UNIVERSIDADE FEDERAL DE PERNAMBUCO
CENTRO DE TECNOLOGIA E GEOCIÊNCIAS
DEPARTAMENTO DE ELETRÔNICA E SISTEMAS
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

SÉRGIO FRANCISCO TAVARES DE OLIVEIRA MENDONÇA

A BLOCKCHAIN-BASED ONTOLOGY FOR THE INTERNET OF THINGS SECURITY

Recife

2019

SÉRGIO FRANCISCO TAVARES DE OLIVEIRA MENDONÇA

A BLOCKCHAIN-BASED ONTOLOGY FOR THE INTERNET OF THINGS SECURITY

Thesis submitted to Programa de Pós-Graduação em Engenharia Elétrica of the Universidade Federal de Pernambuco, in partial fulfillment of the requirements for the Degree of Doutor in Engenharia Elétrica.

Area of concentration: Eletrônica

Advisor: Profa. Dra. Fernanda Maria Ribeiro de Alencar

Recife

2019

Catalogação na fonte
Bibliotecária: Maria Luiza de Moura Ferreira, CRB-4 / 1469

M539b Mendonça, Sérgio Francisco Tavares de Oliveira.
A blockchain-based ontology for the internet of things security / Sérgio Francisco Tavares de Oliveira Mendonça. – 2019.
152 folhas, il., tabs., abr., sigl.

Orientadora: Profa. Dra. Fernanda Maria Ribeiro de Alencar.

Tese (Doutorado) – Universidade Federal de Pernambuco. CTG. Programa de Pós-Graduação em Engenharia Elétrica, 2019.
Inclui Referências, Apêndices.

1. Engenharia Elétrica. 2. Internet das coisas. 3. *Blockchain*. 4. Segurança.
5. Privacidade. 6. Autenticidade. 7. Integridade dos dados. I. Alencar, Fernanda Maria Ribeiro de (Orientadora). II. Título.

UFPE

621.3 CDD (22. ed.) BCTG/2019–437

SÉRGIO FRANCISCO TAVARES DE OLIVEIRA MENDONÇA

"A BLOCKCHAIN-BASED ONTOLOGY FOR THE INTERNET OF THINGS SECURITY"

Tese apresentada ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Pernambuco, como requisito parcial para a obtenção do título de Doutor em Engenharia Elétrica.

Aprovada em: 30/08/2019.

BANCA EXAMINADORA

Profa. Dra. Fernanda Maria Ribeiro de Alencar
(Orientadora e Examinadora Interna)
Universidade Federal de Pernambuco

Prof. Dr. Marco Aurélio Benedetti Rodrigues
(Examinador Interno)
Universidade Federal de Pernambuco

Prof. Dr. Marcelo Brito Carneiro Leão
(Examinador Externo)
Universidade Federal Rural de Pernambuco

Prof. Dr. Fernando Antonio Aires Brito
(Examinador Externo)
Universidade Federal Rural de Pernambuco

Prof. Dr. Tiago Buarque Assunção de Carvalho
(Examinador Externo)
Universidade Federal Rural de Pernambuco

To my daughter Marina, for her kindness;
And my parents Uiára and Bartolomeu,
who give me strength, with all love and gratitude.

ACKNOWLEDGEMENTS

My gratitude is based on sincerity, respect, and the certainty that we are walking along familiar roads. In this journey, we set objectives, goals, dreams, and outline strategies to make them real, counting on support, fraternity, and the acceptance of our ideas.

Thus I would like to express my deepest gratitude:

To God for the strength, encouragement, and wisdom;

To my Advisor Dr. Fernanda Alencar for her mentoring, encouragement, seriousness, and commitment. I appreciate the support, even in difficulties, always given me the strength to never give up and go ahead;

To my thesis committee members, Dr. Marcelo Leão, Dr. Fernando Brito, Dr. Marco Benedetti, Dr. Tiago de Carvalho, Dr. Daniel Chaves, and Dr. Carla Schuenemann for their valuable advice through this process. Their knowledge, experience, and constructive criticism were decisive to the completion of this work;

To my fellow graduate students, professors, experts, collaborators, and all undergraduates for their singular contribution to this research. Especially to the members of Group on Requirements Engineering, Network, and Computational Intelligence (GRENCI), and to João Ferreira for their unique contribution. I'm deeply indebted to all of you;

To my parents, Uiára and Bartolomeu for the concern and efforts to ensure their children a quality education. Moreover, in unique, I thank Mrs. Emair (*in memoriam*), my grandma present in my heart, for our brief moments together and for all her care;

I likewise thank the friends that I have made on this journey. Additionally, for the dialogue opportunity with researchers in the engineering school of Pernambuco, CTG - UFPE, and also ones in the UFRPE.

"[. . .] We are left pondering some fundamental questions – what constitutes force? What is a hostile act? When is self-defense justified in response to a cyber attack? Is the use of traditional means of force ever justified in response to a cyber attack? These are not easy questions, and the international legal regime is lagging far behind the problems presented by the increasingly sophisticated technological possibilities in this area." (WALKER, 2001, p. 337).

ABSTRACT

Recent studies have revealed serious security breaches in the Internet of Things (IoT) devices. Today's architecture does not guarantee an adequate level of security, so attacks on data authenticity and integrity are among the top concerns when dealing with IoT-based environments. In this context, the objective of this work was to develop an ontology model for Blockchain-based IoT (BIoT) that ensures an adequate level of security. We implemented an ontology-based middleware that represents semantic knowledge. BIoT is independent of application context and protects against reported attacks from the fundamentals of blockchain networks. Initially, we built, through the hypothetical-deductive method, a BIoT model based on particular domain ontologies. We then interact between IoT devices and security ontologies and blockchain network concepts to capture characteristics. We then performed performance tests (sandbox); bench testing with Zigbee devices (testbed); knowledge base assessment; and research with experts through a questionnaire and semi-structured interviews to evaluate the proposal. We still adopt security criteria against possible known attacks in the literature. Thus, the ontology provided insight into security properties to monitor vulnerabilities in the IoT ecosystem and blockchain network structure, thereby ensuring data integrity, confidentiality, and privacy. Through the collected information, the BIoT model was built that presented the following advantages: adequate time processing; decentralized architecture, less susceptible to attack; presence of a stable network, even with the increase in the number of nodes, and consequently the packet traffic; possibility of improving the efficiency of data integrity verification; and increased availability of processing and memory resources for specific need environments. Thus, the model can be considered a promising alternative.

Keywords: Internet of things. Blockchain. Security. Privacy. Authenticity. Data integrity.

RESUMO

Estudos recentes apontam para graves falhas de segurança em dispositivos da Internet of Things (IoT). A arquitetura atual não garante um nível adequado de segurança e, por isso, ataques à autenticidade e integridade dos dados estão entre as principais preocupações quando tratamos de ambientes baseados em IoT. Neste contexto, o objetivo deste trabalho foi desenvolver um modelo a partir de ontologia para a IoT baseado em Blockchain (BIoT) que garanta um nível de segurança adequado. Foi implementado um middleware baseado em ontologia que representa um conhecimento semântico. BIoT é independente do contexto do aplicativo e fornece proteção contra os ataques relatados, a partir dos fundamentos das redes blockchain. Inicialmente, construiu-se, através do método hipotético-dedutivo, um modelo BIoT baseado em ontologias de domínios particulares. Em seguida, realizou-se a interação entre dispositivos da IoT e ontologias de segurança e conceitos das redes blockchain, para capturar características. Em seguida, foi realizado testes de performance (sandbox); testes de bancada com dispositivos Zigbee (testbed); avaliação da base de conhecimento; e, pesquisa com especialistas através de questionário e entrevistas semi-estruturadas para avaliação da proposta. Ainda adotamos critérios de segurança contra possíveis ataques conhecidos na literatura. Desse modo, a ontologia proporcionou conhecimentos sobre propriedades de segurança para monitorar vulnerabilidades presentes no ecossistema da IoT e estrutura de redes blockchain, garantindo assim integridade, confidencialidade e privacidade dos dados. Através das informações coletadas, foi construído o modelo BIoT que apresentou as seguintes vantagens: processamento de tempo adequado; arquitetura descentralizada, menos suscetível a ataques; presença de uma rede estável, mesmo com o aumento do número de nós, e consequentemente o de tráfego de pacotes; possibilidade de melhoria da eficiência de verificação da integridade dos dados; e maior disponibilidade de recursos de processamento e memória para ambientes com necessidade específicas. Assim, o modelo pode ser considerado uma alternativa promissora.

Palavras-chave: Internet das coisas. Blockchain. Segurança. Privacidade. Autenticidade. Integridade dos dados.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

B2B	Business-to-Business
B2C	Business-to-Consumer
BIoT	Blockchain-based Internet of Things
BLONDiE	Blockchain Ontology with Dynamic Extensibility
DBMS	Database Manager System
EVM	Ethereum Virtual Machine
IEEE	Institute of Electrical and Electronics Engineers
IEEE-SA	IEEE Standards Association
IoT	Internet of Things
KDC	Key Distribution Center
LWC	Local Water Control
M2M	Machine-to-Machine
MQTT	Message Queuing Telemetry Transport
NIST	National Institute of Standards and Technology
OWL	Ontology Web Language
RDF	Resource Description Framework
SDN	Software-defined Networking
SQL	Structured Query Language
TCP/IP	Transmission Control Protocol/Internet Protocol
W3C	World Wide Web Consortium

CONTENTS

1 INTRODUCTION

This chapter presents the research context, problem statement and hypothesis, research goals, methodological research procedures, and roadmap of this work.

1.1 RESEARCH CONTEXT

Computing is of great relevance to the areas of telecommunications and information security around the world. The growing number of devices for heterogeneous environments points to pervasive computing.

These devices interact with each other and produce data by mapping consumer habits (GUHA; KUMAR, 2018). Also, they enable automation of user tasks, e.g., the control of doors and windows by a security system.

Many security incidents are reported in heterogeneous environments (SFAR *et al.*, 2018). Security incidents are classified as spoofing, injections, unauthorized access, and traffic sniffing (ALI; SABIR; ULLAH, 2019). Attacks to billions of devices represent a challenge for smart device security experts.

