International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 15 (2023) pp. 201-213
© MIR Labs, www.mirlabs.net/ijcisim/index.html

Received: 3 Nov.2022; Accepted: 10 May, 2023; Published: 9 June, 2023

Modeling IoT based Forest Fire Detection System with IoTsec

Meziane Hind[1]*, Ouerdi Noura[1] and Ajith Abraham[2]

[1] ACSA Laboratory, Faculty of Sciences, Mohammed First University,
Oujda, Morocco,
wwhindmeziane94@gmail.com

[2] Faculty of Computing and Data Science, FLAME University, Pune, India
ajith.abraham@ieee.org

***Abstract*: The Internet of Things (IoT) has become a real technological revolution in different sectors starting from body sensors to professional eras. The current growth of the IoT field and its use in multiple domains attracts the attention of attackers. However, this technology creates new security issues. Security is frequently critical and demands cybersecurity specialists and the IT community for looking for a reliable solution. Nowadays, forest fires have become the most widespread around the world targeting the ecosystem (trees, plants, animals, and people). Therefore, designing and modeling an IoT-Forest Fires Detection System is a real challenge. To overcome this challenge, UML is a resource for representing IoT systems in different views. In this context, the IoT has become a real technological revolution that is increasingly used in several fields. However, security, fault tolerance, real-time are the specific problems of an IoT based Forest Fire Detection System. The Forest Fires Detection System is another important service that IoT offers several opportunities to monitor, control and collect data. Forest fires can undoubtedly destroy the ecosystem. Despite its rapid spread, security of forests faces many issues, like the confidentiality and integrity of data, and the functionality and availability of equipment (such as sensors). The goal is to focus more on extensions rather than languages. It is rather imperative to compare these extensions in order to choose the best and most effective UML extension for IoT security modeling. We used a UML extension called IoTsec to model an IoT based Forest Fire Detection System through a use case diagram. This work aims to ensure the security and safety of the proposed system against attacks exploiting the vulnerability of the system.**

***Keywords*:** IoT Systems, Security Modeling, Fire Forest detection, UML extensions, IoTsec, Security Requirements**.**

I. INTRODUCTION

This work presents the extension of the conference paper presented at the 22nd international conference of on Intelligent Systems Design and Applications (ISDA) [1] with an improvement of results. The actual work is an occasion to highlight the application domain and to perform a concrete case which is the forest fires detection.

The IoT refers to a system of devices connected to the Internet with the ability to exchange and collect data from environment with no human intervention [2][3]. The IoT are increasingly used in several fields with critical data that require security. IoT has applications in every domain starting from personal needs to professional eras. Among the innovations in terms of IoT, there are forest fires detection in which user can protect the environment from some bad case or any disaster like fire, user can control temperature using the tablet, computer, or phone. The issue of finding reliable methods and techniques for detecting forest fires to protect the ecological balance from such threats must be urgently discussed by security specialists and developers.

Forests must be preserved because they contain timber and minerals, filter water and air, help regulate the climate, and are home to animals, etc. However, wildfires can lead to several consequences such as killing and injuring animals and people, burning tracts/acres of land, etc. In addition, wildfires generate 30% of the CO2 (carbon dioxide) in the atmosphere [4][5][6]. In forests, fires start with some flammable plants and grow rapidly with wind and high temperatures. One reason for the fires is human inattention, human errors or bad human behaviors. The other reason is the natural events (because of the heat produced by the sun (burning branches), global warming, ...). The fires caused by human inattention can result from multiple reasons, such as leaving unattended fires in the forest or throwing a burning cigarette, etc. The detection of forest fires may be better by detecting it in a real-time.

Forest fires detection is an IoT application that needs a smart environment. In this context, security breaches in this area can have disastrous and serious consequences. For example, if the sensors deployed in such area start collecting Temperature data falsely, then it will cause material and human losses. In this example, data confidentiality and integrity must be ensured. Because the attacker focuses only on the useful data.

Therefore, and due to the widespread use of IoT systems dedicated to detecting forest fires, it is crucial/necessary to secure them. One of the goals of this system is to detect fires based on IoT in an early stage and real time to minimize damages and losses. I have chosen this application domain because of its importance and its dangerous link to the life of animals, plants and humans in those dangerous areas. The contribution of this study is therefore to ensure the safety of the system in an earlier stage.

The forest fires detection system is basically about early fire detection, real time monitoring and early-warning systems which will help improve accuracy and trustworthiness. For that reason, we need to build/develop a performant IoT based forest fires detection system for monitoring and controlling. Therefore, drones could be used for forest fires detection system. Drone is an IoT device with a camera that can be used for fire detection. For early detection of fires and real-time monitoring, drones equipped with cameras and specific sensors can be used for an efficient solution and cost-effective [4].

In this research, we try to use UML extensions that can model IoT system or security, for example, IoTsec, SysMLsec, UML4IoT, etc. The proposed contribution is particularly interested in this axis. Before choosing the tool or UML extension on which we are going to work, we need a detailed analysis of the whole IoT architecture which was provided in [3]. The goal was to look for the security modeling in these layers. To choose the layer(s) on which we must interest, a study on the IoT layers was made including structure and explanation (at the cloud level, at the object level, etc.). The choice of the layers that need more security in the IoT architecture was based on a detailed study provided in [3], this choice was justified by an in-depth analysis of the trust and security concerns in different layers. The comparison in terms of layers was based on several parameters such as [3]: security challenges and problems, vulnerabilities in each layer of IoT systems, security requirements for IoT layers and security threats analysis.

In IoT systems, configuration [2], implementation [2], or design-related flaws [3] are the exploited vulnerabilities. For this reason, security process must be integrated into the whole IoT systems during the design phase. Before starting the implementation phase, the system must first be validated during the design phase. The verification process included (1) checking that the system has no vulnerabilities that could allow an attack, in addition to (2) ensuring that the system specifications are satisfied. Therefore, the contribution consists in modeling IoT-Forest Fires Detection System, based on IoTsec and using the SysML requirement diagram.

A. *Originality and objectives*

The main aims of this current work are as following:

- At the beginning, we analyze each UML extension separately (UMLsec, IoTsec, IoTReq, SysML4IoT, etc.) to outline the best UML extension for IoT security modeling.
- This paper proposes the requirements for modeling the security for IoT-Forest Fires Detection System.
- Then, it illustrates modeling IoT based Forest Fires Detection System.
- Modeling and designing security issues in IoT based Forest Fires Detection System based on UML extension are also detailed.
- This paper also illustrates the conception of the secure physical layer and the network layer of the IoT architecture for the proposed system (forest fires detection) including 2 solutions.

B. *Outline*

The remainder of this paper is organized as follows: At first, we make a study on UML extensions to differentiate between them. For that, in the second section, we approach some state of the art regarding the study of languages/extensions and related works. The third part covers the application domain which is the forest fires detection system. The fourth section discusses the results and funding that we have had. Finally, we sum up the paper with a conclusion.

II. State of the Art

This second section presents the state of art with relevant concepts to underline the language/tools we are working on and the related works to make a study on IoT and UML extensions to differentiate between extensions that allow us to model IoT systems and extensions which lead us to model only security.

