Journal of Universal Computer Science, vol. 25, no. 6 (2019), 571-590
submitted: 7/1/19, accepted: 20/5/19, appeared: 28/5/19

Harmonizing IoT-Architectures with Advanced Security Features – A Survey and Case Study

Łukasz Apiecionek
(Kazimierz Wielki University, Bydgoszcz, Kujawsko-Pomorskie, Poland
lukasz.apiecionek@ukw.edu.pl)

Marcel Großmann
(Otto-Friedrich-Universität, Bamberg, Germany
marcel.grossmann@uni-bamberg.de)

Udo R. Krieger
(Otto-Friedrich-Universität, Bamberg, Germany
udo.krieger@uni-bamberg.de)

Abstract: In recent years we have realized a rapid development regarding the Internet of Things (IoT). Its goal is to interconnect all possible devices to the Internet and to enhance these physical objects by new functionalities. In this way a user's life standard shall be improved. Regarding the application of Internet of Things concepts, there are some commonly known types of an IoT architecture which can provide different technical opportunities. However, comparative studies on Internet of Things architectures are rare. To relieve the difficulties of establishing a single universal IoT architecture, we describe some well-known architectures and compare these proposals with a special regard to important security aspects. A major focus is devoted to methods repulsing Denial-of-Service attacks. We compile a set of criteria that support network administrators in their decision-making processes with regard to a considered specific IoT scenario and its solution. The goal is to fit optimally to the requirements of these solutions. Finally, the proposed approach is illustrated by three already deployed IoT systems and a comparison of their related architectures and functionalities is presented.
Key Words: Internet of Things, Network Security, Critical Infrastructure Protection, DDoS
Category: C.2.1, D.4.6, F.2.1, K.6.5

1 Introduction

At present, computer systems are a common element of everyday life. In recent years a rapid enhancement of their scope has been realized by means of the Internet of Things (IoT). Its goal is to interconnect all possible devices to the Internet and to enhance these physical objects by new functionalities. In this way a user's life standard shall be improved [Sundmaeker et al. 2010]. The objective is to ensure everybody's access to any desired service at any place and by any possible transmission medium. The simplest example of the concept realized by the Internet of Things comprises a fridge with access to the Internet that recognizes by its own accord the amount and the volume of its stored products.

In case of any shortage, it places an order in a shop, makes a payment by the saved credit card data and arranges the delivery of ordered goods. After picking up the supply and placing the new products into the fridge, it identifies them and recalculates their quantity or volume to forecast the follow-up ordering.

Presently, the Internet of Things can be described as a sophisticated technical solution process in progress. New architectures emerge all the time and at the same moment the technology allows to develop new services [Carrez et al. 2013, Wu et al. 2010]. One of the currently available and widely used services concerns monitoring all kinds of physical values, production lines or processes which accompany them. The provided IoT solutions allow a better decision making. For instance, Maciej Kranz [Kranz 2017] states that combining technology with the decision-making process helps to improve a company's financial results. For this reason such IoT solutions should be taken into account that use an efficient implementation. However, introducing such IoT systems in a company may generate numerous problems, such as how to make a requirement analysis, which solution should be chosen, or whom should we entrust the IoT modelling, execution and implementation. Taking into consideration the process of implementation, an ideal solution would consist of a universal architecture, combining all technical aspects, including the operational speed as well as the security of transmission and data processing of a proposed IoT solution.

In this paper we present an attempt to develop a universal Internet of Things architecture with specific protocol structures for a wide range of applications. First, the history of IoT development is covered in section 2. Section 3 summarizes basic types of an IoT architecture. In section 4 safety and security issues concerning data processing are discussed. Section 5 contains the description of some ready-to-use solutions related to security, while section 6 analyses three already existing IoT solutions. The conclusions are presented in section 7.

2 A Proposed Design Methodology for Internet of Things

The Internet of Things enables physical objects or their logical abstractions to share information and to coordinate decisions. In this way it changes traditional objects into so-called smart objects. This transformation is achieved by equipping these objects with sensors, transmission protocols and appropriate software to enable data processing and a communication with other devices. Table 1 illustrates the overall IoT concept [Al-Fuqaha 2015, Fremantle 2015]. Herein every domain specific application is interacting with domain independent services, whereas in each domain, sensors and actuators communicate directly with each other. It is assumed that in a course of time more and more devices will be connected to the Internet and altogether they will create an intelligent environment. Synchronizing the IoT solutions will allow, for example, an earlier opening of the garage door when the car approaches the premises. Using intelligent transport systems will enable more efficient traffic control preventing congestion or ensuring the emergency vehicles right-of-way by manipulating traffic lights. However, this approach requires to overcome numerous obstacles. In this regard some important issues include:

– the necessity of providing *power supply* for all elements of an IoT solution;

– the necessity of *connecting various devices* which may have been incompatible before and to determine how to connect the various elements, and which converters and gateways have to be developed;

– addressing the data and to decide how to *address and identify the devices*;

– the necessity of developing *data transmission protocols* and to determine how to send the data as well as how to transmit them efficiently;

– the necessity of *transmitting the data* to remote destinations within the area covered by an IoT solution;

– creating a *data center* or, at least, a virtualized data maintenance system to collect and process the data and to decide where and how to store big amounts of data or how to share them;

– developing the *algorithms for data analysis* and to specify how to analyze the data and how to draw adequate conclusions.

Regarding a smart city, for instance, an intelligent transport system is one of the most interesting ideas [Ambak et al. 2009]. Combined with intelligent and in the near future autonomic cars, it enables not only to ensure green traffic lights for emergency vehicles, but also to prevent congestion in a more intelligent way by manipulating the duration of the green light period for various traffic directions. Intelligent IoT solutions may further open a gate when the car approaches it. Due to the listed issues, it is difficult to select one universal solution for an Internet of Things architecture. Various authors already presented diverse approaches to this design issue including a five-layer model used in [Al-Fuqaha 2015, Khan et al. 2012, Yang et al. 2011]. A related three-layer model operates mostly on the application layer [Lin et al. 2017, Ling 2013, Mukherjee et al. 2017, Ngu et al. 2017, Shang et al. 2012, Zhao et al. 2013] while a five-layer model divides the application layer into three sublevels: *the business layer, the application layer, and the service management.* The *service-oriented IoT architecture* (SOA) is an approach which focusses on creating an IoT architecture based on the use of system services [Deugd et al. 2006, Yuan et al. 2007]. Its *objects layer* represents physical sensors which collect data. Their kind depends on the desired purpose. To make the solutions more universal, this layer

Table 1: Model of an IoT reference architecture (cf. [Al-Fuqaha 2015], [Fremantle 2015]).