1.2 PROBLEM STATEMENT AND HYPOTHESIS

The smart device, always connected to the Internet, is in the daily routine of millions of users. This trend refers to as the Internet of Things (IoT) and provides pervasiveness of such devices in our society. These devices interact with the environment to improve decision making or to track processes or events without human intervention.

Device-Environment interaction improves decision making, tracking processes, or events without human intervention. The IoT is part of a large number of different entities, such as devices (e.g., household, vehicles, available and management of items in supermarkets, measurement and performance statistics of athletes during a competition) (BANERJEE; LEE; CHOO, 2018).

Gartner predicted that by 2020, the number of connected devices surpasses 20 billion. (HUNG, 2017). Since IoT devices collect, process, and traffic user data through today's communications infrastructure. IoT systems generate massive data that demand network connectivity, processing power, and storage. Besides these infrastructure needs, other factors such as data security and privacy are fundamental.

Current systems provide a centralized infrastructure, providing access through a central authority, authentication, ensure privacy, and connect between the various nodes of the network. Besides, device integration in heterogeneous environments also causes security and privacy

issues because of technology vulnerabilities, conducting severe consequences to the users of IoT technologies.

Weber (2010) says that the technical architecture for the Internet has numerous security and privacy impacts. These issues address business processes and require reliability – the basic requirements: attack resiliency, authentication, and data integrity, access control, and privacy.

At the same time Blockchain has recently attracted the attention of several industries, from finance and healthcare to utilities. This interest in the Blockchain-based applications is justified by the need for applications that could not previously run without a trusted intermediary. And, with the adoption of Blockchain strategies, it is possible to operate without the need for a central authority (CHRISTIDIS; DEVETSIKIOTIS, 2016).

Thus the idea of applying Blockchain concepts to the IoT would address the application of an ontology. It establishes a set of formal terms, ensuring flexibility and interoperability in information representation, management, exchange, and discovery (NOY; MCGUINNESS, 2001). Ontology works as an explicit specification of concepts. And must be able to discover the best available resources dynamically according to established requirements.

So the device must contain entities with logical and formal representations that address issues of authentication and integrity. And it must provide terminology for Blockchain-based the IoT.

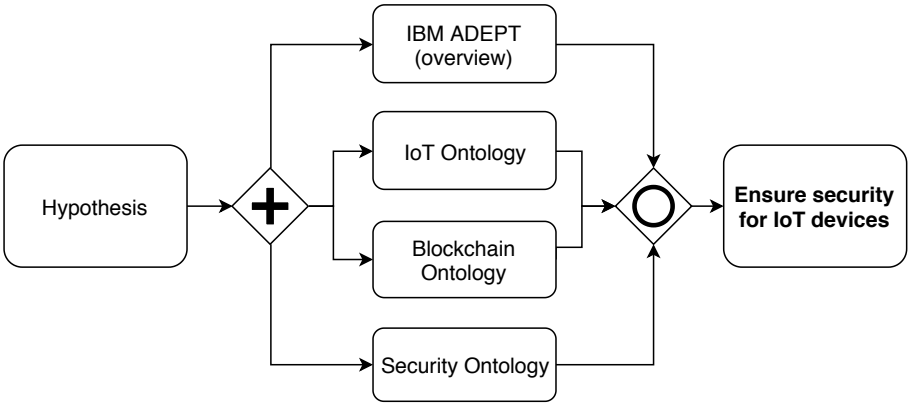
1.3 HYPOTHESIS

There is no rule to define hypotheses, but it is important that the researcher or student knows the bibliography of the area, observes the facts and has a basic notion of how to formulate hypotheses (which can be obtained by reading previous studies).

We should not elaborate a hypothesis only for the problem under analysis, but also fundamental hypotheses of the problem, which are different alternatives. In formulating the problem it is necessary that each alternative medium is specified and that one hypothesis is related to each alternative. Thus, a list of alternative hypotheses should be made and examined to eliminate those alternatives that are not against the purpose stated in the study.

1. **Blockchain** integration with **security** and privacy requirements ensures *resilience to attacks*, *data authentication*, *access control*, *client privacy* of **Internet of Things** through an **IBM ADEPT** overview, see a satellite view in Figure 1;
2. The models, architectures, frameworks, requirements, already defined in the literature and by corporate alliances, are sufficient (mature) to develop applications of the Internet of Things based on Blockchain;
3. The approaches of Blockchain-based Internet of Things applied in hybrid environments (scenarios), even in real-time applications have a significant relationship with processing, power consumption, storage, and memory usage (IBM ADEPT – overview).

Figure 1 – The hypothesis.



Source: The Author (2019).

However, most IoT devices are designed by people who have little expertise with computational security. Therefore the question is how to protect data through the method of Device-Network interaction once the current infrastructure cannot assure reliable security levels?

1.4 RESEARCH GOALS

The **general goal** of this research was to propose a Blockchain-based ontology for the Internet of Things Security. Thus the following steps were defined to achieve this primary purpose:

- To study security critical factors on IoT devices;
- To propose models for IoT devices based on Blockchain that improve security;
- To test the proposed ontology in online and offline scenarios;
- To evaluate the proposed solution with IoT and Embedded Systems experts.

1.5 METHODOLOGICAL RESEARCH PROCEDURES

Figure 2 presents the stages of research development, research classification, and data collection instruments. Still, we offer the method of development and evaluation. Contributions are organized according to each stage of work development, summarized as follows:

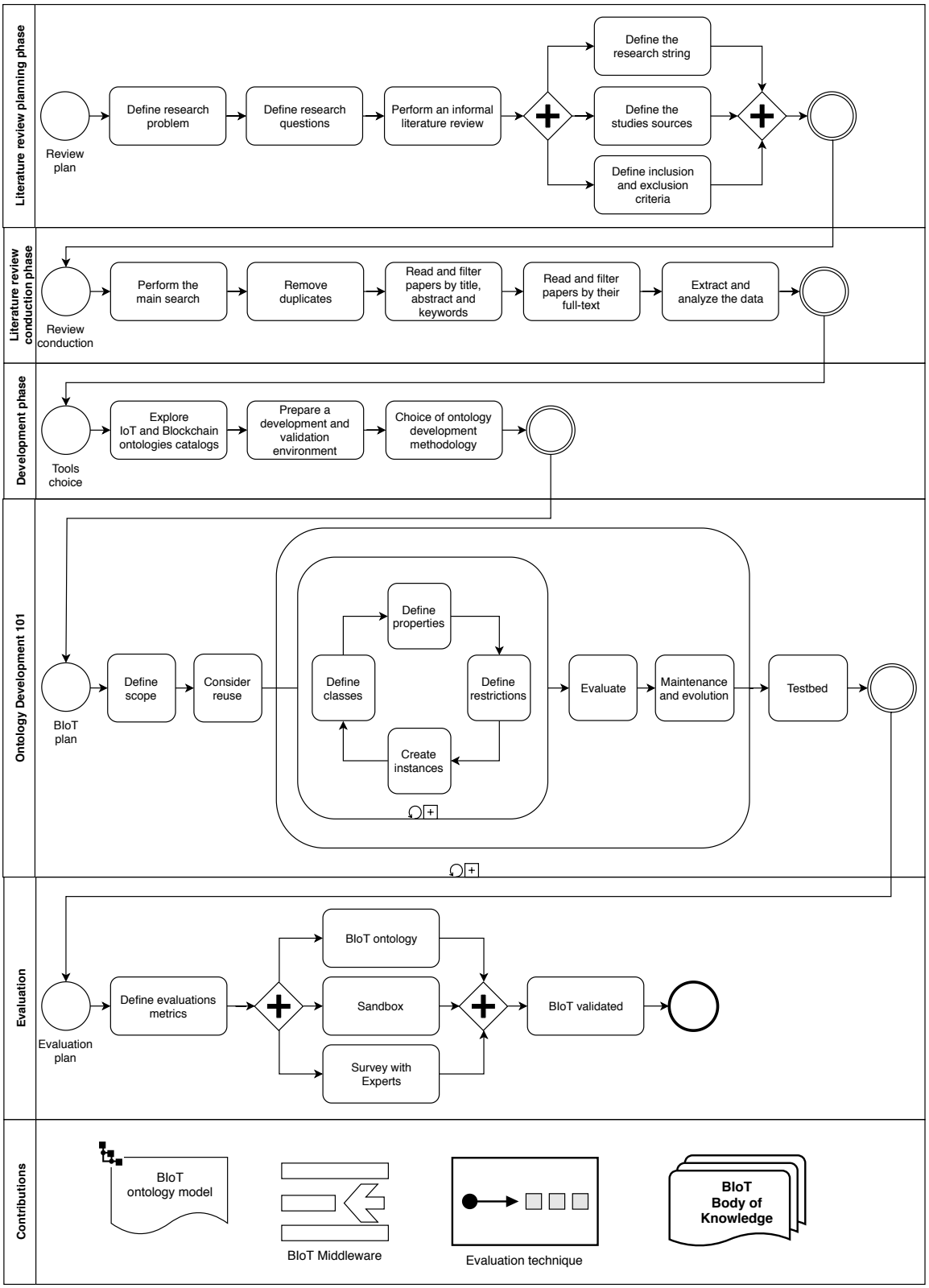
- We are proposing a Blockchain-based ontology for Internet of Things Security (BIoT). When using the proposed BIoT, key factors such as resilience to attacks; data authentication; access control; and privacy must be automatically addressed to the context, without system administrator needs or application of new rules.
- We are proposing a technique for using hybrid services through of the authorized Blockchain network. And, we also are presenting a body of knowledge over security techniques to IoT based on a Blockchain network.

- We have evaluated the performance of our proposed technique and compared it to the existing solution concerning various metrics.

1.6 ROADMAP

This doctoral thesis is organized into six chapters as follows: 1) Introduction - presents the characterization and relevance of the problem, research goal, hypothesis, contributions of this thesis; 2) Background and Challenges - reviews the Internet of Things, Embedded Systems, Semantic Web and, Blockchain foundations called; 3) Material and methods - planning and conducting of the research; 4) BIoT ontology – presents the proposal and available infrastructure of the execution; 5) Evaluation of the BIoT - describes the and presents the entire experiment; and 6) Conclusions and perspectives.