A. *Comparison between the different UML extensions*

Most authors work on the languages, but we have not found work that compares all the extensions, which is the strength of this paper. To be updated with new UML extensions used in the literature, we started by comparing these approaches. Many languages and extensions are available in the literature. Due to UML limitations [1] for describing relationships and creating graphical diagrams, different UML extensions were created to allow modeling of IoT systems and to give developers a useful nomenclature.

The modeling of IoT systems is a necessary. Hence, modeling IoT systems can be categorized into two main parts: General modeling language (e.g., SysML, ...) and Security modeling language in the IoT domain (e.g., IoTsec, ...).

The goal is to focus on extensions. *So, what is the UML extension able to design an IoT security systems effectively?* The main aim of this subsection is to present an overview on the existing UML extensions by giving a brief paragraph to criticize these tools to compare the differentiation between them. To achieve this goal, we took several UML extensions, and we found that IoTsec have the best IoT security modeling.

- **UMLsec** [7][8] is an UML extension for deploying secure systems. It has twenty-one stereotypes addressed to security concerns, they encapsulate knowledge about prudent security engineering and thus make it accessible to developers who may not be specified/specialized in security. However, this UML extension does not enable us to model IoT systems.
- **SysMLsec** [8]: is a new SysML environment that presents diagrams for security issues and an associated methodology. The same lack of UMLsec occurs in SysMLsec.
- **UML4IoT** [9]: is based on the use of a UML profile required for cyber-physical components to be integrated into the IoT. However, this UML extension does not enable security modeling.

- **SysML4IoT** [10]: It is very useful for IoT applications. The same lack of UML4IoT occurs in SysML4IoT.
- **IoTReq** [11]: is based on the use of a UML profile for modeling the system's domain. IoTReq describes a framework to orient the challenges posed by the use of the IoT in the product development process.
- **IoTsec** [8] is a subset of UML and SysML. It is an UML extension for IoT systems security modeling. It applies UML/SysML diagrams, UML stereotypes, UMLsec stereotype mechanisms. Figure 1 shows where IoTsec is among other approaches, it mainly extends to UML, and then it also extends to SysML and includes some suggested stereotypes in UMLsec. The relationship between SysML and UML is shown in Figure 2. The UML extension is suggested to guide developers throughout the design life cycle of IoT systems, and this is in regard to security requirements at each stage. IoTsec suggests a graphic representation of security modules, a nomenclature that encapsulates the IoT security issues and UML diagrams extensions. The benefit of this extension/method is that it leads us to model IoT security. It also describes security in the best way.

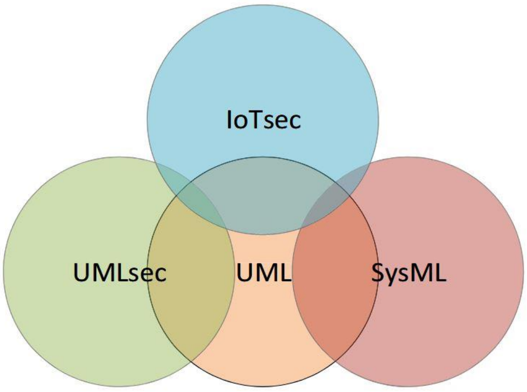

Figure 1. IoTsec among other approaches [8]

The figure “Figure 2” shows the relationship between SysML and UML language. UML is a popular technique for documenting and modeling systems. It is a standard in object-oriented modeling which allows to address many needs of Systems Engineering. Indeed, SysML presents the advantage of expressing constraints. SysML is based on UML 2.0, it is a language derived from UML, unlike UML limitations for modeling physical constraint like battery life and energy consumption [1], as well as expressing relationships. Comparing to UML language, it satisfies the needs for modeling IoT security. Figure 2 bellow shows a Venn diagram that presents the UML/SysML relationship.

- UML4SysML (UML reused by SysML): marked by the intersection of UML and SysML circles which means that SysML reuses UML modeling constructs.
- UML not required by SysML: means a part of UML 2 which is not required for a SysML implementation.
- SysML's extensions to UML: means new modeling constructs for SysML that replace UML constructs.

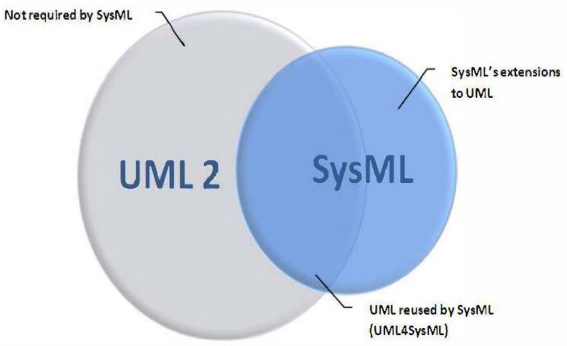

Figure 2. Relationship between SysML language and UML [12]

The comparison between these tools often raises the following parameters:

- IoT modeling
- Security modeling

It will be interesting to know that IoTsec method/tool answers all these parameters/questions (see Section 4 Table 5). For that, we proposed to work with IoTsec.

B. Related Works

In this section, we give you an idea about current and similar research on this subject. In order to determine the present contribution, we must see the existing in this field of IoT security modeling. For that, the first constraint was to find the layer that must be specified and to choose the tool or the UML extension that must be specified. The second constraint was to know which is the best language and the best diagram that suit us in terms of security. Their answers led to many valuable insights regarding the language side and layer side that lead to ensuring the security of IoT systems.

Therefore, several researchers have worked on modeling with UML/SysML languages. But, in this research, we prefer to work with extensions instead of languages.

In [3], 2022, Meziane et al., proposed an IoT architecture based on four layers. The interest of this study was to choose the layer to be modeled. This paper first introduces the state of the art of modeling IoT security systems, and marked the Physical and Network layers as the layers that need more IoT security. The idea of the proposed architecture for IoT is to make a comparison in terms of layers. Furthermore, the author presented the system architecture for modeling then the physical layer and network layer in general to prove the limitations of UML. The differentiation between UML extensions was based on two parameters: IoT systems, and security. Moreover, it provides a good background for a IoT security aspects. To secure IoT as whole, we need to secure the layers and the communication between these layers, especially the physical and network layers. The author proposed to work with IoTsec. It was considered as the best one that can model security in IoT environment/systems compared to UML4IoT, UMLsec, SysML4IoT, SysMLsec.

The research [1], performs the comparative study on UML and SysML. The aim of using UML is to compare this language with an informal modeling language called SysML language, in order to discover which is the efficient one for modeling IoT security. This paper starts by presenting a general overview on UML and SysML. It also provides the security requirements for the physical and network layers. In addition, we try to model the findable IoT layers using two diagrams. The results obtained in [1] have shown us that the use of SysML in our case is a good solution and marked Requirement diagram as the best diagram among SysML diagrams for IoT in terms of security. The benefit of this language is that it does model the system, the requirements, and the traceability between system and requirements. Furthermore, this paper presented a good background for proposing and presenting a new IoT security modeling. To conclude, *Why SysML is better compared to UML?* requirements diagram is better compared to use case diagram. In other words, the requirements diagram compared to the use case diagram is more expressive in terms of design.

However, in [1][3] authors focused more on the languages rather than the extensions themselves which is not the objective of this work. Indeed, we did not take a very concrete example and we base ourselves on it, we talk in general. The objective was to have a general model and then we will take a practical case. Therefore, the main aim of this paper is to model a reality which is wildfire.