Applications		Dashboard	Web/Portal	API Management	
Basic Services	Event Processing	Data Analytics	Visualization	Storage	
Middleware & Coordination Layer	Message Broker		HTTP 2.0	Aggregation	Bus Layer
	MQTT	CoAP			Enterprise Service Bus (ESB)
Communication Layer	TCP	UDP	WebRTC QUIC	IETF DetNet	
	IPv6	IPv4	IPSec	IEEE 1905.1	IEEE 1888.3
Access Layer	6LoWPAN	4G/ LTE-A	5G	4G/5G-WLAN	
	IEEE 802.15.4 BLE NFC	NB-IoT eMTC	5G-ULL Network IEEE 802.1 TSN 5G-mmWave	IEEE802.11ac IEEE802.11ad IEEE802.11ah	
	Identification		Sensing		
	Naming	Addressing			
Edge Technologies, Edge Devices, and Environments	Embedded Systems	SBCs	Smart Environments		
	Sensors Actuators Wearables RFID Tags	Smartphone Raspberry Pi Arduino	Smart City Smart Vehicles Smart Cars	Smart Buildings Smart Homes Industrial IoT e-Healthcare	Smart Grid Smart Meter

should consist of plug-and-play mechanisms which would not only work in the layer of sensor's physical plugging in. Its connection should generate an adequate information for upper layers to enable a decision making based on the acquired data. An *object abstraction layer* is responsible for secure communication between the objects and an upper layer. It is noteworthy that the communication may proceed via various media. The *service management layer* is in charge of addressing the solutions. It allows to control physical objects through the application layer using *names and addresses*. The *application layer* provides the solutions for the end users according to their needs. Finally, the *business layer* is responsible for managing the entire IoT solution.

The *classification of IoT solutions* can be performed based on various criteria, for example, according to the technology used for the communication [Ray 2016]. An IoT architecture is also described by its elements [Al-Fuqaha 2015]. Relevant security aspects can be identified for all elements of an IoT system and need to be considered level-wise during the system development phase. Presently, there are numerous projects aiming to develop new IoT solutions and to make an attempt on their standardization. Many of these solutions are open source, which makes their development easier and enhances their range of applicability. An alternative classification method of the IoT components proposed in [Sebastian and Ray 2015] lists the following six basic elements:

– *Device* provides sensing, actuation, control, and monitoring activities;

– *Communication* performs the communication between the devices and the remote servers;

- *Services* are employed for device modeling, device control, data publishing, data analysis, and device discovery;
- *Management* provides various functions to govern the IoT system and to interwork with an underlying IoT management system;
- *Security* provides functions such as authentication, authorization, privacy, message integrity, content integrity, and data security;
- *Application* is an interface which provides necessary modules to control and monitor various aspects of the IoT system.

3 Harmonizing IoT Architectures From a Computational Perspective

To pursue the development of a universal structure of an IoT architecture, the sketched existing models should not be disregarded. In the course of analysis of the present-day offers, they may be divided with regard to the place of data processing, namely, in the sensor itself or in a central point. Thus, the following hierarchical architecture with three layers can be specified:

- a *sensor-actuator layer* with sensors, actuators, and smart IoT devices,
- a *fog computing layer* hosting virtualized fog computing in fog cells,
- a *cloud computing layer* hosting cloud services in data centers.

These perspectives on IoT solutions are illustrated in Figure 1 [Al-Fuqaha 2015]. At the lowest level there are *sensors*, i.e., those IoT devices which are responsible for collecting data. They are and will be the most numerous ones as they are accountable for connecting an IoT solution to the network. Due to their great number, they generate the highest requirements regarding the address pools and the network traffic when transmitting the acquired data. The middle layer is a *fog computing layer*, which gathers, aggregates and preliminarily processes the collected data. Such an approach fosters a reduction of the network traffic. It is estimated that millions of IoT sensors will generate a lot of unnecessary traffic and the fog computing layer helps to prevent such situations. The computational and networking solutions at this level are more complicated than at the sensor level. They possess a higher computing power and in most cases they require different working conditions. The uppermost layer is the *cloud*. Here all data are processed in an IaaS, PaaS or SaaS cloud environment. It requires a suitable structure to build a centralized or distributed center for data processing and to manage it in a proper manner, but also to send all these data to the center where their smart and efficient processing takes place.

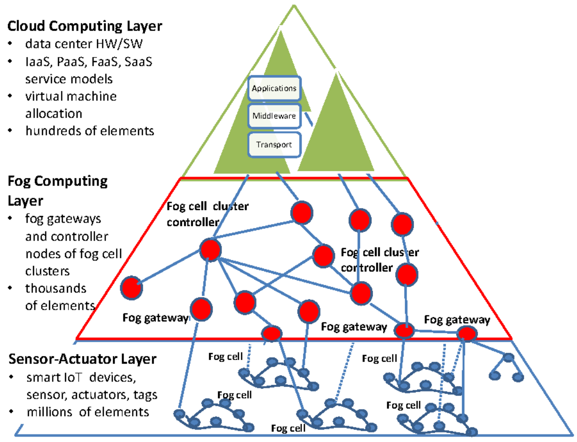


Figure 1: Computing model of a hierarchical IoT reference architecture.