Figure 2 – Methodological research procedures.



Source: The Author (2019).

2 BACKGROUND AND CHALLENGES

In this chapter, we present more information about the Internet of Things and their standards, starting with the background about its basic paradigm. It contains an overview of the different approaches to standards by the leading responsible organizations when it comes to the Internet of Things-based Embedded Systems building, as follows.

2.1 INTERNET OF THINGS

The concept of "thing" in the Internet of Things means a physical entity of individual interest, such as a bicycle, an industrial machine, the air temperature in a room or monitoring movement to activate cameras or lighting control. Depending on the nature of the "thing" (device), different technologies are used to connect them to the IoT as identification devices (RFID, labels or bar codes), monitoring devices and actuators (temperature and other sensors, cameras in vehicles, door lock or window openings). We refer to a massive diversity of billions of devices and applications, and we aim for levels of interoperability in its multiple levels.

The IoT consists of a global network of billions of uniquely identifiable and addressable objects, embedded with sensors, actuators, and controllers) and these are connected to the Internet in wireless mode (ATZORI, 2017). The International Telecommunication Union determines the IoT as a "dynamic, global network infrastructure that can self-configure using standards and using interoperable protocols where (physical and virtual) things have identities, attributes, and personalities, use intelligent interfaces, and can seamlessly integrate into the network" (ITU Telecommunication Standardization Sector, 2012).

Institute of Electrical and Electronic Engineers (IEEE) presents IoT as an application domain that incorporates different technological and social fields, and address in its special report on Internet of Things issued in March 2014 (IEEE, "Internet of Things", 2014). IEEE described the phrase "Internet of Thing" as "*a network of items – each embedded with sensors – which are connected to the Internet*". It is a (non-approved) description of the Internet of Things. However, this statement is treating just an of the physical aspects of the Internet of Things (MINERVA; BIRU; ROTONDI, 2015). Despite various different definitions for the IoT, there can be named *Web of Things*, *Internet of Everything*, *Cloud* or *Fog Network*. These concepts are similar, and they belong to the same paradigm, both in pervasive and ubiquitous computing.

As a consequence of the numerous devices (or things) and their productive interactions, about 20 billion connected things will be in use by 2020. Moreover, regarding hardware expenses, consumer and enterprise applications will amount to about $3 trillion by 2020, according to Gartner (MEULEN, 2015). The Internet of Things ecosystem brought many advantages, but it will also produce a significant amount of data, which will require developers' concern for critical user

privacy issues such as personal data, behaviors, and preferences. Therefore, several organizations have presented privacy and trustability standards for IoT building.

2.1.1 IoT key requirements

The following we have presented the Uckelmann, Harrison and Michahelles (2011)' key requirements (kR) that need to be considered in the Internet of Things:

kR_1 Address the critical social needs of the IoT, including open governance, security, privacy, and reliability: The security and privacy framework must provide capabilities to dynamically adapt access rules and information granularity (for example, incorporate conditions into the practices of access or tokens assessed at access time).

kR_2 While there has been a clear focus on B2B requirements in recent years, B2C and M2M will gain importance in the future IoT. While B2C ease of use and human-readable data are essential, in M2M communications, data must be structured and machine-readable and semantically well defined.

kR_3 Create an open, scalable, flexible, and sustainable IoT infrastructure: The IoT has to be accessible by definition. Open standards are required to use and extend their functionality. It will be a vast network, considering that every object has its virtual representation. The IoT will need to be flexible enough to adapt to new requirements and technological developments.

kR_4 Developing migration paths for disruptive IoT technological developments: Instead of requiring disruptive new and parallel approaches, there need to be ways to integrate new developments into the core infrastructure; otherwise, there will be no guarantee of sustainability or lasting value. However, providing a migration path for standalone control in IoT would extend its use and provide reliable network infrastructure for autonomous objects.

kR_5 Encourage and enable businesses and people to contribute to the IoT: If stakeholders cannot benefit from the IoT, they will not participate. On the contrary, if people benefit from IoT, they will attract more people.

kR_6 Enable companies in different industries to develop high value-added products and services: A new business model based on information retrieval and input from/to the IoT is needed. Researchers can help identify new potentials, but business owners are required to increase the potential of the IoT.

kR_7 Encourage new market players, such as third-party information and service providers, to enter the IoT: Information on the Internet can be accumulated, processed, and sold regardless of the ownership of the physical product. Service providers should be encouraged, for example, to provide access to various sources of information about things and add technical billing capabilities for access to information.

kR_8 Provides an open solution for sharing costs, benefits, and revenue generation on the IoT: Information must be freely negotiable, regardless of a physical product. Today, the broader

use of IoT is often hampered by the lack of concepts about human, organizational, and technical disabilities to share costs and benefits or even generate revenue from IoT.

kR_9 Public initiatives to support the use of the IoT for socially relevant topics: Legislation has always been a lobbying mechanism for the adoption of new technologies. While it is evident that the IoT can be used to provide society with relevant data, some legislative requirements on topics such as carbon footprint, green logistics, and animal welfare would help to show the usefulness of IoT to society.

kR_{10} Let people identify things to access and contribute information without problems: Near Field Communication (NFC) is the next logical step for user interaction with the IoT. However, it is questionable how many cell phone owners will use these technologies. In addition to cell phones, there may be cheap dedicated devices. There are several opportunities to enable mass participation in the IoT.

We have identified that the Uckelmann, Harrison and Michahelles' key requirements kR_1, kR_2, kR_3, kR_4 and kR_{10} are related to the Back-end Internet of Things Core Architecture oriented to the IoT, it is the Scope of this research. While the requirements kR_5, kR_6, kR_7, kR_8 and kR_9 were excluded from the scope because they point to a social context of IoT application.

2.1.2 Internet of Things standards

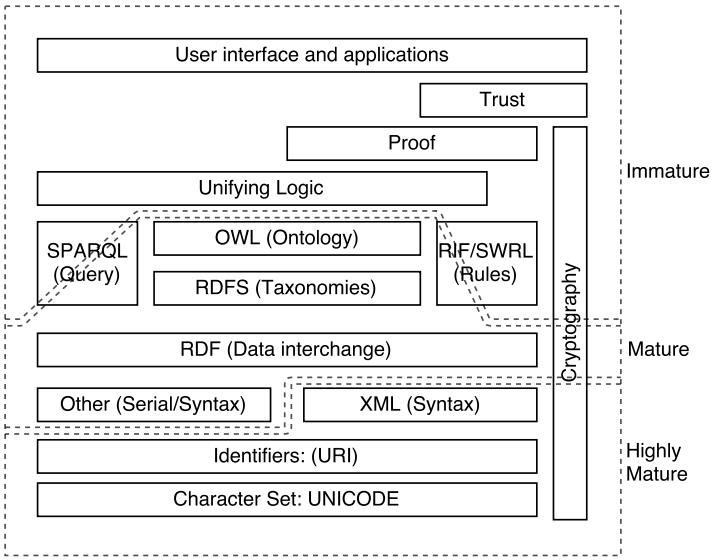
Several standardization organizations have described IoT requirements and established minimum parameters for building new solutions, such as W3C, in their texts on Semantic Web activities; besides, IEEE relating aspects of systems architecture, human factors and, entities of interest, establishing reference models for the development of IoT solutions.

2.1.2.1 W3C-based Internet applications

We must assume the capture of massive data produced by the reactions of the environment in which these solutions are immersed, and adapt them to Semantic Web Stack. The Semantic Web technologies are now, in their right, starting to reach a state of maturity. Generally, W3C Semantic Web Activity Lead describes these technologies concerning the Semantic Web Stack, as shown in Figure 3. Several of the elements described as layers (see Figure 3) in the Semantic Web Stack are already in place, although not nearly as widely implemented as most Semantic Web missionary would like it. (GREENBERG; RODRIGUEZ, 2012). Figure 3 presents Highly Mature, Mature and Immature areas. According to the Figure 3, we can observe that the components Cryptography, Unifying Logic, Proof, Trust and User interface, and applications are immersed in the Immature area.

The community classifies some technologies involved as highly mature, mature, and immature. The Resource Description Framework (RDF) and Ontology Web Language (OWL) technologies are considered the foundations of the Semantic Web, with recommended standards in 2004 were remarkably stable from its initial reviews (see Figure 4).

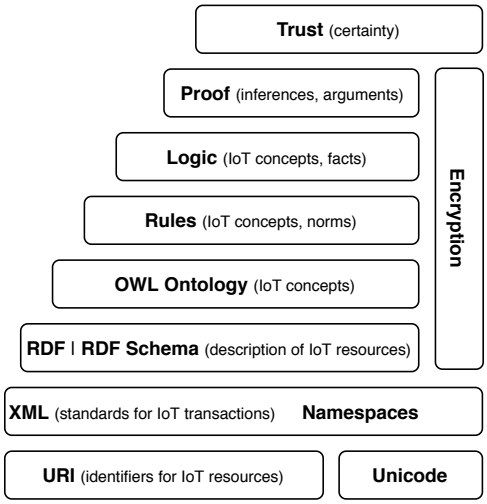
Figure 3 – W3C Semantic Web Stack.



Legend: Figure is highlighting immature, mature and highly mature areas of the W3C Semantic Web Stack, separated by dashed lines.

Source: Adapted from Wanjawa and Muchemi (2018).

Figure 4 – Ontology Web Language (OWL) Stack for IoT.



Legend: Figure is highlighting IoT concepts, facts, norms, description, resources, identifiers..., all described within parentheses.

Source: Adapted from Berners-Lee (1996).

Think of a complete software application can seem very complicated. If we visualize all aspects involved in the construction of an Embedded System, and try to relate them to Semantic Web Stack, we will be able to recognize some issues involved in programming languages oriented to objects and relational databases.