Yandounzi et al., [4] conducted a review of current advancement in forest fire detection and monitoring using both deep learning techniques and unmanned aerial vehicles (UAVs), or drones. They conducted a comprehensive analysis of the latest developments in deep learning object detection, including Region-based Convolutional Neural Network (R-CNN), You Only Look Once (YOLO), and its variants, with a focus on their application in forest fire monitoring. The use of drones equipped with sensors and cameras offers an efficient and cost-effective solution for detecting fires in real-time. Furthermore, there are research works which use new approaches for monitoring and detecting wildfires as Grari's work [5] that focused on the use of advances in ML (Machine Learning), CV (Computer Vision), and remote sensing technologies. For predicting wildfires, they proposed an approach based on ML. A regression model was used to train over NASA's FIRMS dataset (fire information for resource management system) to predict in megawatts fire radiant power. The obtained simulation results shows that the ensemble learning is an effective model for predicting wildfires.

Yandounzi et al., [6] conducted a review on wildfires detection and prediction using DL (deep learning) and drones. The combination of DL and drones which is a key foundation that can be used for detecting forest fires using images with high accuracy.

In [8], authors utilized IoTsec. They also presented methods that employ UML and its extensions. The benefit of this extension is that it does not model only a security, but rather an IoT systems. Thus, they compare extensions and languages including UMLsec, SysML, SysMLsec, UML4IoT, ThingML, UML, IoTsec. However, SysML4IoT and IoTReq extensions were not mentioned. That is why, this work collaborated a comparison between all these extensions.

Authors in [9], utilized UML4IoT, that integrate IoT environment and the CPS. UML4IoT is a UML based approach to exploiting model-driven engineering in the development.

In [11], the authors explored UML as visual language to represent IoT systems with the suggested extensions.

The authors in [13] suggested IoTReq method which is a UML extension. IoTReq is an UML modeling method for modeling the IoT domain.

In [14], the authors worked on ThingML. ThingML is a modeling language. It is a methodology and a set of tools designed for IoT and Cyber-Physical Systems (CPS). It models the complete behavior of components. Although that ThingML language models the IoT systems, it is not yet confirmed that this language would be useful for security modeling of IoT systems.

Each researcher/author has his own approach and method. Several researchers focus on modeling languages like SysML, ThingML, and UML, however, there is not enough work/research on the use of UML extensions (e.g., SysMLsec, IoTsec, etc). Table 1 summarizes the contributions of the previous surveys on modeling IoT security.

Table 1. Related surveys work on modeling IoT security.

Study	Year	Contributions
[1]	2023	A Comparative Study for Modeling IoT Security Systems
[3]	2022	A Study of Modelling IoT Security Systems with Unified Modelling Language (UML)
[4]	2023	Investigation of Combining Deep Learning Object Recognition with Drones for Forest Fire Detection and Monitoring
[5]	2022	Early wildfire detection using machine learning model deployed in the fog/edge layers of IoT
[6]	2022	Review on forest fires detection and prediction using deep learning and drones
[8]	2017	IoTsec: UML extension for Internet of things systems security modelling
[9]	2016	UML4IoT—A UML-based approach to exploit IoT in cyber physical manufacturing systems
[13]	2018	A UML-based proposal for IoT system requirements specification
[14]	2016	ThingML: a language and code generation framework for heterogeneous targets

III. Application Domain: Forest Fires Detection System

The study of IoT security modeling is very difficult to conduct because there is a lack of standardization of a language at the modeling level. At the beginning, we pursued the following research questions:

- At the security level, which layer should we specialize? This research question was the first step of the work.
- At the modeling level, which is the effective language for IoT security modeling? Which diagram is the best in term of security? and what is exactly the UML extension that enable us to model IoT security? This research questions were the second step of the work.

According to deep studies [1][3] on IoT security modeling, I got the following results: The **Physical and Network layers** necessitate more modeling in term of security. **SysML** is the best language, **Requirements diagram** is the best in term of security, **IoTsec** is the efficient one which allow us to model IoT security systems. The main objective was to obtain the best results as well as the best IoT security modeling.

Keeping the results of these two studies [1][3] and as mentioned before, our objective is to illustrate these results with a clear application domain of an IoT system which is a Forest Fires Detection System. Designing and modeling the security of the proposed system is a challenging task. Therefore, security breaches of the smart environment applications should be avoided.

This section aims at presenting an example of IoT system namely Forest Fires Detection System. So, we need to understand the domain of application we are working on. Then, we are trying to find solutions for this concept. It is true that the IoT based Forest Fire Detection System specifications are complex. Forest Fires Detection System is a system for monitoring and detecting 24/7 forest fire based on IoT. The detection of forest fires can be performed to monitor and to prevent from any damages or losses caused by natural/human resources.

The following subsections describes the steps used to achieve our objective which is designing an IoT-based forest fire detection system that really brings it all together (communication technologies, security requirements). To solve the problem, this paper suggested some requirements for forest fire detection system. The modelization were made by the IoTsec.

To model IoT systems, we firstly based on four layers architecture. This section highlights the proposed strategy that we followed to model the system. This paper provides an appropriate design for IoT based forest fire detection system by proposing a general model which will aim to take into consideration all the security aspects. The proposed system is made up of two parts: physical layer, and network layer.

A. Injection of Model to System

The goal is to inject the proposed model [3] to forest fires detection system. According to [3], IoT architecture consists of four layers (Physical, Network, Cloud/Middleware and Application layer). Each one has its security threats and vulnerabilities. According to a detailed analysis of the IoT architecture, I found that we should specify to the physical and network layers due to multiple reasons among which limited resource, enormous/huge heterogeneity and compatibility problem. All these two layers have security issues and problems specific to them.

To design the proposed system, we need first, to offer the system requirements. The goal is to make a general model that will take into consideration all the security aspects. IoT security modeling is still challenging. Therefore, security requirements within an IoT based Forest Fires Detection System should be considered by researchers, IoT developers and security specialists. The findable layers are physical and network [3]. So, the proposed system will be made up only of two parts: physical layer, and network layer. Because on the middleware layer or cloud side, we do not have the challenges of the physical and network layer. No problem on the cloud side, the problem is in the sensors.

By gathering all the different results [2][3][15], so, the main aim is to try to inject a 2-layer architecture for an IoT based that includes a physical layer (which collects data from IoT devices and transmits this data to network/next layer), and a network layer (which transmits real-time data within the network and incorporates various sources of data). To present the forest fires detection system, we firstly inject it to four layers architecture (see Figure 3). The components of the proposed system are represented in Figure 1. Figure 1 presents a real example that allows us to cite these different layers.

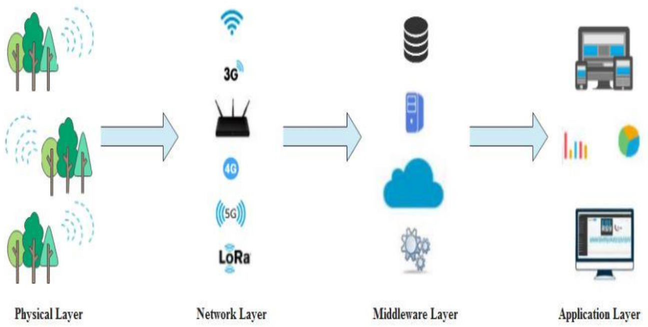

Figure 3. Proposed model for IoT based forest fires detection system monitoring.