All layers may exist separately or in connection with others, providing services for the other ones. The *sensor layer* covers not only the *sensors* themselves, which measure physical values, but also the *actuators* that are responsible for controlling, changing and setting of physical values. This layer requires to develop adequate converters, which may be active, passive, mechanical, optical, magnetic, thermal, electrical, biological, chemical, etc. The *fog computing layer* may also operate as a bridge between the sensor-actuator and cloud computing layers, aggregating the collected data and reducing the network traffic. It allows to move the data processing services closer to the terminal elements, i.e. the sensors. The fog computing layer also enables earlier reaction to the data obtained from the sensors. It is recommended to the designers due to the following properties [Al-Fuqaha 2015]:

– *Location:* Fog resources are positioned between smart objects and the cloud data centers and in this way they provide better delay performance.

– *Distribution:* As fog computing is based on micro-centers with limited storage, processing and communication capabilities in comparison to the cloud, it is possible to deploy many such micro-cloud centers closer to the end users, as their cost is usually a small fraction compared to current cloud data centers.

Basic features of this 3-layer computational structure include [Al-Fuqaha 2015]:

– *Scalability:* Fog computing allows the IoT systems to be more scalable, i.e. when the number of end users increases, the number of deployed micro-cloud centers in the virtualized fog layer can follow to cope with the growing load. Such increase cannot be achieved by the cloud computing layer itself because the deployment of new data centers is cost-prohibitive as it entails the need to provide special server rooms, network administrators, monitoring systems, etc..

- *Security:* The system constitutes a distributed solution. Thus, its security issues are complex and time-consuming and often require to implement new approaches that are completely different from the commonly applied ones.
- *Density of devices:* Fog computing helps to provide resilient and replicated services.
- *Mobility support:* Virtualized fog computing resources act as a mobile cloud, as this layer is located close to the end users.
- *Real-time aspects:* Fog computing has the potential to provide better performance for real-time interactive services.
- *Standardization:* Fog computing resources can interoperate with various cloud providers.
- *On-the-fly analysis:* Fog computing resources can perform data aggregation to send partially processed data, instead of raw data, to the cloud data centers for further processing.

Fog computing solutions can be deployed hierarchically, resulting in multi-tier solutions [OpenFog 2017]. Regarding the fog computing layer numerous ready-to-use open source solutions have been developed so far and they can be easily adapted to the individual needs. The LinuX Container is one of these examples. A widely used set of IoT solutions is also provided by a distributed virtualized computing environment derived from Docker container technology [Großmann et al. 2016]. It is widely known that cloud computing offers a new management mechanism for big data, which enables the data processing and the extraction of valuable information from them. Employing cloud computing in the IoT area involves the following challenges [Al-Fuqaha 2015]:

- *Synchronization:* Synchronization between different cloud providers poses a challenge to offer real-time services since they are built on top of various cloud platforms.
- *Standardization:* Standardizing the cloud computing is also a significant issue for cloud-based services in the IoT due to a necessity to interoperate with various providers.
- *Balancing:* Achieving a balance between general cloud service environments and the IoT requirements may raise difficulties due to the differences in the infrastructure.

– *Reliability and security:* The security of the IoT cloud-based services presents another challenge due to the differences in the security mechanisms between the IoT devices and the cloud platforms.

– *Management:* Managing the cloud computing and the IoT systems is also a demanding task as they use different resources and components.

– *Enhancement:* Validating the IoT cloud-based services is necessary to ensure high-quality services that meet the customers' expectations.

– *Network transfer:* Collecting all data in cloud computing generates an intensive data traffic throughout the network equipment.

An important layer, distinguished lately in the models of IoT architectures, is given by the *middleware layer* [Katasonov et al. 2008]. Its role is to provide the access to an IoT solution for the users, both the system administrator and the end users. This is the reason why this layer is also referred to as the *management layer*. Depending on the solution at hand, this layer allows to access the services provided by the IoT by means of mobile devices such as laptops, tablet computers, smartphones, as well as stationary devices. At this layer it is a crucial issue to provide a secure access as well as proper algorithms, i.e., preferably light ones, without unnecessary data load. For this purpose existing protocols can be applied, such as the Constrained Application Protocol (CoAP), Message Queue Telemetry Transport (MQTT), and Message Queue Telemetry Transport for Sensor Networks (MQTT-SN) [Stanford-Clark and Truong 2008]. The latter have been developed and optimized for machine-to-machine communication.

The presented IoT architectures allow to implement numerous systems with their services and applications. Some of those are discussed in the subsequent sections. The literature reports on many other solutions, which present models that are suitable for the specified use of an IoT system. For instance, Sivabalan et al. [Sivabalan et al. 2013] propose a model of an architecture to enhance the interoperability between various devices and its application in a multi-vendor scenario incorporating a distributed cloud infrastructure. Based on the sketched survey of layered IoT architectures, it can be concluded that there is no single, general model of an IoT architecture which is suitable for all purposes. Selecting the right architecture for the IoT solution under development depends on numerous factors. To choose the most suitable one, the analysis of the following issues should be considered and related decision must be taken:

– what *means of communication* are to be applied in the solution, what is the *transmission capability* of the designed solution, is there any already existing network architecture or should it be set up from a scratch;

– what is the *bit error rate* in the applied network;

- what is the *amount of the data* to be transmitted by the IoT solution;
- what is the *number of the data receivers*;
- *how many sensors* are to be applied,
- what is the *desired memory capability* of the applied sensors,
- what *power supply* is planned for the solution,
- what is the *longest admissible failure time* of the elements in the IoT solution,
- what is the *required security level*, how will the data be accessed, how will the user access be controlled, what level of data encryption is required.

Choosing the most advantageous IoT solution must be preceded by gathering the adequate answers to all these raised questions. Obviously, there are also other influential factors such as the budget of the planned solution. They may be conclusive to what extent a proposed IoT solution will adequately cover all three layers, i.e. the sensor-actuator, fog computing and cloud computing layer.