2.1.2.2 IEEE-based Internet of Things standards

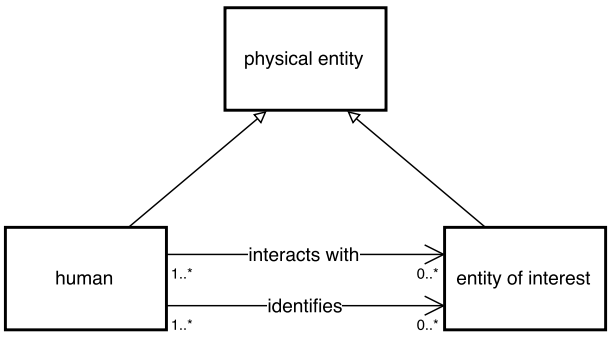
IEEE, a global association, applied to encourage innovation and technological excellence for the benefit of humanity, is the world's largest technical professional society. It is intended to serve professionals involved in any characters of the electrical, electronic and computing fields and relevant areas of science and technology that underlie modern civilization.

The IEEE Standards Association (IEEE-SA) is a globally recognized standards-setting body within IEEE. The IEEE-SA develops consensus standards through an open process that engages industry and stakeholder community and set specifications and best practices based on current scientific and technological knowledge.

The goals of the IEEE P2413 (see Figure 5) standard are to:

- accelerate the growth of the IoT Market by enabling cross-domain interaction and platform unification through increased system compatibility, interoperability, and functional exchangeability;
- define an IoT architecture framework that covers the architectural needs of the various IoT application domains;
- increase the transparency of system architectures to support system benchmarking, safety and security assessments;
- reduce industry fragmentation and create a critical mass of multistakeholder activities around the world;
- leverage the existing body of work.

Figure 5 – A satellite view of the IEEE P2413's goals.



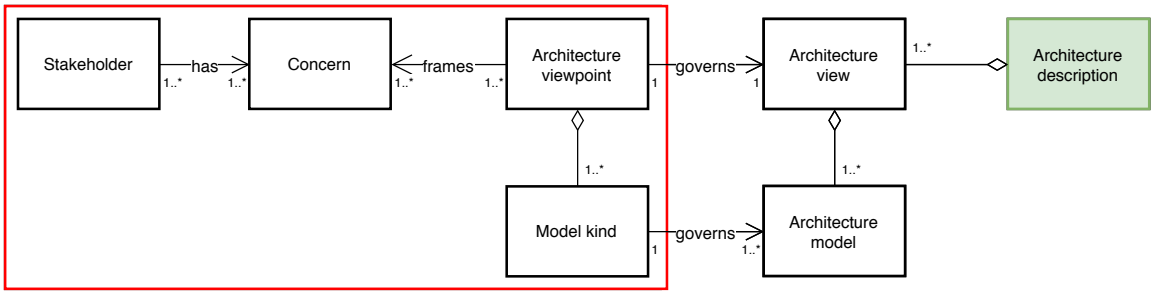
Source: Logvinov *et al.* (2016).

The IEEE-SA has identified over 140 existing standards and projects that are relevant to the IoT. One project that directly relates to IoT is IEEE P2413™ (see Figure 5). It defines an architectural framework, addressing descriptions of various IoT domains, definitions of IoT domain abstractions, and identification of commonalities between different IoT domains.

Figure 6 describes an overview of the IEEE P42010 architecture framework elements (MAY, 2011):

- *Stakeholder* individual, team, organization, or classes thereof, having an interest in a system or systems.
- *Concern* interest in a system or systems relevant to one or more of its stakeholders.
- *Architecture viewpoint* conventions for construction, interpretation and use of architecture views to frame specific system concerns.
- *Model kind* for a type of modelling as class diagrams, sequence diagrams...
- *Architecture view* work product expressing the architecture of a system from the perspective of specific system concerns.

Figure 6 – An overview of the IEEE P42010's architecture framework.



Source: May (2011).

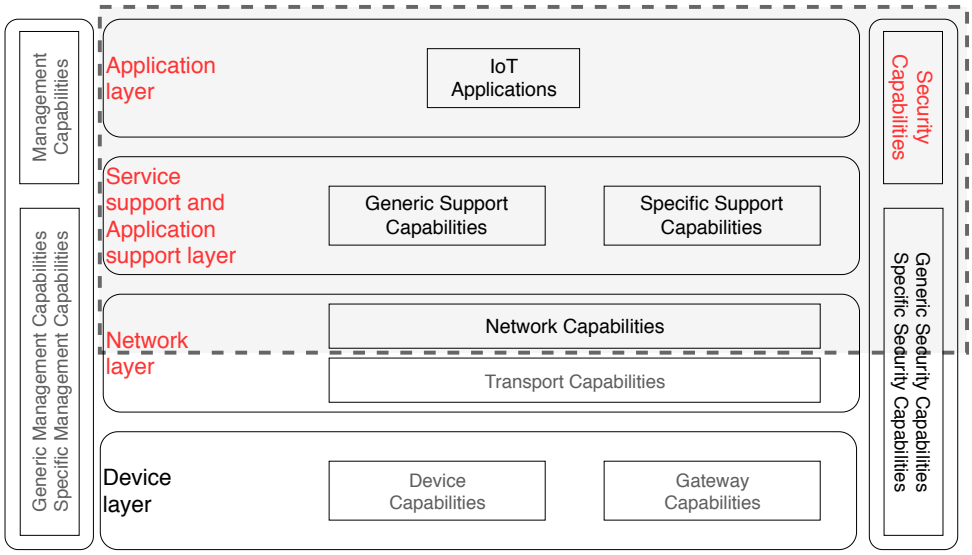
2.1.3 IoT reference model

ITU Telecommunication Standardization Sector (2012) presented the IoT reference model by Telecommunication Standardization Sector (ITU-T), composed of four layers (Figure 7):

- Application layer: Specifies the shared communication protocols and interface methods used for communication between hosts.
- Application and service support layer:
  - Generic support is standard features that can be used by different IoT applications, such as data processing or storage. These features may also be called by specific support features, for example, to create other specific support features.
  - Specific support is specific features that meet the needs of diverse application requirements. These can consist of several detailed capacity groupings, the to provide different application support functions from IoT.
- Network layer consists of two types of capabilities:
  - Network Features: Provide relevant control functions of the network connectivity such as access control functions and resource transport, management or authentication of mobility, authorization, and accounting.
  - Transport Capabilities: These provide connectivity for transporting IoT services and application-specific data information, and for carrying IoT-related control and management information.

- Device layer includes but is not limited to:
  - Direct interaction with the communication network: These can collect and upload information directly (i.e., without using gateway features) to the communication network and can receive data or commands directly from the communication network.
  - Indirect interaction with the communication network: These may collect and send information to the communication network indirectly, i.e., through gateway resources. Devices can receive data or commands indirectly from the communication network.
  - Ad-hoc Networking: These can build ad-hoc networks in some scenarios that require greater scalability and rapid deployment. Sleep and wake up: Device features can support sleep and wake up mechanisms to save power.
- Gateway Features: These include, but are not limited to protocol translation: There are two situations where gateway resources are required. One case is when device layer communications use different device layer protocols, for example, ZigBee technology protocols and Bluetooth technology protocols, the other is when the device layer and network layer communications use different protocols. For example, a device layer ZigBee technology protocol and a network layer 3G technology protocol.
- Multiple Interface Support: At the device layer, gateway features support devices connected through different types of wired or wireless technologies, such as a controller area network bus. (CAN) ZigBee, Bluetooth, or Wi-Fi. At the network layer, gateway capabilities can communicate across various technologies, such as public switched telephone networks (PSTN), second or third-generation networks (2G or 3G), long-term evolution networks (LTE), Ethernet, or digital subscriber lines (DSL).

Figure 7 – ITU-T IoT reference model.



Legend: Figure is highlighting area of interest of the BIoT, separed by dashed lines.

Source: Adapted from ITU Telecommunication Standardization Sector (2012).

This model includes management features and security features associated with the four layers. Management Capabilities cover the traditional classes of failures, configurations, accounting, performance, and security (FCAPS), such as fault management, configuration management, accounting management, performance management, and security management, and are also divided into generic and specific.

We identify the entities presented in the construction design of this prototype from Semantic Smart Gateway Framework, the IoT-ontology (KOTIS; KATASONOV, 2012), and the Technological W3C Stack (LI; XU; ZHAO, 2015). The purpose was to create a fully immersed application with IoT approaches, with semantic, entities, and, logical representations. For this proposal, we have considered the collection, transmission, data storage, such as security issues, privacy reliability, shipping, latency, and other aspects.

2.1.4 IoT ontology

The Internet of Things is provided by machine-to-machine communication and interactions between objects, devices, and people. Shortly, communications and information processing will be ubiquitous and performed through IoT systems.

> An ontology is a data model that represent a set of concepts into a domain and its relationship, inference about domain objects, describe individuals, classes, relationships, object properties. An ontology establishes a set of formal terms, ensuring flexibility and interoperability in information representation, management, exchange, and discovery. (NOY; MCGUINNESS, 2001).

It must be able to discover the best available resources dynamically according to established requirements. The device must contain entities with logical and formal representations that address issues of authentication and integrity.

Bermudez-Edo *et al.* (2016) say that for many years the Semantic Web community has developed ontologies to describe concepts and relationships between different entities in various domains. One of the significant problems for users and developers is the increased complexity and processing time that dynamic and responsive environments present, including the needs of IoT.

Complex models, while being possible to use in any environment, are often complicated to implement and also to use. These need high computational power and are often not suitable for restricted environments. IoT models should consider the constraints and dynamicity of IoT environments. At the same time, they must model the relationships and concepts they represent and allow interoperability between IoT entities.

2.1.4.1 DUL ontology

The DUL ontology is derived from the SSN ontology subset. It includes the classes and properties which are directly used by the SSN ontology and the associated classes and properties

which are required to have a standalone module. This ontology is part of the SSN ontology and defines ten classes and 16 properties.

2.1.4.2 SSN ontology

The Semantic Sensor Network (SSN) ontology describes sensors and observations and related concepts. This ontology does not describe domain concepts, time, locations. These are intended to be included from other ontologies via OWL imports. (HEFLIN; STUCKENSCHMIDT, 2012).