The collected data by several sensors are sent over an internet connection using communication networks and protocols (MQTT protocol, etc.). Then, the IoT data is stored on the cloud for processing and analyzing in order to be pushed to the next layer. The IoT computing provides data that the user can visualize in an online dashboard with a flexible application interface. e.g., these data can be used and accessed by applications. If a fire is occurred, a notification or an alarm will be instantly sent to users (fire department) (see Figure 1). Cloud computing analyzes different types of data collected. The components of the physical layer include sensors, actuators, etc. Sensors is small equipment with limited resources [3] (CPU resource, storage resource, etc.). The components of the Network layer include Internet, gateways, routers, etc. IoT ecosystem generate and exchange continuous and enormous sensitive data on different layers.

As mentioned before, there are no problems in the Middleware/cloud and Application layers. According to a comprehensive and deep study [3], the big challenges are in the Physical and network layers. For that, in the next two subsections, we are going to present and analyze each layer (physical and network) separately.

As said at the beginning, the cloud layer plays a very important role, the data is stored in the cloud. On the cloud side, we do the full analysis, a robust system (SIEM system) can be used to analyze the data for the extraction of information, so there is a value for decision making at the same time detecting anomalies. We need to check the correctness of the data. In other words, every SIEM system's concepts are to gather data from diverse sources, detect anomalies from the standard, and take action accordingly. If a possible issue is detected, a SIEM system can log further information, generate an alert, and inform other security checks to stop an event.

On the application side, this layer delivers graphs. The graphs in the application layer aims at showing us if there is any anomaly/attack/abnormalities/deviants/intrusion/outlier (value very different from other observations) in the physical layer.

This section provides the security requirements for two layers (physical and network) in the IoT environment.

1) *Physical Layer*

Today, billions of devices and connected objects are online. The lack of security in this layer is mainly related to the resource's limitation of IoT devices in the application of security mechanisms. All these constraints depict new security challenges for IoT designers. So, the question is: *how to be sure that the equipment's and hardware behave correctly?* and *how to protect these equipment's and hardware against these attacks?* A trivial solution is to make a general and meaningful model that will take into consideration all the security aspects.

Regarding the IoT devices including equipment's, physical devices and hardware, security must be implemented in the design phase because it is difficult to integrate it once the equipment is achieved.

Wildfires are detected using imagery including (airplanes, satellites, and drones). To detect forest fires, drones are preferable to satellite imagery [6] since the cost of using drones is lower than the cost of using satellites, moreover, drones can send images every day, while satellites may send images once every few days/weeks, furthermore, drones can collect more accurate data and fly low to detect small fires. Another way to detect it is to use sensors. On the physical layer, billions of sensors, actuators, and devices are online today. This consists of different types of sensors distributed on the forest, e.g., heat/temperature, humidity, gases, pressure, wind speed, NH3, CO, O3, CO2, smoke fire sensors, etc. These implemented sensors on the forest or around it is responsible for gathering environmental data or to take measurements constantly. The use of these sensors is for monitoring faster and better detection of fires. Then, these sensors report fire parameters to a remote service center. In other words, the data gathered by sensors need to be transmitted to the cloud to be analyzed and processed. Therefore, these acquired data should be sent to the cloud with encryption. In practice, technologies deployed in the sensing layer are probably the most rapidly evolving among the different layers and is therefore discussed in detail in the following subsection (Table 2 and Table 3).

2) *Network Layer*

Communication standards are important for enabling IoT devices to communicate on the Internet. Based on [2][15], communication in IoT is categorized into two types:

- Short range wireless communication technologies (e.g., RFID, NFC, WSN, 6LowPan, Z-Wave, Zigbee, Bluetooth, Wi-Fi, BLE) and
- LPWAN (Low-power wide area network) or Long-Range Technologies (e.g., Sigfox, LoRa/LoRaWAN, NB-IoT, 2G/3G/4G/5G).

These IoT communication technologies are also classified into three categories including:

- Short Range technologies,
- LPWAN (Sigfox, LoRa/LoRaWAN, NB-IoT), and
- Cellular Communication (2G/3G/4G/5G) [3].

A comparison of these communication standards based on [16] is presented in Table 2. The purpose of this comparison is to present the new technologies used as well as their added values in order to choose the best communication technology that is currently widely used in IoT systems in general (as example fire detection).

In the connectivity layer, different communication technologies (e.g., Wi-Fi, Zigbee, Bluetooth, LoRa, etc.) are used to connect and communicate devices with each other. However, each communication technology has its own advantages and disadvantages. Therefore, the proposed comparison is based on the data rate, distance, power efficiency, reliability, cost, service [16], as well as advantages and disadvantages.

Table 2. The different communication protocols used by the IoT systems.

	Data rate	**Distance**	**Power efficiency**	**Reliability**	**Cost**	**Service**	**Advantages**	**Disadvantages**
Wi-Fi	Approx. 54 Mbps (High)	Approx. 50 m	Medium	Medium	Low	Home IoT, Smart cities, office IoT,	Very high data rates, Very simple setup	It requires a lot of power and it cannot be useful for applications that need long battery life.
Zigbee	Approx. 250 kbps (Low)	Approx. 100 m	Low	High	Low	Tracking home automation, Indoor asset	Low cost, Low-power consumption	Short range
BLE	Approx. 0.27 Mbps (Medium)	Approx 100m	Low	High	Low	Smart connected devices and Wearable devices	Very easy to use, Low-power, Very low data rates.	Limited in range
NFC	Approx. 42 kbps (Medium)	Approx. 20 cm	Very Low	High	Very Low	Contactless payment transaction, local asset tracking	Easy data exchange	Information leakage
4G LTE H	Approx 12 Mbps (Low)	Large	High	High	High	Transportation, Agriculture, industries, fleets	High speed	Connectivity is limited to certain specified regions and carriers;
LoRa (Long Range)	Approx. 50 kbps (Low)	Several miles	Low	High	Medium	Smart city, supply chain management, energy management	Long distance connectivity	
Sig Fox	Approx. 1 kbps (Very Low)	Several miles	Very Low	High	Medium	Smart meters, environmental sensors,	Easy connectivity, low cost	SigFox provides no valuable collision avoidance mechanisms

To answer this question: *What is the best communication technology that is currently widely used in IoT systems in general (as example in fire detection)?* we need a comparison between the different technologies to choose the best technology. The fires as example, at the level of the Sidi Maafa Forest, they are distant not short range but long range. So, we need one of the technologies that are used among LoRa, LoRaWAN, Sigfox, etc. we prefer LoRaWAN technology, because:

- (1) Long range,
- (2) LoRaWAN is an open and standardized LPWAN technology. LoRaWAN is a long-range wireless protocol. Any LPWAN must incorporate the necessary security. LoRaWAN offers 2 simple layers of security: one is the network specific and the other is application specific. The first ensures that the node is authenticated with the network server, and the second ensures that the network operator cannot access the application data of the end user. The main properties of LoRaWAN security are integrity protection, confidentiality, and mutual authentication.
- (3) LoRaWAN connect several objects using LoRa technology. It uses minimum battery consumption.
- (4) LoRaWAN uses bidirectional communications between the base station and equipment.