4 General Security Issues

The analysis of safety and security issues in the IoT architecture reveals the following main challenges in this field [Al-Fuqaha 2015], [Tankard 15]:

Unique identification	Data encryption	Privacy
Reliability	Availability	Serviceability
Denial-of-Service attack possibility		

Collecting the data from the sensors requires protecting them from their unauthorized use. IoT solutions offer numerous opportunities in the range of monitoring physical parameters as well as making faster and more appropriate decisions. For instance, collecting sensitive information, such as the patients' health data by means of remote monitoring, is restricted by law. To ensure their confidentiality they must be encrypted, which in turn requires properly selected algorithms, their examination and the control of their vulnerabilities. Furthermore, these algorithms demand a suitable computing power from the sensors that are responsible for the data collection. Another crucial aspect is availability. On choosing an IoT solution, for example to optimize the production process in a company, the access to the data which serve as basis of the decisions must be provided. A lack of access to this information or an unavailability of the device may be caused by a Denial-of-Service attack, which is able to block the access to the sensors. In such a situation the IoT system may stop, e.g., a production line, which may result in financial loss. This is the reason why the security issues in any IoT solution are a crucial factor, especially in the case when the solution operates in a public network. When working in a network, IoT based services face the same problems as any other service operating in a public network. Serviceability refers to the issues of automatically installing, updating and connecting new elements to the system, including them into the security system as well as automatic and autonomic failure detection. Due to the specificity of the system, often none of the elements can be omitted. For instance, an attack on an IoT system responsible for the coordination of traffic lights may result in a complete cutting off the green lights. Such situation may in turn lead to immense traffic congestion and in consequence to social unrest. Moreover, an important factor concerns the correct addressing and identification of the IoT elements. It ensures that the source of information is authorized to share them and that nobody is impersonating it. In [Roman et al. 2013] the following attacker models and threats are defined:

Physical damage	Node capture	Controlling
Eavesdropping	Denial-of-Service (DoS)	

Stojmenovic et al. [Stojmenovic and Wen 2014] describe the specific problems in the fog computing architecture. They define the main security issues, such as *authentication* at different levels of gateways as well as (in case of smart grids) at the smart meters installed in the consumer's house. This threat occurs because each smart meter and smart appliance has an IP address, so a malicious user can either tamper with its own smart meter, report false readings, or spoof IP addresses. There are some solutions for such authentication problem. One of them is provided by *Public-Key Infrastructure* (PKI) based solutions, which involve multicast authentication. Some authentication techniques using Diffie-Hellman key exchange can be applied as well. In the fog computing architecture *intrusion detection techniques* can also be applied. Intrusion in smart grids can be detected using either a signature-based method, in which the patterns of behavior are observed and checked against an already existing database of possible misbehavior. Intrusion can also be captured using an anomaly-based method. Borgohain et. al. [Borgohain et al. 2015] also present a possible attack on the IoT infrastructure. The types of attack can be divided according to the layers to which they belong [Borgohain et al. 2015, Farooq et al. 2015, Granjal et al. 2015, Jing et al. 2014]. For example, there are different types of an attack on wireless sensor networks, which can be categorized as follows:

<table>
<tr><td rowspan="2">Attacks on</td><td>Authentication and secrecy</td></tr>
<tr><td>Network availability</td></tr>
<tr><td colspan="2">Silent attacks on service integrity</td></tr>
</table>

Denial-of-Service (DoS) attacks can be divided into the following categories:

– *DoS attacks on the link layer*, such as collision, unfairness, battery exhaustion,

- *DoS attacks on the network layer*, such as spoofing, replaying and misdirection of traffic, hello flooding attack, homing, selective forwarding, Sybil attack, wormhole, acknowledgement flooding,
- *DoS attacks on the transport layer*, such as flooding, de-synchronization,
- *DoS attacks on the application layer*, such as a path-based DoS attack initiated by stimulating the sensor nodes to create a huge amount of traffic in the route towards the base station.

There are solutions to overcome these problems concerning the security aspects within the whole development process of an IoT service. To achieve security throughout the device lifecycle, from the initial design to the operational environment, Tankard [Tankard 2015] lists five essential requirements:

Secure booting	Updates and patches
Device authentication	Access control
Firewalling and intrusion prevention systems (IPS)	

Bekara [Bekara 2014] describes the problems of implementing security in the IoT setting. Due to dealing with security algorithms, protocols and policies for the IoT, several challenges need to be taken into consideration. For instance, Balte [Balte et al. 2015] provides some information about ongoing European projects on IoT security. The summary of the security solutions according to these projects provides the conclusion that, unfortunately, there are some projects in which security is not concerned at all. There is also research work on the *taxonomy of attacks* on the IoT infrastructure and its services [Hossain et al. 2015], which can be divided in the following way:

Attacks based on	Device properties	Information damage level	Attack strategy
	Access level	Adversary location	
	Hosts	Protocol features	Communication protocol stack

In the following we will focus on some selected, highly relevant security aspects and their circumvention.

5 Selected Security Aspects

During the process of IoT system development numerous security aspects have to be taken into account. This requirement applies also to stationary IT systems connected to public networks. These aforementioned security aspects include:

- methods to *secure the access*,
- *access control* methods,

– *data encryption* methods,

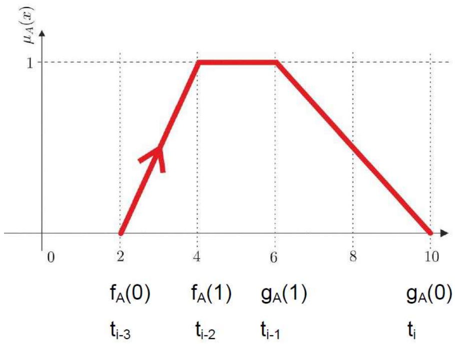
– methods and mechanisms to *monitor the correct network operation*,

– methods and mechanisms to *detect anomalies*.

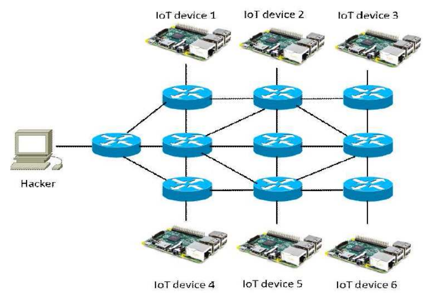
There are a number of methods and mechanisms to solve these issues. However, depending on the type of the IoT system, not all of them can be implemented in certain cases. Low computing power of the terminal devices in the sensor layer implies the need to create new security mechanisms. In the following subsections the applicability of a PKI infrastructure and a new method to detect DDoS attacks are presented as major solution techniques.