The SSN ontology is organized, conceptually but not physically, into ten modules, and consists of 41 concepts and 39 objects properties, directly inheriting from 11 DUL concepts and 14 DUL objects properties. (COMPTON *et al.*, 2012). The ontology can be seen from four main perspectives:

- a sensor perspective, with a focus on what senses, how it senses, and what is sensed;
- an observation perspective, with a focus on observation data and related metadata;
- a system perspective, with a focus on systems of sensors and deployments; and,
- a feature and property perspective, focusing on what senses a particular property or what observations have been made about a property.

In the proposals studied, many papers have begun to present the reuse of existing ontologies by combining them to propose a new ontology outside the domain of IoT. These new ontologies are usually developed to serve specific platforms to integrate data within IoT. However, these efforts always present a need for one or more concepts for the IoT domain. Among them, we can highlight **IoT-Lite**, as such, which provides a preliminary instance of the SSN ontology. The **VITAL ontology** follows the same approach, combining concepts from the ontologies SSN, QUDT, OWL-Time, and WGS84, to define sensors, measurements, time and localization concepts, respectively.

The OpenIoT ontology also uses the SSN ontology as a basis for its required concepts for IoT applications, such as OBSERVATION, SENSOR and LOCATION. The ontology IoT-O is modular and focused on two sets of requirements, Conceptual and Functional. These defined *conceptual requirements* as the requirements that form the basis of any ontology related to IoT while functional requirements have been defined as requirements that follow ethical practices defined by the Semantic Web community.

Other authors also present generic definitions, through ontological spectral schemes, to classify ontologies according to their level of expressiveness, from a low semantic level to very complex semantic relations, as we can observe in the ontologies IoT-Lite, FIESTA-IoT (AGARWAL *et al.*, 2016), oneM2M (ALAYA *et al.*, 2015a; ALAYA *et al.*, 2015b) and Open-MultiNet (WILLNER *et al.*, 2015). This work also has been inspired for an ontology of attacks and countermeasures for M2M communications by The IoTSec Ontology (Security Toolbox:

Attacks and Countermeasures) (GYRARD *et al.*, 2014; GYRARD; BONNET; BOUDAOUD, 2014b), cryptographic concepts and security properties by An ontology of Information Security (HERZOG; SHAHMEHRI; DUMA, 2007), Security in the Semantic Web using OWL (DENKER; KAGAL; FININ, 2005), and Security Ontology for Annotating resources. (KIM; LUO; KANG, 2005).

We use the same principle, reuse of the concepts discussed here, mainly the concepts of the SSN and IoT ontologies, trying to extract the more general concepts, properties and relations, in order to facilitate and facilitate their extension and interoperability among the various devices of the IoT ecosystem. Next, we describe the elements of the ontology used for our construction.

2.1.4.3 IoT-Lite Ontology overview

Classes: ActuatingDevice, Attribute, Circle, Coverage, Metadata, Object, Polygon, Rectangle, Service, TagDevice;

Properties: relativeLocation, altRelative, interfaceDescription, endpoint, exposedBy, hasAttribute, hasCoverage, hasMetadata, hasPoint, hasQuantityKind, hasSensingDevice, hasUnit, id, isAssociatedWith, radius, type, value.

The IoT ontology can be viewed as an extension to SSN ontology, by adding two new ontology layers to support the requirements for an IoT ontology. The layer for representing IoT entities and the layer for representing IoT entities' alignments. The IoT layers present as follows:

- IoT entities layer
  - iot:IoT_Entity
  - iot:SmartEntity
  - iot:ControlEntity
- IoT entities' alignment level
  - iot:OntologyAlignment
  - align:AlignmentCell

2.2 SECURITY CONCEPTS

Face a growing demand for new embedded devices connected in diverse environments. More significant problems arise regarding data security, privacy, resilience, storage, and centralization. Some of these risks are known, like malicious attacks on authentication, silent attacks on service integrity, attacks on network availability, such as the denial of service (DoS). (BORGOHAIN; KUMAR; SANYAL, 2015).

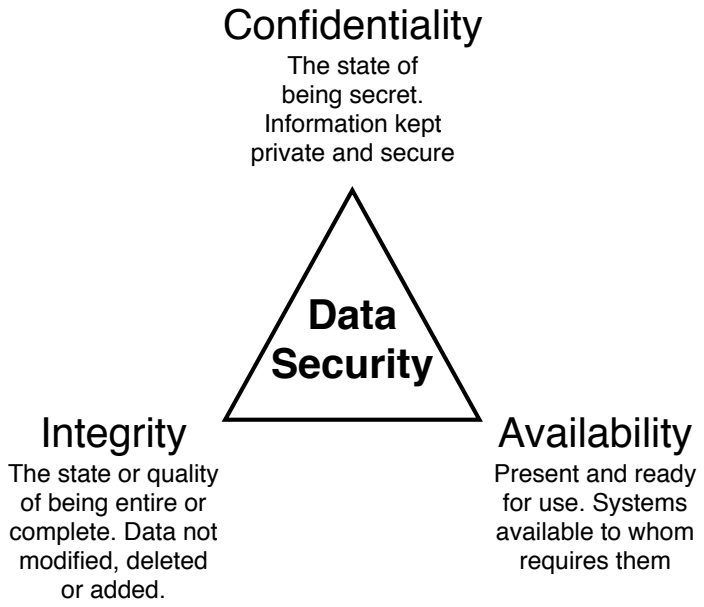
The NIST Computer Security Handbook (BAKER, 1991; GUTTMAN; ROBACK, 1995) defines the term *computer security* as follows:

> The protection afforded to an automated information system in order to achieve the applicable objectives of preserving the integrity, availability, and confiden-

> tiality of information system resources (includes hardware, software, firmware, data, and telecommunications).

The definition also has three key characteristics (STALLINGS, 2012), considered as main objectives or the classic triad of CIA (confidentiality, integrity, availability) for the area of computer security (see Figure 8):

Figure 8 – The CIA triad.



Source: The Author (2018).

2.2.1 IoT security

- *Confidentiality* is the protection of personal information and means keeping an information between server and clients. This term covers two other related concepts:
  - *Data confidentiality* ensures that private and confidential data is not available or disclosed to unauthorized people.
  - *Privacy* ensures that people control or influence what data related to them may be obtained and stored, as well as how, by whom, and to whom this information may be disclosed.
- *Integrity* refers to methods of ensuring that data is accurate, real and safeguarded for unauthorized modification. This term covers two other related concepts:
  - *Data integrity* ensures that data and applications are modified only in a specified and authorized way
  - *System integrity* ensures that a system performs its functionality safe, free of deliberate or negligent manipulation of the system.
- *Availability* refers to the ability of a client to access some resources or information in a determined place and in the correct format.

The growing concern of literature for the Internet of Things has recently prompted a significant number of technical and legal recommendations, to reduce its harmful effects. Privacy-by-policy and privacy-by-design have emerged as new approaches to the IoT and the tremendous, unprecedented flow of data it generates—which would require being thought and treated in itself as infrastructure. (GOODMAN, 2015).

Goodman (2015) presents some security recommendations and basic principles: (1) Device reliability; (2) data integrity; (3) safety for active systems.

- IoT systems should adopt security-by-design principles, deploying risk-based security measures.
- Governments should continue to develop protection, disaster response, and redundancy plans for critical infrastructure, focusing on the special threats that the IoT poses.
- The private sector and government should adopt expiration dates for autonomous IoT devices.
- There should be National Institute of Standards and Technology (NIST) standards development for safety and security practices, including resilience after a breach (e.g., cybersecurity framework, encryption).

2.2.2 IoT Security Standardization

We highlight three of the leading computer security standardization entities, The Open Web Application Security Project (OWASP), IoT Security Foundation, and European Telecommunications Standards Institute (ETSI), along with their IoT device security vulnerabilities and recommendations.

2.2.2.1 Open Web Application Security Project

Open Web Application Security Project (OWASP), an online community that creates and makes free articles, methodologies, documentation, tools, and technologies available in the field of web application security, related the top 10 IoT vulnerabilities that include the following topics (MIESSLER, 2015):

1. *Weak, guessable, or hardcoded passwords*: It is considered the main problem affecting IoT systems. Using a password policy composed of authoring rules is a solution used by companies to increase the security of these passwords. When there is a significant concern from the company or the IoT device setup, the risk of a system intrusion through password discovery is minimized.
2. *Insecure network services*: A most secure system begins by installing like a few packages and components as possible, especially those that implement network services. This minimization of components depends mainly on the context in which the system is involved. The justification for this recommendation is that it is common for unused services not to be monitored for security holes. Reducing the number of components (services) installed

decreases the chance that the system has a vulnerability that could be exploited by an attacker.

3. *Insecure ecosystem interfaces*: "Insecure web, backend API, cloud, or mobile interfaces in the ecosystem outside of the device that allows a compromise of the device or its related components. Common issues include a lack of authentication/authorization, lacking or weak encryption, and a lack of input and output filtering." It is not always evident whether interfaces are actually allowing compromise, but authentication, encryption, and filtering are still good ideas.
4. *Lack of secure update mechanisms*: "Lack of ability, firmware validation, secure delivery (un-encrypted in transit), anti-rollback mechanisms to securely update the device, and lack of notifications of security changes due to updates." Many vendors and companies don't bother to think about the future of their devices and implementations. Also, it is not always a technology problem. In some cases, the physical location of IoT devices makes upgrading, repairing, or replacing a significant challenge.
5. *Use of insecure or outdated components*: "Use of deprecated or insecure software components/libraries that could allow the device to be compromised. It includes insecure customization of operating system platforms and the use of third-party software or hardware components from a compromised supply chain." It increases the likelihood that an attack will be successful as vulnerabilities are more likely to be found by the attacker.
6. *Insufficient privacy protection*: "User's personal information stored on the device or in the ecosystem that is used insecurely, improperly, or without permission." The field of protection of personal data, using specific legislation and recognition in many countries, allows the individual to decide for themselves on the display and use of their data. Almost nothing that the device does with someone's personal information is correct unless they have permission from that person.
7. *Insecure data transfer and storage*: "Lack of encryption or access control of sensitive data anywhere within the ecosystem, including at rest, in transit, or during processing." It is necessary to understand better which ways data is used to travel across the network, that is, how files are shared. This way, you can get a good idea of the potential vulnerabilities present in this transfer and data process.
8. *Lack of device management*: "Lack of security support on devices deployed in production, including asset management, update management, secure decommissioning, systems monitoring, and response capabilities." IoT devices can be small, inexpensive, and deployed in large numbers, but that doesn't mean you don't have to manage them. It makes management more critical than ever even if it's not always easy, cheap, or convenient.
9. *Insecure default settings*: "Devices or systems shipped with insecure default settings or cannot make the system more secure by restricting operators from modifying configurations." A problem that can be solved through a previous study of the IoT device configurations that will be used in the device network. It allows the user not to be limited to certain types

of devices if you have to deal with the limitation of tools within your reach, with this pre-configuration study you can choose the equipment that best fits your needs.