Therefore, LoRaWAN is the best technology in IoT systems in general and fires detection. LoRaWAN is a credible wearable/portable wireless technology.

To exchange data between IoT devices, several application protocols are used like MQTT (message queue telemetry transport), AMQP (advanced message queuing protocol), XMPP (extensible messaging and presence protocol), CoAP (constrained application protocol). According to [16], authors were based on QoS (quality of service); SSL (secure sockets layer); TCP (transmission control protocol), etc. to compare the different protocols for application-level messaging.

Table 3. Comparison of different protocols for application-level messaging [16].

Protocol	**Restful**	**Trans. layer Protocol**	**QoS**	**Architecture**	**Security**	**Header Size (Bytes)**	**Sync**
CoAP	Yes	UDP	Yes	Pub-Sub, Req-Res	DTLS	4 (minutes)	async/sync
XMPP	No	TCP	No	Pub-Sub, Req-Res	SSL	—	async
MQTT	No	TCP	Yes	Pub-Sub	SSL	2	async/sync
AMQP	No	TCP	Yes	Pub-Sub	SSL	8	async
HTTP	Yes	TCP	No	Req-Res	SSL	—	async

Based on Table 3, and on [17][18] MQTT has the higher QoS/reliability compared to other protocols, it is the most preferred publish-subscribe (Pub-Sub) lightweight messaging protocol. It is the popular IoT protocol that can be used on unreliable networks. It is currently used in a broad range of IIoT and IoT sectors. It is also lightweight message transport for IoT pub/sub.

B. Requirements for Modeling the Security for IoT based Forest Fires Detection System

To design an IoT security system, multiple requirements are needed [3]. For example, in the physical layer, security requirements include [3] Confidentiality, Integrity, Availability, Authentication, Authorization. In the network layer, security requirements include [3] Confidentiality, Integrity, Availability, Authentication, Authorization, Non-repudiation, Privacy, Compatibility. The concerns give us a clear idea about the security requirements that we need as a security remedy in a specific layer. In this part, we also defined the security requirements for two layers (physical and network) in the IoT environment, as described and detailed in the following subsections.

1) Physical Layer

Therefore, we need to give a good expression and modeling of requirements. *Installation of an IDS is it feasible in these equipments or it is not possible?* Indeed, if the collected data from IoT devices are transmitted to the next layer without any encryption, then, this gathered data will be altered and manipulated. In addition, the IoT devices should be legit and trustworthy. Thus, data tampering must also be avoided.

To improve IoT security system modeling, this paper suggests in the following new requirements. Thanks to SysML language, we were able to express the requirements. i.e., thanks to its diagrams, SysML provides tools and concepts which make easier the expression of requirements. The proposed system requirements include:

- *Confidentiality*: Only authorized person must be able to access other’s data. The attacker does not aim to modify the information. They wish to obtain data.
- *Integrity*: The sensor can give false indications that may endanger actions. As an example, the hacker may falsify the data collected by IoT devices or sensors. For that, the integrity of the data collected by the sensors in the IoT system is crucial.
- *Sensor tampering*: The attackers may get access to the sensors physically, due to unfriendly environments deployment. Therefore, we should take into account the tampering protection. So, the vision or the goal/objectives of the attacker (who launch an attack) should be considered.
- *Resistance*: the sensors must be discreet, otherwise the sensor will be stolen. i.e., these sensors must not be physical (for animals). So, a cartographic study of the forest is necessary [1];
- *Performance*: for energy limitations, the battery must be charged/rechargeable for a specific period of time (6 months for example), it must save as much energy as possible. It is very important for the IoT devices to keep the battery for a longer time/duration for an application where it is not easy to recharge for example sensor in a disaster place or in river, or rainfall, Wind [1];
- According to the estimation of statistics [19], the profits of IoT connected devices have reached more than 1 billion dollars and are expected to grow to more than 1 trillion dollars by 2026. So, IoT devices should be secured and managed suitably.

In summary, the common security issues and problems in the physical layer can be handled using lightweight cryptography [1]. So, there are many functional objectives of the attack including attacks against availability, attacks against integrity and attacks against privacy. For instance, disrupting the sensor’s functionality, data tampering, sensor tampering, object tampering, security breaches.

2) Network Layer

At this level, IoT devices communicate wirelessly. The collected data are routed to the IoT gateways. In my opinion the best communication technology that is currently widely used in the IoT system in general is LoRaWAN.

- The behavior of the collected data must be defined in order to predict or detect anomalies and intrusions.
- Therefore, an early fire detection and real time monitoring on the Forest Fires Detection System must be performed. For that, the use of fully equipped drones with specific sensors and cameras is an efficient and cost-effective solution [4] for early fire detection and real-time monitoring.
- Implementing existing IDSs over the network layer will be a good solution. In other words, installing an IDS on Gateways is the best way to report immediately if any possible breaches occur on the perception/physical layer. But before installing any IDS, we first need to choose the most performant IDS among the existing open source IDSs. The choice depends on detecting all existing attacks and does not produce any true negative or false positive. In addition, the recommended IDS should be chosen depending on resource restriction and on CPU/Memory Consumption.
- A comparison between IoT communication technologies is carried out to have the best IoT technology that is already widely used in general for example in fire detection.
- Due to the existence of IoT communication technologies that are very vulnerable, so the goal is to propose a model that is secure and recommend some technologies.
- *Compatibility* [1]: The heterogeneity of IoT systems can be defined as a challenge related to different

communication technology. A compatibility is the major challenges because in IoT system. Therefore, we need to make certain that the different communication technologies can work with each other.

- *Network Security* [1]: we can assume that the IDS on the gateway layer is the best idea to detect intrusions and anomalies in the time it occurs. IDS need to be enhanced and efficient against intrusions.
- *Communication Protection*: avoid the use of communication technologies that are vulnerable.
- New techniques based on AI (Artificial Intelligence), particularly those related to machine/deep learning must be implemented.

In summary, the common security issues and problems in the network layer can be handled using an IDS. Therefore, a new 100% sure solution is needed, that will facilitate the decision against detected intrusions and according to their magnitudes and their intentions to put the necessary reaction in the right place. Among the techniques and methods used we opted for the choice of Artificial Intelligence (AI) and more precisely Machine Learning (ML) using NNs (Neural Networks).

Concerning communication in IoT systems, and based on Table 2, LoRaWAN (low-power long-range wide area network) is suitable for long range communication. Forest has langue distance, so for the link between them we will be based on LoRaWAN (i.e., long range, the battery has a duration). Therefore, LoRaWAN is the best communication technology for IoT system in general and for forest fires.

C. Modeling the Security of the Proposed System with IoTsec

In this subsection, we are going to work with an UML extension called IoTsec. The choice of using IoTsec was justified by the fact that is enable us to model IoT security. UML is a visual modeling resource which allows extensions to be used to represent these systems. Various extensions offer the possibility to model IoT systems, however these extensions are not all suitable for modeling IoT security systems. IoTsec is the only extension that enable us to model IoT security. In this subsection, we employed the IoTsec method as the UML extension. To model common actors, IoTsec uses extensions of UML for security encapsulated in a useful UML nomenclature and stereotypes. The purpose of this extension is to represent security issues/concerns with a visual notation. It proposes a nomenclature with security issues within each element. With this method, we managed to model the security of IoT based forest fires detection system.