5.1 On a DDoS Detection Service of an IoT Architecture

Presently, the most common security mechanism is provided by the *Public-Key Infrastructure* (PKI). Regardless whether it is a bank using the *Transport Layer Security (TLS)* to secure the sessions or some complex network system using the *IPSec* protocol, the security is warranted by *X.509 certificates*. Thus, it is a natural choice for solutions in an IoT setting to use the best security model that is available on the market. Depending on the design objectives, one may incorporate *IPsec and TLS* with an implementation of PKI for authentication purposes. The research on the most popular solutions available on the market raises some severe questions whether their computing power is sufficient or not to run the encryption algorithms used by the sketched mechanisms. The devices at the lowest layer, i.e. the sensors, have, of course, the lowest computing power. One of the most common devices is provided by the Raspberry Pi SBC. Depending on the type of encryption, it is able to achieve a throughput of 21 up to 46 Mb/s on a 100 Mb/s Fast Ethernet interface, which should be sufficient for the majority of implementations [strongSwan 2018]. Choosing a hardware solution like Raspberry Pi, it is advisable to select an operating system which is already equipped with some security mechanisms, like Contiki OS. Apart from these elements that are associated with IoT devices to ensure the confidentiality of information, the questions of preventing problems such as the aforementioned Denial-of-Service attacks are raised, too. For this purpose it is recommended to use solutions like IDS [Raza et al. 2013]. But this solution cannot be implemented in all places where it should be. However, while assigning the task of data encryption to the sensors, it is advisable to seek new solutions that are helpful to prevent DoS attacks. One option is to apply lightweight algorithms, using, for instance, fuzzy logic. In this respect Apiecionek [Apiecionek 2017] presents some simple method to detect Denial-of-Service attacks on a constrained IoT device which is also mentioned by other authors [Jing et al. 2014]. The network administrator may notice an increasing number of connections to a device, but he is



(a) Membership function $\mu_A(x)$ of a positively ordered fuzzy number in OFN style.



(b) Network test bed.

Figure 2: The DDoS control setting and the used network test bed.

unable to identify and analyse it beyond the issue of a moment of attack. The proposed algorithm measures the number of connections for subsequent periods of time, i.e. $t_i, t_{i-1}, t_{i-2}, t_{i-3}$, where t_i marks the end of the current time slot. All four measurements together give a fuzzy number in the ordered fuzzy number (OFN) notation. A positively ordered fuzzy number is presented in Figure 2 (a), where $f_A(0) \leq f_A(1) \leq g_A(1) \leq g_A(0)$, and $f_A(0)$ responds to t_{i-3}, $f_A(1)$ responds to t_{i-2}, $g_A(1)$ responds to t_{i-1}, and $g_A(0)$ responds to t_i.

A Fuzzy observation of an IOTd device at time t_i is a set [Czerniak et al. 2016]

$$\mathrm{IOTd} \equiv \mathrm{IOTd}[t_i] = \{f_A(0)[t_{i-3}], f_A(1)[t_{i-2}], g_A(1)[t_{i-1}], g_A(0)[t_i]\} \qquad (1)$$

where $t_i > t_{i-1} > t_{i-2} > t_{i-3}$, $\mid t_i - t_{i-1} \mid = \mid t_{i-1} - t_{i-2} \mid = \mid t_{i-2} - t_{i-3} \mid = \Delta t$ determine the time slots of the measurement. We set

$$\mathrm{IOTd}_{\ positive} = true \quad \text{if} \begin{cases} f_A(0) < f_A(1) < g_A(1) & or \\ f_A(1) < g_A(1) < g_A(0) \end{cases} \qquad (2)$$

holds, otherwise $\mathrm{IOTd}_{\ negative} = true$. According to this definition we obtain an OFN with positive order when the packet count increases, an OFN with negative order when the packet count decreases. The analysis of the packet statistics along with the appropriate counters give a fuzzy number and allows us to define a fuzzy observation of a group $\{\mathrm{IOTd}_1, \ldots, \mathrm{IOTd}_n\}$ of IoT devices. A Fuzzy observation of such a group is described by the following formula

$$\mathrm{IOT}_m = \sum_{i=1}^{n} \chi(d_i), \qquad \chi(d_i) = \begin{cases} \mathrm{IOTd}_i \cdot w_i \text{ , if } \mathrm{IOTd}_{i\ positive} = true \\ -\mathrm{IOTd}_i \cdot w_i \text{ , if } \mathrm{IOTd}_{i\ negative} = true \end{cases} \qquad (3)$$

where $w_i \in \{w_1, \ldots, w_n\}$ describes the impact of the IoT device IOTd_i on the entire solution. Then our DoS decision rule reads as follows: *An attack on the IoT infrastructure is recognized when IOT_m is positive.*

This method to detect DoS attacks targeting the IoT infrastructure has been validated subject to laboratory conditions specified in the following subsection.

5.2 Implementation of the DDoS Detection Method

To validate the proposed detection method, some laboratory tests were performed [Apiecionek 2017]. The network depicted in Figure 2 (b) has been used during these experiments and comprises 6 devices. They serve as the IoT platform and are subject to an attack executed by means of the DDoS simulator DDOSIM - Layer 7 [DDOSIM 2018]. In this test bed TCP connections to the IoT devices have been established in the following way. The DDOSIM software sends a TCP SYN packet on port 80. The IoT device answers with a TCP SYN/ACK packet and reserves the resources. The software sends a TCP ACK packet, and the DDOSIM software sends a HTTP/GET packet. The whole DDoS attack process has been covered by the following eight steps:

1. The hacker machine is working on IP address 192.168.10.12.
2. 6 IoT devices are working on the IP addresses 192.168.x.4, $x \in \{1, 2, 3, 4, 5, 6\}$.
3. Packets are sniffed using Wireshark at the IoT devices 1 to 6.
4. The hacker host starts the DoS attack on the first IoT device.
5. After one minute the hacker starts the attack on another IoT device.
6. The hacker machine is sending 1000 HTTP GET messages to the IoT device every 30 seconds.
7. When all IoT devices are under attack, the hacker continues his attack for 5 minutes.
8. When the attack has ceased, the packets are sniffed for another 5 minutes.