10. *Lack of physical hardening*: “Lack of physical hardening measures, allowing potential attackers to gain sensitive information that can help in a future remote attack or take local control of the device.”. IoT devices must always be updated and enhanced to mitigate the effects of the vulnerabilities encountered. Effective measures must always be sought and taken to protect these devices involved.

We notice that OWASP collaborators paper, about 10 IoT vulnerabilities, there are factors which impact directly on human decision (it is not reported in this study, as such as Topics 1, 9 and 10); and other technical factors (it is goals this study, as such as Topics 2, 3, 4, 5, 6, 7, 8).

2.2.2.2 IoT Security Foundation

IoT devices, services and software, and the communication channels that connect them, are at risk of attack by a variety of malicious parties, from bedroom hackers to professional criminals or even state actors. Possible consequences to consumers of such an attack could include:

- Inconvenience and irritation;
- Infringement of privacy;
- Loss of life, money, time, property, health, relationships, etc.

For vendors, operators and suppliers, potential consequences may include loss of trust, damage to reputation, compromised intellectual property, financial loss and possible prosecution. Malicious intent commonly takes advantage of poor design, but even unintentional leakage of data due to ineffective security controls can also bring dire consequences to consumers and vendors. Thus it is vital that IoT devices and services have security designed in from the outset.

IoT devices can also be targets of intrusion like any other more robust computing equipment, and their functionality can be compromised to cause the user from irritation, financial loss, and even death. For the commercial sector, credibility losses, data leaks, lawsuits, and other harms that harm the company can happen. Therefore it is essential to think about the safety of these devices.

1. *Classification of Data*: Data can have security levels according to its importance.
2. *Physical Security*: Devices need to be physically protected as IOT equipment is often exposed and easily manipulated or accessed for extended periods and may contain sensitive information that could lead to damage.
3. *Device Secure Boot*: Prevent any unauthorized code from executing at boot time to ensure the reliability of the next stages.
4. *Secure Operation System*: Keep systems up to date with minimal access, sufficiently needed.

5. *Application Security*: Security in applications must be implemented from the outset, whether from own or third party projects, procrastination of security for when in production is a bad practice that poses a risk to the project.
6. *Credential Management*: Prevent unauthorized access to the system, identifies people, and promote secure communication.
7. *Encryption*: Ensure data protection and privacy to maintain system integrity and availability.
8. *Network Connection*: Limit network connection routes to valid devices to able a secure system.
9. *Securing Software Updates*: Updating systems produces bug fixes and security vulnerabilities, updates should be downloaded from vendors and installed, avoiding unknown or unreliable sources for updates.
10. *Logging*: Allow users and security products to analyze sensitive data to verify that the device is maintaining privacy. Make specific identifiers are removed or anonymized when necessary to prevent sensitive data from being collected and analyzed by unauthorized devices.
11. *Software Update Policy*: It is essential that processes and mechanisms for updating software are robust, reliable and secure. Devices must provide fault handling and status monitoring to meet availability requirements and so many other safety features are mentioned in the foundation for implementation.

2.2.2.3 European Telecommunications Standards Institute

Technical Committee (TC) CYBER (Cybersecurity) of European Telecommunications Standards Institute (ETSI) says that due the rapid evolution and growth in the complexity of new systems and networks, coupled with the sophistication of changing threats, present demanding challenges for maintaining the security of Information and Communications Technologies (ICT) systems and networks.

Security solutions must include a reliable and secure network infrastructure, but they must also protect the privacy of individuals and organizations. Security standardization, sometimes in support of legislative actions, has a key role to play in protecting the Internet and the communications and business it carries. They offer market-driven cybersecurity standardization solutions, along with advice and guidance to users, manufacturers, network, infrastructure and service operators and regulators. (EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE, 2019).

Cybersecurity for Consumer Internet of Things consists of the items as follows:

1. *No universal default passwords*: All passwords on IoT devices must be unique and never must be reset to universal default users and passwords. Because intruders easily obtain these passwords, so this practice has been a source of many problems in IoT and needs to

be discontinued.

2. *Implement a means to manage reports of vulnerabilities*: Companies providing connected devices and services should offer a public contact point as part of a vulnerability disclosure policy so that security researchers and others can report security issues found on their sold products so they can address the vulnerabilities. On-time, thereby avoiding compromising the security and privacy of your customers' data. Therefore, companies must continually monitor the services and devices they sell so that they can identify and correct security issues. Also, companies through vulnerability report management will be able to more quickly inform those most affected by these issues, which will help them more promptly take appropriate action to protect themselves from such effects.
3. *Keep software updated*: All software components on consumer IoT devices must be current. The customer must be notified by the responsible company (manufacturer or service provider) that an update is required. Also, the company must have a transparent and accessible policy to explicitly indicate to the consumer the minimum period for which the device will receive software updates and an apparent reason why this support period. Updates are intended to address issues and vulnerabilities in software on devices, so they are so important. For devices that do not have updates, these should be replaced, as they are more susceptible to attacks.
4. *Securely store credentials and security-sensitive data*: Because of the importance and sensitivity of user credentials and sensitive data, users must securely store them across services and devices. Therefore, reliable storage is a priority, as vulnerabilities and problems with this storage can damage system clients.
5. *Communicate securely*: Security-sensitive data, including any remote management and control, must be encrypted, with encryption appropriate to the properties of the technology in use, where all keys must be securely managed to make the communication process as secure and secure as possible. Reliable as possible.
6. *Minimize exposed attack surfaces*: The "least privilege principle" is a good practice of safety engineering, applicable to both IoT and any other field of application. The principle of least privilege is a security strategy, which is based on the idea of granting authorizations only when they are essential for the performance of a specific activity, i.e., software services should not be available if not used, such as an open port that is not required to run the service in question.
7. *Ensure software integrity*: Software on IoT devices must be verified using a secure boot mechanism, which requires a trusted root of the hardware. If unauthorized software changes are detected, the device should alert the consumer or administrator of the problem and should not connect to networks larger than those required to perform the alert function. The ability to remotely recover from these situations may depend on a well-known state, such as a locally stored version of consistent state software, to enable secure device recovery and upgrade. It will prevent denial of service and costly recall costs. Another benefit is

that if an IoT device has detected that something unusual has happened to its software, it can inform the right person, thereby speeding up the problem resolution process and further ensuring the safety and reliability of the system as a whole.

8. *Ensure that personal data is protected*: Device manufacturers and service providers should provide customers with clear and transparent information about how their data is being used, by whom, and for what purposes, for each device and service. It also applies to third parties who may be involved, including advertisers. It makes the whole process more transparent to the customer. Data are processed based on consumer consent, which may be revoked at any time by the user. Also, the appropriate entity, such as the device manufacturer's service provider, is expected to ensure that personal data are processed under data protection laws, thereby ensuring legal protection for the customer against the misuse of the data.
9. *Make systems resilient to outages*: Resilience must be incorporated into IoT devices and services, taking into account the potential for disruptions in the data and power networks. Therefore, as far as possible, IoT services should remain operational and locally functional in the event of a network loss and should recover cleanly in the event of a power outage. Devices should be able to return to a network in an expected and consistent state rather than large-scale reconnection. Because, as registered customers are increasingly relying on IoT systems and devices for increasingly essential use cases, where any problem could negatively impact their lives. The fault recovery capability for fault tolerance (IoT) is critical. By maintaining services performed locally (if network loss stops), resilience can be increased. Other measures may include redundancy in associated services as well as mitigations against, for example, attacks or Distributed Denial of Service (DDoS) signaling. Thus, the level of fault tolerance required is expected to be proportional and determined by use, as disruption of service may cause damage to its customers.
10. *Examine system telemetry data*: Telemetry is a technology that allows the measurement and communication of information of interest to the operator or system developer. If telemetry data is collected through IoT devices and services, it should be analyzed for security anomalies. Telemetry analysis, including log data, and its use, is beneficial for security vulnerability scanning, as well as allowing early identification of system abnormalities, thereby minimizing security risks and allowing rapid mitigation of the problems.
11. *Make it easy for consumers to delete personal data*: IoT devices usually change ownership and end up being recycled or discarded. Thus, devices and services should be configured so that data can be easily removed from them when there is a transfer of ownership. Therefore, consumers should be given clear and well-defined instructions on how to delete their data. When a consumer wants to delete their data, they also expect it to include backup copies that their service provider or device may have.
12. *Make installation and maintenance of devices easy*: Installation and maintenance of IoT devices must follow safety and usability best practices. Customers should also be provided

with security guidance for device usage, thereby mitigating security vulnerabilities caused by users with clear and accurate advice for securely configuring devices and minimizing system risks and vulnerabilities.

13. *Validate input data*: Data entry through user interfaces and transferred by (APIs) or across networks on services and devices must be validated. Validating incoming data ensures that the preconditions for services and devices are met to provide the correct service, and prevents the system from using malicious code.

2.3 THE BLOCKCHAIN

Blockchain consists of chained blocks of data that have been time-stamped and validated by miners, using elliptic curve cryptography (ECC) and SHA-256 hashing to provide strong cryptographic proof for data authentication and integrity. The block data applied by these technology contains a list of all transactions and a hash to the previous block. Moreover, the blockchain has a full history of all transactions and provides across-border global distributed trust; and during the blockchain use, each transaction in the shared public ledger is verified by a majority consensus of miner nodes which a reactively involved in verifying and validating transactions. Unlike, a TrustedThirdParties (TTP) or centralized authorities and services can be disrupted, compromised, or hacked. They can also misbehave and become corrupt in the future, even if they are trustworthy now. (ANTONOPOULOS, 2014; KHAN; SALAH, 2018).