SysML is based on a minimal subset of UML. The UML has fourteen diagrams in 2.5 version, the extended diagram used in this paper will be introduced. This section aims to model the proposed system requirements. The use of SysML language is advantageous, because it enables us to model the requirements. Indeed, SysML language satisfies the needs of SE (Systems Engineers). It also represents and relates requirements to the model of the IoT system.

Four actor's categories for the proposed system were identified: sensor, actuator, tag, and IoTdevice as shown in Figure 4. The nomenclature **TP** (**Tampering Protection**) in the physical means that the IoT devices are in great danger since these devices may deal with confidential and sensitive data. Therefore, these devices are intended to be inviolable.

- *Compromised device*: The attackers may get access to the sensors physically which could cause a loss of control of the system. More, IoT devices are more easily compromised. Compromised IoT devices may include sensor failure or compromised sensor that may lead to abnormal behavior.
- *Control of equipments*: Functionality of hardware and equipment [1]. In other words, *is the humidity/temperature/wind/speed sensors working or not [1]?* which means the mode of working or non-working of the equipments. If the sensor does not work, then, it will not detect fire.
- *Availability of sensors*: sensors need to be always available. i.e., the sensors must collect data on the times we need them to be collected.
- *Physical security*: the problem is in the sensors. The sensors must not be physical to animals and theft. Therefore, sensors must be discrete (under the earth), not just any location, so, a cartographic study of the forest is necessary [1]. Entries must invisible. These sensors contain small resources and controllable.
- *Confidentiality and Integrity of data*: The temperature sensor measure and gather information about the environment for the purpose of detecting fire. Therefore, confidentiality and integrity requirements are guaranteed if the content cannot be tampered by attackers. As an example, an attacker could tamper with sensor measurements to introduce false data into the system in order to tend actuations.
- *CIA security* [1]: CIA could be guaranteed in every IoT devices/system. Cryptography is the most known solution for considering the CIA security pillars (Confidentiality, Integrity, Availability). Therefore, an IoT device should cipher the data.
- For all these reasons, we should take into account the tampering protection (TP). TP aims at protecting the IoT devices against compromising. It encapsulates the CIA security (Confidentiality, Integrity and Availability) [1].
- In addition, connected objects have limited resources and if we think of cryptography, we choose lightweight cryptography and not RSA (which allows to consume energy and resources). So, we must propose algorithms which have already been proposed and which guarantee in fact the lightweight use of energy and resources, to be able to subsequently ensure the integrity and confidentiality. So, an IoT device should cipher the data. The problem of connected objects are resources and energy. The battery has a specific energy constraint.

For all these reasons, to enhance security, lightweight cryptography [1] is the proposed solution that should be implemented on sensors. Once these requirements are ready, we start the modelisation with the IoTsec tool. e.g., this modelization provided by Figures 4 and 5 is done through IoTsec tool by importing a useful nomenclature.

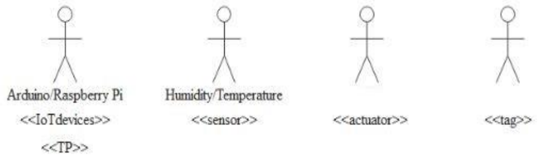

Figure 4. IoTsec actors for forest fires detection system

For the actors in use case diagram with IoTsec, developers may utilize each stereotype's category as seen in Figure 4. Based on [3] security requirements for physical layer are Confidentiality, Integrity, Availability, Authentication, and Authorization. i.e., an IoT device should authorize, authenticate, and cipher the data. Indeed, based on [3], CIA security could be guaranteed in every IoT devices or system. It is very important to have CIA security when implementing a solution to secure the IoT system. Therefore, cryptography is the most known solution for considering the CIA security pillars [3].

Since IoT devices carry sensitive and confidential information, there is a possibility that it could be misused if it is leaked. Therefore, security requirements in the physical layer must firstly guarantee data integrity and confidentiality. The confidentiality and integrity of data are crucial parts and two important factors of security. For that we need to focus on securing the two first parts of IoT architecture, which means that security needs to be improved in the physical and network layers. Hence, the accuracy of the data needs to be checked and verified in order to improve security in these two layers. In the physical layer, the data is being collected by sensors to be later transferred in the network. For that, to enhance security, we need to verify data integrity, and improve the accuracy and trustworthiness of the data collected by the sensors in the IoT system.

- *Confidentiality*: It must ensure that the exchanged data captured by sensors cannot be understood by the unauthorized entities.
- *Integrity:* It must ensure that the exchanged information was not falsified, tampered or altered by a third party.
- *Availability:* It must ensure that the sensors are not interrupted. IoT device availability [3] is highly necessary.
- *Authentication:* It must ensure that the entities involved in a process/operation are who they claim to be authorized.
- *Authorization:* It must ensure that entities have the control permissions required to perform the operation they are requesting to be performed.

The **Cipher** (C), **Authentication** (N), and **Authorization** (Z) elements are used as security requirements for IoT devices. C, N, and Z are depicted over the actor's head in a text box. The Table below presents use cases list for IoT-based forest fires detection system.

The nomenclature C, N, and Z are used in the IoT device use cases as shown in Figure 5. Table 4 shows use cases list for IoT-based forest fires detection system, it illustrates the modeling of IoT-based forest fires detection system functionalities [23-30].

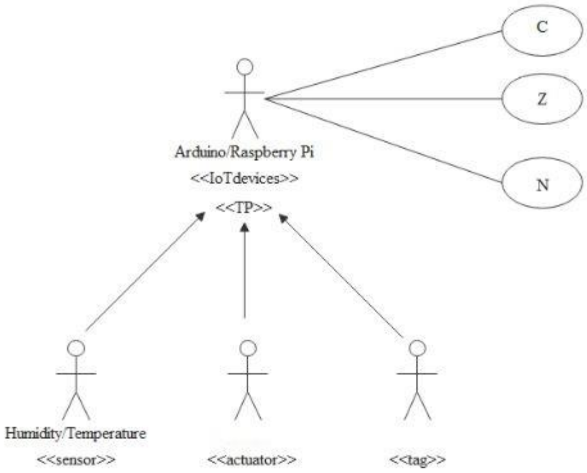

Figure 5. Use case diagram with IoTsec for IoT-based forest fires detection system

Table 4. Use cases list for IoT-based forest fires detection system.

Actors	Use Cases
The sensor	Implemented on the forest to constantly take measurements such as temperature, humidity, wind speed, etc. Sensors record and report specific data.
The actuator	Perform actions and generate the response
The Tag	Identification of things
IoTdevice	Authorization, authentication, and encryption

IV. RESULTS AND DISCUSSION

This section discusses and treats the results and findings which we have had. Based on the parameters/points of security modeling and IoT modeling, we found that IoTsec is the best compared to other extensions. This approach is relevant because it enables IoT security modeling. The criteria were relevant in order to outline the effective UML extension for IoT security modeling.

In this paper, we thought about IoTsec extension to meet our goals in term of IoT security. Thanks to this method, we could model IoT security systems presented in "Figure 4" and "Figure 5". In this study we chose to compare six different UML extensions (UMLsec, Sys-MLsec, UML4IoT, SysML4IoT, IoTReq and IoTsec). Each one can be applied differently depending on two parameters. The choice of the parameters was fulfilled based on our needs. The results show that IoTsec demonstrates a satisfaction in modeling IoT security. In conclusion, IoTsec was able to further model security in the IoT environment.