The devices were connected to Cisco routers using the OSPF routing protocol. No Quality-of-Service methods were implemented in the network. According to the proposed method, the IoT devices collected the statistics of the connections in a given time slot of 1 minute duration. In compliance with our specification (3), the OFN metrics can be determined for each IoT device in each time slot. Then, IOT_m can be calculated. Assuming that all devices are of equal importance to the IoT system in the situation considered in the experiment, the w_i parameter has been set to 1. Then the method could successfully identify the attack instant.

In this way, the presented lightweight detection method can be implemented in IoT solutions and quickly provide information about a possible DoS attack.

6 A Case Study of Deployed IoT Solutions

In this section three already existing technical systems which may be classified as IoT solutions are analysed according to the harmonized IoT model presented in sections 2 and 3. The presentation covers the related architecture, the development approach, and the requirements of the underlying design.

6.1 Monitoring – A Fire Brigade Monitoring Tool

The Monitoring system, described in [Apiecionek and Krieger 2019], has realized an IoT solution to supervise the equipment of fire brigades. It has been developed for fire brigades in the Poznań region in Poland. In the sensor and actuator layer the following devices are connected: pumps, electric shears and a water tender. The scope of monitoring covers the working parameters of these units. The security issues of this IoT system include data encryption, unauthorized use and DoS-type attacks on the system. It is still in the testing phase and the remaining problems are going to be solved now.

The system allows to monitor the used technical equipment. For instance, the process of pumping out water from a flooded basement takes hours. So far, a firefighter has to supervise the whole action. But now it is possible to control the pump by the monitoring component and in case of any problems it will trigger an alarm. The architecture of the system was affected by power supply issues. All devices possess an external power supply unit (for example a generator), but their voltage level is unstable. Thus, it was necessary to construct a suitable battery supply system, charged by the power supply, which solved the problem.

6.2 Battlefield Management System JASMINE

The already existing Battlefield Management System JASMINE, produced by the Polish company TELDAT, can serve as an example of an IoT solution for military command vehicles. A wide range of tests has been performed on this solution, for which the first author served as a co-originator. They allow to draw conclusions and to present this specific solution in the context of an IoT system. Its aim is to provide a support system for the command units, which is able to automate the command and control process and to assist the military operations by these means. For this purpose both the equipment and corresponding software solutions were developed. The equipment solutions in the sensor layer of the IoT model allow to connect a number of sensors used in command vehicles, i.e. sensors of chemical or biological contamination, radiation or laser radiation detectors, 360-degree cameras, night vision devices, an inertial navigation system, GPS, etc. Aside from these items, devices belonging to the actuator type were also connected, i.e. extraction and filter systems, emergency signaling for the crew, lightning control in the vehicle. The sensors allow the system to detect emergencies, such as entering a contaminated area, to send the information automatically to the control system and to forward them by radio to other vehicles operating in this mobile system, as well as to the headquarters. The information is automatically displayed on the maps. In the fog computing layer there are WAN access box devices, which are responsible for the integration of the sensors and storing of data on local databases containing the information on events, plans, orders, the location of own troops, allies and enemies. These data are stored in a database operating in accordance with the J3CIEDM standard. The data from the fog computing layer are sent to the command post, i.e. the headquarters, which can be regarded as the cloud computing layer. The latter is analysing the information from thousands of command vehicles operating in the system. The middleware layer contains client applications, adapted to the user's needs, which allow to display maps, situations in the battlefield, as well as the state of sensors and effectors. These applications are dedicated to rugged tablets, resistant to harsh environmental conditions and suitable for the use outside of the vehicles. The security issues of the system include data encryption, unauthorized use and DoS-type attacks against the system. Due to its military use, the system solves the problem by applying suitable policies and mechanisms provided by military standards. The management system provides automation of commanding in a modern battlefield. For instance, the transmission of orders along with the maps may proceed in a smooth, intuitive and fast manner via the available means of communication. Passing the information to one vehicle allows to spread them according to the set hierarchy of operations. The integration devices are equipped with an original routing and radio data transmission protocol, the Battlefield Replication Mechanism [Palka et al. 2016].

6.3 MonTreAL

Storing antique printed matters as well as new books requires suitable conditions to prevent the loss of their quality in the course of time. The conditions for preserving the books printed on modern kinds of paper differ from those ones that are required to keep photographic films. The range of the temperatures in a room is strongly affected by the presence of ventilation ducts, windows and heaters, in the latter case even abruptly. The humidity of air is also a changeable parameter, even by 5%. Thus, it is difficult to indicate which part of the room offers the best conditions for storing the valuable collections. This is the reason why the IoT system MonTreAL (Monitoring Treasures of all libraries) has been developed [Großmann et al. 2017]. It is responsible for monitoring the temperature and air humidity in libraries. The sensor layer of the system covers the sensors with battery power supply, operating even for 6 months, and enables a wireless data transmission. The implemented configuration allows to connect

Table 2: The elements of the analysed IoT systems.

IoT Elements		Monitoring system	BMS JASMINE	MonTreAL
Identification	Naming	DNS	DNS	DNS
	Addressing	IPv4	IPv4, IPv6	IPv4, IPv6
Sensing		Smart sensors, GPS	Smart sensors, chemical sensors, radiation sensors, GPS, laser detectors	Temperature and humidity sensors
Communication		WiFi, 3G/4G/LTE	WiFi, HF, VHF, Wideband, Satellite, wire	Wireless, Ethernet, WiFi
Computation	Hardware	Proprietary project with ARM processor	TELDAT manufactured projects	Raspberry Pi or standard hardware
	Software	Linux	Windows on client, Teldat proprietary OS	Hypriot/Debian based Linux with Docker
Service		Data analysis, failure prediction, equipment position monitoring	Battlefield Management System, position on map visualization, MIP database	Temperature and humidity monitoring, alerting service
Semantic		-	Teldat proprietary for MIP version translation	-

eight or even more sensors to a single computer with its 26 data pins. The latter collects the data from these sensors and forwards them to a server. The solution includes a server, but it can be hardly classified as a cloud-type solution. The security issues of the system include, of course, data encryption, authorization of the users by means of login and password, as well as the DoS-type attacks on the system. Regarding solutions for data transmission security, it is still in the phase of implementation and testing. In conclusion, the IoT system MonTreAL solves the problem of monitoring the temperature and humidity in libraries. In the past, no measurements were performed at night and data were collected in a form of cards, which made it difficult to analyze historical data. Now MonTreAL allows the users to have a full view of those historical data sets.