For instance, when Blockchain is applied for Bitcoin, once transactions are validated and verified by consensus, block data are immutable, i.e., data can never be erased or altered. Blockchain can be built as (1) authorized (or private) network that can be restricted to a specific group of participants, or (2) permission-less or public network that is open for anyone to join in. Permission blockchains provide more privacy and better access control. (ZHOU *et al.*, 2018).

The New Yorker could rerun Peter Steiner's 1993 cartoon of one dog talking to another without revision "On the Internet, nobody knows you ar a dog." Online, we can not know if the identity provided is reliable or even trust each other to exchange money, without a bank or government validation. (TAPSCOTT; TAPSCOTT, 2018).

Before the advent of blockchain technologies, virtual money or virtual money transactions were performed through a central structure (a secure and reliable central server) that would prevent duplicate operations for these services.

Despite advances in the area of encryption, there was always a significant problem when it came to ensuring compatibility between centralization, anonymity, and the prevention of duplicate operations within the centralized framework.

Blockchain technology gained notoriety from a simple proposal: replacing the central server with a consensus mechanism based on proof of work.

The great news and improvement were in the decentralization of operations, where each

node in the network would be able to perform verification operations and return a validation to all other nodes in the network. The result of these operations would become known by all, validated by all the nodes in the network.

In scientific terms, the blockchain is a data structure that allows the execution of orderly transactions, with back-linked of blocks of transactions.

The Blockchain technology ensures the elimination of the double-spend problem, with the help of public-key cryptography, whereby each agent is assigned a private key (kept secret like a password) and a public key shared with all other agents. A transaction is initiated when the future owner of the coins (or digital tokens) sends his/her public key to the original owner. The coins are transferred by the digital signature of a hash. Public keys are cryptographically generated addresses stored in the blockchain. Every coin is associated with an address, and a transaction in the crypto-economy is simply a trade of coins from one address to another (PILKINGTON, 2016).

Blockchain is a trust-free, tamper-proof, auditable, and self-regulating system, with no human intervention required to execute computation. As a secure and decentralized computational infrastructure, it is widely acknowledged as a disruptive solution for the problems of centralization, privacy, and security when storing, tracking, monitoring, managing, and sharing data (ATZORI, 2015).

Blockchain can be stored as a flat file (files with records arranged in rows, where each record corresponds to a row where the fields are arranged positionally or by some kind of separator, such as a comma, for example, or in a simple database.

The block in the blockchain is back-linked to its previous block in the blockchain. The blockchain is often visualized as the vertical stack, with blocks layered on top of each other and the first block serving the foundation of the stack. The visualization of blocks stacked on top of each other results in the use of terms such as “height” to refer to the distance from the first block, and “top” or “tip” to refer to the most recently added block.

A hash identifies each block within the blockchain, generated using the SHA256 cryptographic hash algorithm on the header of the block. Each block also references a previous block, known as the parent block, through the “previous block hash” field in the block header. In other words, each block contains the hash of its parent inside its header. The sequence of hashes linking each block to its parent creates a chain going back to the first block ever created, known as the genesis block.

> Although a block has just one parent, it can temporarily have multiple children. Each of the children refers to the same block as its parent and contains the same (parent) hash in the “previous block hash” field. Multiple children arise during a blockchain “fork”, a temporary situation that occurs when different blocks are discovered almost simultaneously by different miners. Eventually, only one child block becomes part of the blockchain, and the “fork” is resolved. Even though a block may have more than one child, each block can have only

> one parent. It is because a block has one single "previous block hash" field referencing its single parent. (ANTONOPOULOS, 2014).

The "previous block hash" field is inside the block header and thereby affects the current block's hash. The child's identity changes if the parent's identity changes. If the parent is modified, then the parent's hash changes.

2.3.1 Structure of a block

A block is a container data structure that contains transactions for inclusion in the ledger, the blockchain. The main parts of a block are the header and the transactions. Transactions are the grouping of data that is stored in the block. The block is composed of a header that has several fields, of which the most important are: hash of the previous block, difficulty, nonce, and root of the Merkle tree; also composed of metadata: block height and header hash, which are stored in order to identify the block and its position in the chain, followed by a long list of transactions that make up the most significant part of the size of a block. The header block has 80 bytes; a transaction has an average of at least 250 bytes. A block consists of more than 500 transactions. These fields will be detailed below because Blockchain's correct understanding depends on them.

According to Antonopoulos (2014), a block is composed of the principal parts that are the header and the transactions. Transactions are the grouping of data that is stored in the block. Moreover, the header, in turn, has several fields, the most important for its operation: the hash of the previous block, difficulty, and nonce, as we will explain later. Besides these, it is also necessary to understand two important concepts: block height and the header hash, which have the function of identifying the block and its position in the chain, respectively. The following briefly presents each of these concepts (ANTONOPOULOS, 2014):

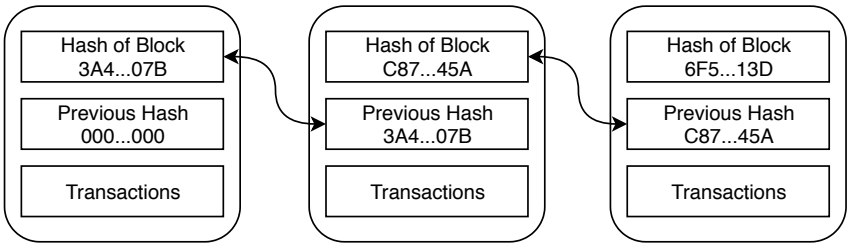
- *Transactions*: In Bitcoin, a transaction is a transfer of values. In a nutshell, it is a set of source addresses of the values and destination addresses to which the values will be sent. When a node creates a transaction, it sends it to its neighbors, which in turn forward the others until it reaches all the nodes in the network. When the transaction reaches a mining node, it saves it to include in the next block that will be mined. When the block is validated and included in the chain, the transaction will become public and unchangeable.
- *The genesis block*: The genesis block is software-level coded, and its main function is to serve as the initial state of the blockchain. The genesis block is known by all nodes of the blockchain network, containing specific rules, structure, and schedule of its creation, which guarantees the integrity of the data and invalidates any attempt of change in any block of the blockchain. The genesis block still contains a hidden message. This message should serve as proof of the creation of the first block.
- *Header block*: The header block consists of three sets of block metadata. In the first set of metadata, there is a reference to the hash of the previous block, which connects this block

to the previous block in the blockchain. In the second, there is a presentation of *difficulty*, *timestamp*, and *nonce*, used in mining competition. In the third set of metadata is Merkle tree root, a data structure used to summarize all the transactions contained in the block efficiently.

- *Nonce*: Consists of a number used as a variable to change the result generated by the header. It is used to prove that a miner has done work and was able to find a hash that is valid for the block, that is, that meets the criteria established by the network.
- *Block Height*: Blocks are included in the chain sequentially. The difference between the position of a block and the block of the genesis of the chain is called the height of the block.
- *Difficulty*: Difficulty is a partial hash collision. After the contents of the transactions are inserted into a block, its hash will be generated by the mining process. Hash algorithms always generate the same result if corresponding entries occur, so it is up to the computing power of the mining node to find a hash that satisfies this partial collision. For this, the nonce is used, as it is part of the header, whenever the hash of the header changes, so finding a suitable nonce that satisfies the network's difficulty is a task that requires high computational power, which implies in time mining and energy consumption.
- *Timestamp*: Next, to each block of the chain is stored the moment it was generated. This record is given by a number that represents the number of seconds elapsed between the moment the block was generated and January 1, 1970. .

A more detailed representation of the structure of the blocks and the main fields contained in them as shown in Figure 9. In the lower rectangle of each block in the figure are the transactions that are being added one by one until the block limit is reached. In the upper rectangle of the blocks is the representation of the header, containing the hash of the transactions, the hash of the previous block, the hash of the block that is its identifier, it is sent to the next block, also the timestamp recording the moment in which that block was generated and the nonce that is used for block validation.

Figure 9 – Blockchain structure.



Legend: Figure is highlighting the hash of the previous block, the hash of the block that is its identifier, it is sent to the next block on Blockchain.

Source: Adapted from Antonopoulos (2014).

2.3.2 Smart contracts

A smart contract is a program that runs on the blockchain and has its correct execution enforced by the consensus protocol (LUU *et al.*, 2015). A contract can encode any set of rules represented in its programming language, for instance; a contract can execute transfers when certain events happen (e.g., payment of security deposits in an escrow system).

We can apply smart contracts to many applications, from financial instruments and self-enforcing (e.g., currencies, financial derivatives, savings wallets, wills) to the conception of an autonomous government (e.g., outsourced computation, decentralized gambling). (LUU *et al.*, 2015; PEREZ; LIVSHITS, 2019).

Ethereum, a more recent cryptocurrency, is a prominent Turing-complete smart contract platform (BUTERIN, 2014). Unlike Bitcoin, Ethereum supports stateful contracts in which values can persist on the blockchain to be used in multiple invocations. In the last six months alone, roughly 15,000 smart contracts have been deployed in the Ethereum network, suggesting a steady growth in the usage of the platform. As Ethereum receives more public exposure and other similar projects like Rootstock (FALLIS, 2013) and CounterParty emerge on top of the Bitcoin blockchain, we expect the number of smart contracts to grow.

> A smart contract (or contract for short) is an "autonomous agent" stored in the blockchain, encoded as part of a "creation" transaction that introduces a contract to the blockchain. Once successfully created, a smart contract is identified by a contract address; each contract holds some amount of virtual coins (Ether), has its private storage, and is associated with its predefined executable code. (SULTAN; RUHI; LAKHANI, 2018).

A contract state consists of two main parts: private storage and the number of virtual coins (Ether) it holds (called balance). Contract code can manipulate variables like in traditional imperative programs. The code of an Ethereum contract is in a low-level, stack-based bytecode language referred to as Ethereum virtual machine (EVM) code.

