. Islam, M.M., et al., *Internet of Things: Device Capabilities, Architectures, Protocols, and Smart Applications in Healthcare Domain.* IEEE Internet of Things Journal, 2022. **10**(4): p. 3611-3641.

109. Al-Kashoash, H.A., et al., *Congestion control in wireless sensor and 6LoWPAN networks: toward the Internet of Things.* Wireless Networks, 2019. **25**: p. 4493-4522.

110. Cerullo, G., et al., *Iot and sensor networks security*, in *Security and Resilience in Intelligent Data-Centric Systems and Communication Networks*. 2018, Elsevier. p. 77-101.

111. Dizdarević, J., et al., *A survey of communication protocols for internet of things and related challenges of fog and cloud computing integration.* ACM Computing Surveys (CSUR), 2019. **51**(6): p. 1-29.

112. Singh, N. and A.K. Das, *Energy-efficient fuzzy data offloading for IoMT.* Computer Networks, 2022. **213**: p. 109127.

113. Sadhu, P.K., et al., *Prospect of internet of medical things: A review on security requirements and solutions.* Sensors, 2022. **22**(15): p. 5517.

114. Sharifi, A., A.R. Khavarian-Garmsir, and R.K.R. Kummitha, *Contributions of smart city solutions and technologies to resilience against the COVID-19 pandemic: A literature review.* Sustainability, 2021. **13**(14): p. 8018.

115. Nandy, S., et al., *An intrusion detection mechanism for secured IoMT framework based on swarm-neural network.* IEEE Journal of Biomedical and Health Informatics, 2021. **26**(5): p. 1969-1976.

116. Hireche, R., H. Mansouri, and A.-S.K. Pathan, *Fault Tolerance and Security Management in IoMT*, in *Towards a Wireless Connected World: Achievements and New Technologies*. 2022, Springer. p. 65-104.

117. Howell, D., et al., *Patient-reported outcomes in routine cancer clinical practice: a scoping review of use, impact on health outcomes, and implementation factors.* Annals of Oncology, 2015. **26**(9): p. 1846-1858.

118. Tootoonchian, A., et al. *{ResQ}: Enabling {SLOs} in Network Function Virtualization*. in *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*. 2018.

119. Yan, D., et al., *IEA EBC Annex 66: Definition and simulation of occupant behavior in buildings.* Energy and Buildings, 2017. **156**: p. 258-270.

120. Gazis, V., et al. *A survey of technologies for the internet of things*. in *2015 international wireless communications and mobile computing conference (IWCMC)*. 2015. IEEE.