6.4 Summary

The analysis of the three deployed IoT solutions is summarized in Table 2, see also [Al-Fuqaha 2015]. It lists the elements, protocols and standards that have been implemented in the presented three IoT systems.

7 Conclusions

Advanced IoT solutions are not only a matter of the future, but also of the present-day life. Regarding IoT architectures numerous new technologies and capacities as well as related virtualization techniques are developed nowadays. Applying already existing technologies and open source solutions allows us to achieve quickly new capabilities and to launch new services for the users. It is the services which are essential in these IoT solutions. IoT systems are meant to provide new services which help us to make the life easier, to accelerate some processes, and to increase the efficiency of production while reducing its cost at the same time. The analysis of already existing solutions and trends reveals that there is no single universal IoT architecture. Considering the preliminary analysis of an IoT architecture that is the optimal one to solve a given problem, we presented in this paper some criteria which should be taken into account. This view is particularly determined by the fact that IoT is dedicated to solve specific problems and to provide specific services. In particular, four factors affecting the model of the IoT architecture were discussed. Apart from networking, the storage capability and computing capacity, they also include the energy consumption. It concerns the fact of possessing or lacking power supply in the sensor layer which is conclusive about many features of a system under construction, i.e. which sensors to select, how to design them or which means of communication can be used. The analysis of the security issues revealed that regarding a network solution IoT systems are susceptible to the same threats as the already used network systems. An IoT solution can be vulnerable to attacks of DoS type. That is the reason why it is necessary to develop new, lightweight mechanisms to prevent them and to eliminate their effect on the network level, not only by means of traditional IDS solutions. We believe that it is essential to develop lightweight mechanisms that are able to work on the sensors, as well as to analyze the traffic and to inform the administrator about a problem. These features allow the network layer to launch actions with the aim to block the malicious traffic. Such an efficient method has been proposed already [Apiecionek 2017].

Acknowledgements

The first author acknowledges funding under the grant Miniatura 1 number 2017/01/X/ST6/00613 from National Science Centre, Poland. The authors also declare that there is no conflict of interest regarding the publication of this paper.

References

[Al-Fuqaha 2015] Al-Fuqaha, A., et al.: “Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications”; IEEE Communication Surveys & Tutorials, 17, 4, (2015) 2347–2376.

[Ambak et al. 2009] Ambak, K., et al.: “Intelligent Transport System for Motorcycle Safety and Issues”; European Journal of Scientific Research, 28, 4 (2009), 600–611.

[Apiecionek 2017] Apiecionek, Ł.: “Fuzzy Observation of DDoS Attack”; Prokopowicz, P., et al., eds., Theory and Applications of Ordered Fuzzy Numbers, Studies in Fuzziness and Soft Computing, 356, Springer, Cham (2017), 241-254.

[Apiecionek and Krieger 2019] Apiecionek, Ł., Krieger, U.: "Monitoring - The IoT Monitoring Tool for Fire Brigades"; Image Processing and Communications Challenges 10, IP&C 2018, Advances in Intelligent Systems and Computing, vol. 892, Springer, Cham (2019), 185–191.

[Balte et al. 2015] Balte, A., Kashid, A., Patil, B.: "Security Issues in Internet of Things (IoT): A Survey"; International Journal of Advanced Research in Computer Science and Software Engineering, 5, 4 (2015), 450–455.

[Bekara 2014] Bekara, C.: "Security Issues and Challenges for the IoT-based Smart Grid"; Procedia Computer Science, 34 (2014), 532–537.

[Borgohain et al. 2015] Borgohain, T., Kumar, U., Sanyal, S.: "Survey of security and privacy issues of internet of things"; arXiv preprint arXiv:1501.02211 (2015).

[Carrez et al. 2013] Carrez, F., et al.: "Internet of Things – Architecture, IoT-A"; Deliverable D1.5 – Final architectural reference model for the IoT v3.0, EU-Project IoT-A (257521), (Jul 2013) https://iotforum.org/wp-content/uploads/2014/09/D1.5-20130715-VERYFINAL.pdf

[Czerniak et al. 2016] Czerniak, J. M., et al.: "Practical Application of OFN Arithmetics in a Crisis Control Center Monitoring"; Fidanova, S., ed., Recent Advances in Computational Optimization, Studies in Computational Intelligence 655, Springer, Cham (2016), 51–64.

[Deugd et al. 2006] Deugd, D. S., et al.: "SODA: Service Oriented Device Architecture"; IEEE Pervasive Comput., 5, 3 (2006), 94–96.

[DDOSIM 2018] sourceforge.net: "DDOSIM - Layer 7 DDoS Simulator"; accessed on 2018.06.11, https://sourceforge.net/p/ddosim/wiki/Home/

[Farooq et al. 2015] Farooq, M. U., et al.: "A Critical Analysis on the Security Concerns of Internet of Things (IoT)"; International Journal of Computer Applications (0975 8887), 111, 7 (2015).

[Fremantle 2015] Fremantle, P.: "A Reference Architecture For The Internet of Things"; White Paper, (Oct 2015) https://wso2.com/whitepapers/a-reference-architecture-for-the-internet-of-things/#41

[Granjal et al. 2015] Granjal, J., Monteiro, E., Silva, J. S.: "Security for the Internet of Things: A Survey of Existing Protocols and Open Research Issues"; IEEE Communications Surveys & Tutorials, 17, 3 (2015), 1294–1312.