The Table 5 shows a comparison of several modeling methods between the different extensions of UML. According to [3][8][11], we compare the different UML extensions to synthesize the most effective extension. Table 5 summarizes those UML extensions. The lines represent the characteristic of each tool: (1) UML extensions or visual representation; (2) Extension specific for IoT; (3) System security concerns model; and (4) security requirements modeling. The difference between these extensions is seen in Table 5.

Table 5. UML extensions comparison for IoT security.

	UMLsec	SysMLsec	UML4IoT	SysML4IoT	IoTReq	IoTsec
UML extension	✓	✓	✓	✓	✓	✓
Extension specific for IoT	×	×	✓	✓	×	✓
System security concerns model	✓	✓	×	×	×	✓
Security requirements modeling	✓	✓	×	×	×	✓

None of the five extensions (UMLsec, SysMLsec, UML4IoT, SysML4IoT, and IoTReq) is the best because each one has different modeling abilities. However, IoTsec is considered as a great tool because it gives us to model exactly what we need. In this comparison, we mention that IoTsec is more effective than UMLsec, SysMLsec, UML4IoT, SysML4IoT, and IoTReq regarding IoT security modeling.

We also discovered that UMLsec, SysMLsec, UML4IoT, SysML4IoT, and IoTReq do not lead to good results, due to the lack of IoT modeling or security modeling. The IoTsec allows to properly provide developers with a useful nomenclature. Although that IoTsec extension models the security for IoT systems, it is not yet confirmed that this extension would be useful for security of IoT systems.

Due to (1) the increasing heterogeneity of IoT systems, (2) the lack of a standard and representative language for such systems, and (3) the UML limitations [1] for providing graphical diagrams and expressing relationships, the UML is making possible the use of extensions for representing and modeling systems in IoT.

Forest fires will increase more and more due to several factors such as human activities, climate change, etc. According to [20], with 50000 fires and 600000 ha burned on average each year, forest fires in the Mediterranean basin represent a significant part of the planet's fires. According to various sources, the total annual cost of firefighting and security measures in the region exceeds US$1 billion.

It is true that the IoT based forest fire detection system specifications are complex. To protect the system from potential risks, some measures should be taken. According to my opinion, the best way to achieve a secure system is to follow these proposed criteria:

- Forest fire must be detected and inhibited at an early stage to stop, slow down its growth, or even do actions against it. i.e., The start of the fire must be predicted in real time. In other words, real-time and early detection of forest fires are crucial to the success of a defense system. For early fire detection and real-time monitoring, drones fully fitted with sensors and cameras is required to provide an efficient and cost-effective solution [4].
- Therefore, IoT and AI (artificial intelligence) could be recommended for early-warning systems. Moreover, IoT and DL (deep learning) [6] could be used to detect and predict wildfires since the use of DL can help to identify the properties/characteristics of fire. Furthermore, drones and DL can be used to increase the accuracy and speed of forest fire detection. The choice of these technological innovations (IoT, AI and DL) is justified by the previous results that have been done in the literature review.
- Data integrity and confidentiality are crucial parts of security.
- At the physical layer, lightweight cryptography [1] is the proposed solution that should be implemented on sensors.
- At the network layer, IDS (Intrusion Detection System) on the gateway layer is the best idea to detect intrusions/anomalies in the time it occurs.
- To detect intrusions and anomalies, we need to perform a real time monitoring on the network.
- Use communication protection.
- Fire's location with its characteristics should be identified more accurately. For that, drones can be used to identify the location and spread of the fire [4];
- Ensuring the security of connected objects.
- Choose components offering suitable security mechanisms such as robust cryptographic properties.
- Ensure that all links in the chain offer a satisfactory level of security.
- Installation of connected sensors. These sensors must be safe.
- Ensuring that the system is not vulnerable.
- More sophisticated methods and techniques should be followed.
- Preventive actions and corrective control methods must be established against this kind.
- Protection mechanisms must be implemented to minimize damages and losses and to avoid cybercrime's impact.
- Designers must pay very attention and awareness to the security and safety of their products, since the attackers of security systems are attentive, and constantly developing new sophisticated techniques and being updated with new technologies to access information.

Table 6 compare the results in works [1][3] with the new one. The words in bold are the findings of IoT security modeling obtained. The results obtained from the two different works [1][3] are compared to an actual work. The actual work showed an efficiency, a satisfaction and a significant improvement by further modeling the system using a UML extension called IoTsec.

Table 6. Comparison of the IoT security modeling between the different languages and extensions.

Language/ Extension	**Our results**	**Related works**	**Related works results**
UML	*Contribution 1:* gave us that the **physical layer** and the **network layer** are the two layers chosen for modeling [3].	[11]	Authors represented a small system with those models
		[13]	Authors used a UML extension called IoTReq for modeling the IoT domain
UML and SysML	*Contribution 2:* The most efficient language is **SysML** The best diagram is **Requirements diagram** [1]	[9]	Authors used SysML and UML to address the issues of a cyber-physical system components development
IoTsec	*Contribution 3:* Modeling and designing an application domain which is forest fires detection system with **IoTsec**	[8]	Authors represented security concerns with a visual notation

Based on the previous work [1], if we compare the contribution 2 with the actual work, we notice that the requirement diagram does not only exist in SysML language, but it also exists in UML extensions, more precisely IoTsec. Moreover, extensions work better than languages in modeling security or systems. The findings were high compared to (those in paper [1][3]). SysML proves its effectiveness than UML, for example, for the sensor network [21], it is necessary to be able to express the constraints relating to the battery, or even the low/small quantity of available resources.

To design the security of the IoT system we need to model IoT security system requirements, among which some functions [1] can be used like Lightweight Cryptography, Encryption, Privacy, Availability, Confidentiality, Integrity, Key Management, Trustworthiness, Reliability, IoT communication technologies and protocols protections, Authentication, Authorization, Access Control, Intrusion Detection System (IDS).

- Security requirements and solutions in the physical layer, include [3] Availability of sensors and functionality of equipments [1], the integrity and confidentiality of data, Lightweight cryptography, Authentication, Authorization, Key management, trustworthiness [22] etc.
- In the network layer, security requirements and solutions include [3] Compatibility, Non-repudiation, Authentication, Authorization, Integrity, Availability, Key management, Privacy, Authentication, Encryption, etc.
- In the middleware layer, security requirements and solutions include [3] Authenticity, Secure cloud computing, Access control, Integrity, Confidentiality.
- In the application layer, security requirements and solutions include [3] Authentication, Authorization, Privacy, Access control, Integrity.

V. Conclusions

In [1], the study was based on a comparison between UML and SysML for modeling IoT security systems in order to choose the best model and diagram that meet our goal. In the actual work, a continuation and improvement of results were made.

This paper limited the field of the work within an IoT based forest fires detection system to have real example and to properly master the security context. Nowadays, forest fires have become the most extremely dangerous in targeting the ecosystem. IoT become highly essential for forest fires detection systems. Therefore, there is a lack of the modeling of IoT systems.

The best tool or UML extension to follow as a recommendation is IoTsec for IoT security modeling. This paper has also discovered that IoTsec lead to good results in terms of providing a useful nomenclature and modeling security systems in IoT environment. Using this extension, we have proven that IoTsec is the best compared to other methods of IoT modeling.