[Großmann et al. 2016] Großmann, M., Eiermann, A.: "Automated Establishment of a Secured Network for Providing a Distributed Container Cluster"; Proc. 28th International Teletraffic Congress (ITC28), Würzburg, Germany, (Sept. 2016).

[Großmann et al. 2017] Großmann, M., Illig, S., Matějka, C.: "Environmental Monitoring of Libraries with MonTreAL"; Kamps, J., et al., eds., Research and Advanced Technology for Digital Libraries, TPDL 2017, Lect. Notes Comp. Sci. 10450, Springer, Cham (2017), 599–602.

[Hossain et al. 2015] Hossain, M. M., Fotouhi, M., Hasan, R.: "Towards an Analysis of Security Issues, Challenges, and Open Problems in the Internet of Things"; 2015 IEEE World Congress on Services, New York, NY (2015), 21–28.

[Jing et al. 2014] Jing, Q., et al.: "Security of the Internet of Things: perspectives and challenges"; Wireless Networks, 20, 8 (2014), 2481–2501.

[Katasonov et al. 2008] Katasonov, A., et al.: "Smart Semantic Middleware for the Internet of Things"; Proceedings of Fifth International Conference Conference on Informatics in Control, Automation and Robotics, Intelligent Control Systems and Optimization (ICINCO 2008), Funchal, Madeira, Portugal (May 2008).

[Khan et al. 2012] Khan, R., et al.: "Future Internet: The Internet of Things Architecture, Possible Applications and Key Challenges"; 2012 10th International Conference On Frontiers of Information Technology (FIT), (Dec 2012), 257–260.

[Kranz 2017] Kranz, M.: "Building the Internet of Things: Implement New Business Models, Disrupt Competitors, Transform Your Industry"; J. Wiley & Sons (2017).

[Lin et al. 2017] Lin, J.: "A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications"; IEEE Internet of Things

Journal, 4, 5 (2017), 1125–1142.

[Ling 2013] Ling, Z.: "Research on application of Internet of Things in teaching practice of security technique defense [J]"; Internet of Things Technologies, 1, (2013), 029.

[Mukherjee et al. 2017] Mukherjee, M., et al.: "A Vision of IoT: Applications, Challenges, and Opportunities with Dehradun Perspective"; Proc. of International Conference on Intelligent Communication, Control and Devices, (2017), 553–559.

[Ngu et al. 2017] Ngu, A. H., et al.: "IoT Middleware: A Survey on Issues and Enabling Technologies"; IEEE Internet of Things Journal, 4, 1 (2017), 1–20.

[OpenFog 2017] OpenFogConsortium: "OpenFog Reference Architecture for Fog Computing"; accessed on 2017.12.27, www.OpenFogConsortium.org

[Palka et al. 2016] Palka, R., et al.: "QoS Mechanism for Low Speed Radio Networks – Case Study"; Choraś, R., eds., Image Processing and Communications Challenges 8, IP&C 2016, Advances in Intelligent Systems and Computing, 525, Springer, Cham, (2016), 240–246.

[Ray 2016] Ray, P.P.: "A survey on Internet of Things architectures"; Journal of King Saud University - Computer and Information Sciences, 30, 3 (2016), 291–319, https://doi.org/10.1016/j.jksuci.2016.10.003

[Raza et al. 2013] Raza, S., Wallgren, L., Voigt, T.: "SVELTE: Real-time intrusion detection in the Internet of Things"; Ad Hoc Networks, 11, 8 (2013), 2661–2674.

[Roman et al. 2013] Roman, R., Zhou, J., Lopez, J.: "On the features and challenges of security and privacy in distributed internet of things"; Computer Networks, 57, 10 (2013), 2266–2279.

[Sebastian and Ray 2015] Sebastian, S., Ray, P.P.: "Development of IoT invasive architecture for complying with health of home"; Proceedings of I3CS, Shillong, (2015), 79–83.

[Sivabalan et al. 2013] Sivabalan, A., Rajan, M. A., Balamuralidhar, P.: "Towards a lightweight internet of things platform architecture"; Journal of ICT Standardization, 1, 2 (2013), 241–252.

[Shang et al. 2012] X. Shang, R. Zhang, Y. Chen, "Internet of Things (IoT) Service Architecture and its Application in E-Commerce"; J. Electron. Commerce Org. (JECO), 10, 3 (2012), 44–55.

[Stanford-Clark and Truong 2008] Stanford-Clark, A. S., Truong, H. L.: "MQTT for sensor networks (MQTT-S) protocol specification"; (2008)

[Stojmenovic and Wen 2014] Stojmenovic, I., Wen, S.: "The Fog computing paradigm: Scenarios and security issues"; 2014 Federated Conference on Computer Science and Information Systems, Warsaw (2014), 1–8.

[strongSwan 2018] strongSwan: "Raspberry Pi 2 ESP Benchmark"; accessed on 2018.01.10, https://wiki.strongswan.org/projects/strongswan/wiki/RaspberryPi2Benchmark

[Sundmaeker et al. 2010] Sundmaeker, P. F. H., et al.: "Vision and Challenges for Realising the Internet of Things"; Pub. Office EU, (2010) http://www.internet-of-thingsresearch.eu/pdf/IoT_Clusterbook_March_2010.pdf

[Tankard 2015] Tankard, C.: "The security issues of the Internet of Things"; Computer Fraud & Security, 2015, 9 (Sept 2015), 11–14.

[Wu et al. 2010] Wu, M., et al.: "Research on the architecture of Internet of Things"; 2010 3rd International Conference On Advanced Computer Theory and Engineering (ICACTE), (2010), V5-484–V5-487.

[Yang et al. 2011] Yang, Z., et al.: "Study and application on the architecture and key technologies for IOT"; 2011 International Conference On Multimedia Technology (ICMT), (2011), 747–751.

[Yuan et al. 2007] Yuan, R., Shumin, L., Baogang, Y.: "Value Chain Oriented RFID System Framework and Enterprise Application Science Press"; Beijing (2007).

[Zhao et al. 2013] Zhao, K., Ge, L.: "A survey on the internet of things security"; 9th Int. Conf. on Comput. Intelligence and Security (CIS '13), (Dec 2013), 663–667.