This paper illustrates the conception of the secure physical layer and the network layer for the forest fires detection system including 2 solutions (Lightweight cryptography at the physical layer and IDS at the gateway layer).

Thanks to SysML, we could model the proposed system presented in “Figure 4” and "Figure 5". The proposed system is easy/simple to understand for anybody familiar with these concepts. This modeling will soon help us to secure our system.

For our future work, we will study the generation of tests based on models or SysML models in order to validate that model. Normally, each modeling that is done whether UML or SysML must be validated. For validation, several methods to do, for example, the method of generating tests related to the model (i.e., generating tests based on the model).

References

[1] Meziane, H., Ouerdi, N., Mazouz, S., & Abraham, A. (2023). A Comparative Study for Modeling IoT Security Systems. *In International Conference on Intelligent Systems Design and Applications*. Springer, Cham.

[2] Meziane, H., Ouerdi, N., Kasmi, M.A., Mazouz, S.: Classifying security attacks in IoT using CTM method. In: Ben Ahmed, M., Mellouli, S., Braganca, L., Anouar Abdelhakim, B., Bernadetta, K.A. (eds.) *Emerging Trends in ICT for Sustainable Development.* ASTI, pp. 307–315. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-53440-0_32.

[3] Meziane, H., Ouerdi, N.: A Study of Modelling IoT Security Systems with Unified Modelling Language

(UML). *Int. J. Adv. Comput. Sci. Appl. (IJACSA)* 13(11) (2022)

[4] Yandouzi, M., Grari, M., Berrahal, M., Idrissi, I., Moussaoui, O., Azizi, M., K. G., & Elmiad, A. K. Investigation of Combining Deep Learning Object Recognition with Drones for Forest Fire Detection and Monitoring. *Int. J. Adv. Comput. Sci. Appl. (IJACSA),* 14(3), 2023

[5] M. Grari, I. Idrissi, M. Boukabous, O. Moussaoui, M. Azizi, and M. Moussaoui. Early wildfire detection using machine learning model deployed in the fog/edge layers of IoT. *Indones. J. Electr. Eng. Comput. Sci*., 27(2), 2022

[6] Yandouzi, M., Grari, M., Idrissi, I., Moussaoui, O., Azizi, M., Ghoumid, K., & Elmiad, A. K. (2022). Review on forest fires detection and prediction using deep learning and drones. *J. Theor. Appl. Inf. Technol.*, 100(12), 2022

[7] J. Jürjens, UMLsec: Extending UML for secure systems development. *Proc. 5th Int. Conf. Unified Model*. Lang., pp. 412– 425, 2002.

[8] Robles-Ramirez, D.A., Escamilla-Ambrosio, P.J., Tryfonas, T.: IoTsec: UML extension for Internet of things systems security modeling. *In: 2017 International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE)*, pp. 151–156. IEEE (2017)

[9] Thramboulidis, F.: Christoulakis, UML4IoT— A UML-based approach to exploit IoT in cyber physical manufacturing systems. *Comput. Ind.* 82, 259–272 (2016)

[10] Costa, B., Pires, P.F., Delicato, F.C., Li, W., Zomaya, A.Y.: Design and Analysis of IoT Applications: A Model-Driven Approach. *In IEEE 14th Intl Conf on Dependable*, pp. 392–399. Autonomic and Secure Computing, Auckland (2016)

[11] Geller, M., & meneses, A.A.: Modeling IoT Systems with UML: A Case Study for Monitoring and Predicting Power Consumption. *American Journal of Engineering and Applied Sciences.* 14(1), pp. 81–93. (2021)

[12] https://uml.developpez.com/cours/Modelisation-SysML. Accessed 2022/04/05

[13] Reggio, G.: A UML-based proposal for IoT system requirements specification. *In: Proceedings of the 10th International Workshop on Modeling in Software Engineering,* pp. 9–16 (2018)

[14] Harrand, N., Fleurey, F., Morin, B., & Husa, K. E. ThingML: a language and code generation framework for heterogeneous targets. *In: Proceedings of the ACM/IEEE 19th international conference on model driven engineering languages and systems*, pp. 125-135 (2016)

[15] Hind, M., Noura, O., Amine, K.M., and Sanae, M. Internet of Things: Classification of attacks using CTM method. *In Proceedings of the 3rd International Conference on Networking, Information Systems & Security*, pp. 1–5 (2020)

[16] Jha, D.N., et al.: IoTSim-Edge: a simulation framework for modeling the behavior of Internet of Things and edge computing environments. *Software: Practice and Experience* 50(6), pp. 844–867 (2020)

[17] G. C. Hillar, MQTT Essentials - A Lightweight IoT Protocol, vol. 53, no. 9. *Packt Publishing Ltd*, (2017).

[18] N. Naik, "Choice of effective messaging protocols for IoT systems: MQTT, CoAP, AMQP and HTTP. IEEE Int. *Symp. Syst. Eng*. (2017)

[19] "Internet of Things Market Size | IoT Market Analysis, Trends, ans Forecast." https://www.verifiedmarketresearch.com/product/global-internet-of-things-iot-market-size-and-forecast-to-2026/. Accessed 2021/08/15

[20] http://www.fao.org/3/x1880f/x1880f07.htm. Accessed 2023/01/15

[21] Belloir, N., Bruel, J.M., Hoang, N., Pham, C.: Utilisation de SysML pour la modélisation des réseaux de capteurs. *In: LMO*, pp. 169–184 (2008)

[22] Akram, H., Konstantas, D., Mahyoub, M.: A Comprehensive IoT Attacks Survey based on a Building-blocked Reference Model. *Int. J. Adv. Comput. Sci. Appl. (IJACSA)* 9(3) (2018)

[23] TF Gharib, H Nassar, M Taha, A Abraham, An efficient algorithm for incremental mining of temporal association rules, Data & Knowledge Engineering 69 (8), 800-815, 2010.

[24] S Dasgupta, S Das, A Biswas, A Abraham, On stability and convergence of the population-dynamics in differential evolution, AI Communications, 22 (1), 1-20, 2009.

[25] F. Xhafa, A. Abraham, Metaheuristics for Scheduling in Industrial and Manufacturing Applications, Studies in Computational Intelligence, Vol 128, 2008.

[26] A. Abraham, N.S. Philip, P. Saratchandran, Modeling chaotic behavior of stock indices using intelligent paradigms, arXiv preprint cs/0405018, 2004.

[27] AK Shukla, M Janmaijaya, A Abraham, PK Muhuri, Engineering Applications of Artificial Intelligence, Engineering applications of artificial intelligence: A bibliometric analysis of 30 years (1988–2018), 97:517-532, 2019

[28] L Dora, S Agrawal, R Panda, A Abraham, Optimal breast cancer classification using Gauss–Newton representation based algorithm, Expert Systems with Applications, 97: 134-145, 2017.

[29] A Rajasekhar, RK Jatoth, A Abraham, Design of intelligent PID/PIλDμ speed controller for chopper fed DC motor drive using opposition based artificial bee colony algorithm, Engineering Applications of Artificial Intelligence, 97: 13-32, 2014.

[30] A Abraham, Intelligent systems: Architectures and perspectives, Recent advances in intelligent paradigms and applications, Springer, 1-35, 2003.