Luu *et al.* (2016) say that users define contracts using high-level programming languages, e.g., Solidity (a JavaScript-like language), which are then compiled into EVM code. To invoke a contract at address γ, users send a transaction to the contract address. A transaction typically includes: payment (to the contract) for the execution (in Ether) or input data for the invocation.

Norta (2015) presents the core structure of a smart contract we will adopt to organize according to the interrogatives *Who* for defining the contracting parties together with their resources and data definitions, *Where* to specify the business and legal context, and *What* for specifying the exchanged business values.

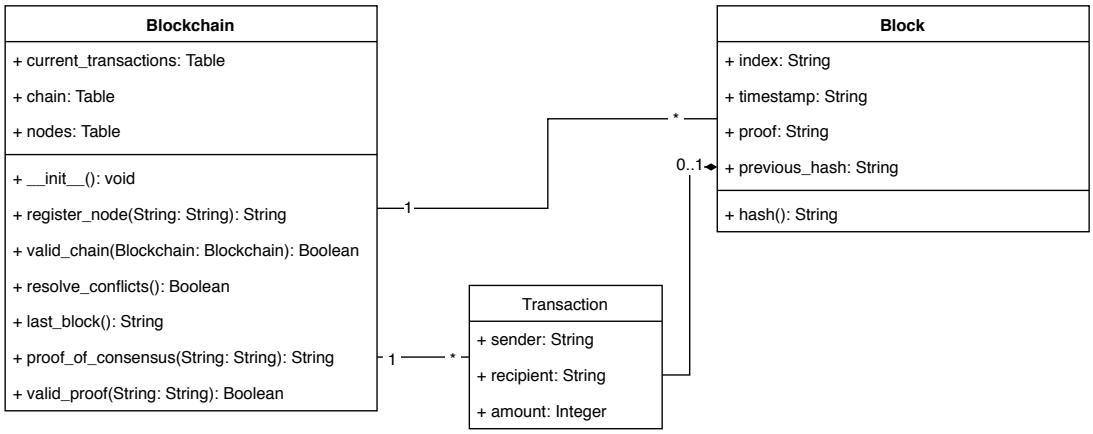
> The autonomy of devices: through smart contracts running on Ethereum, devices can autonomously execute payments, agreements, tradings, barters, and exchange of resources with other peers. They can also detect possible operational problems and do self-maintenance. (SULTAN; RUHI; LAKHANI, 2018).

2.3.3 Blockchain ontology

Ugarte (2017) also says that an ideal scenario would be that everyone would use only the original Bitcoin technology, and forks with minimum modifications. The protocol itself is already standardized and well-defined, but Bitcoin since presents many limitations and not being designed for other functionalities different than financial transactions, is not a realistic scenario.

Currently, the interoperability between Blockchain technologies is one of the most discussed issues in the Blockchain world, and this is where we must focus their efforts. We can see an Blockchain (it used on Bitcoin Technology) class diagram satellite view in Figure 10, that presents some entites and methods. Figure 11 provides an overview of the blockchain domain ontology in the UML Class Diagram.

Figure 10 – Blockchain class diagram – overview.



Source: The Author (2017).

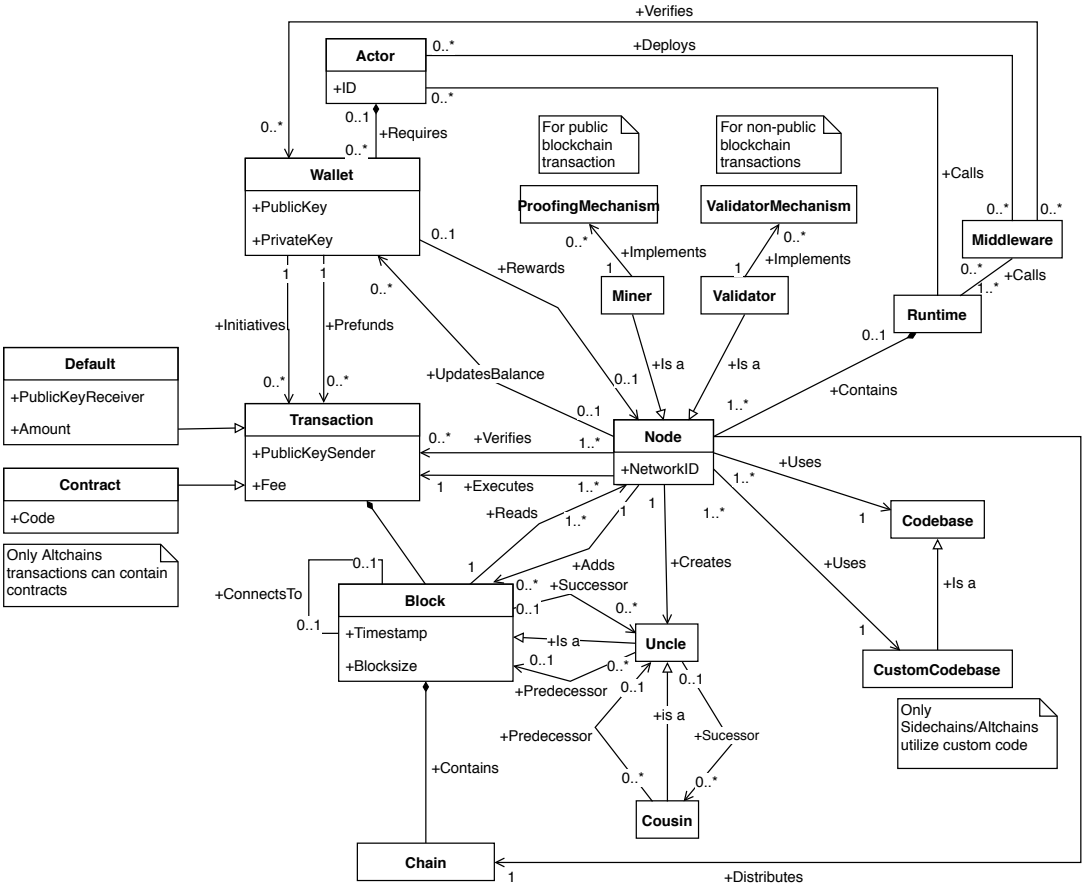
BLONDiE Blockchain ontology

A first effort to standardize this technology is the BLONDiE (Blockchain Ontology with Dynamic Extensibility) ontology. This OWL ontology can be used to express in RDF different fields of the structures of Ethereum or Bitcoin. It can also be extended to cover other Blockchain technologies. Also, BLONDiE being OWL can make explicit knowledge available. (UGARTE, 2017).

Brody and Pureswaran (2015) say that to be safe, scalable and efficient, the Internet of Things networks must be re-architected to gradually shift from managing billions of devices to hundreds of billions of devices, as see in Figure 12, device interaction timeline on closed and centralized IoT networks (before 2005); open access IoT networks, centralized cloud (today); and, open access IoT networks, distributed cloud (2025 and beyond).

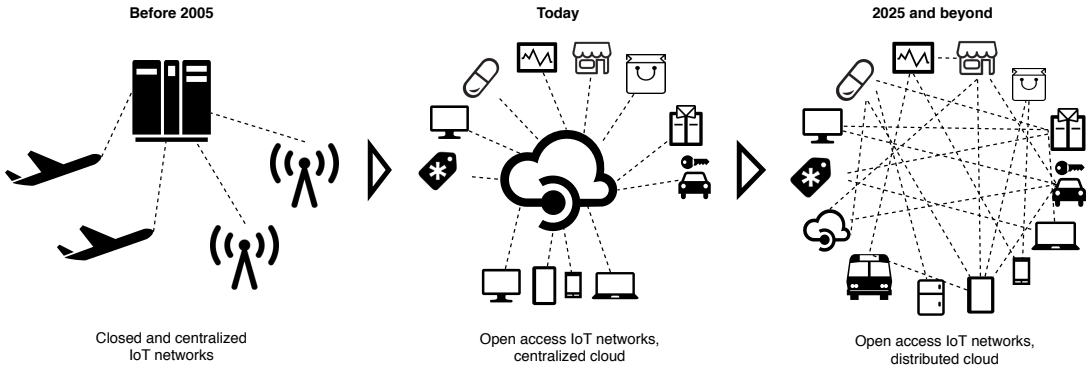
A user-centric model: devices will act in the best interest of the user, rather than third parties (e.g., manufacturers, governments, or service providers). Blockchain by default: products and devices should be registered by the manufacturer into a universal blockchain, at the beginning

Figure 11 – Datalogical Domain Ontology for a Blockchain Transaction.



Source: Kruijff and Weigand (2017).

Figure 12 – Device interaction timeline.



Source: Brody and Pureswaran (2015).

of their lifecycle.

ADEPT represents an interesting attempt of the industry to make the IoT ecosystem more sustainable, through a decentralized, peer-to-peer, and user-centric approach. The platform, however, still needs to overcome several technical issues, as author Signorini (2016) pointed out. Scalability: to manage a global blockchain for the IoT eventually becomes a tremendous

challenge, since the blockchain stores the records of all transactions back to the origin. According to IBM developers, sidechains, tree chains, and mini-blockchains may be used to address the problem. (PANIKKAR *et al.*, 2015).

> Peer-list: the blockchain can store the history of a smart object, but it is not designed to recognize the objects themselves. Therefore, a peer-list is required. After an object ID has been recognized, such ID can be used to browse the blockchain. Single Points of Failure: malicious users can exploit undisclosed or unknown vulnerabilities of the exchange nodes' code and potentially bring down the whole network. Privacy: all the nodes of the blockchain network have access to each others' transactions, so privacy is not guaranteed. (BRODY; PURESWARAN, 2015).

2.3.3.1 IBM ADEPT standards

Device democracy is the term used by research partner Samsung Electronics and IBM Institute for Business Value, to move towards blockchain solutions for the IoT, to develop a new business paradigm and vision of the world: the so-called Economy of Things. In a draft released in January 2015 and titled "ADEPT: An IoT practitioner perspective", the company proposed a blockchain-based project called ADEPT, namely Autonomous Decentralized Peer-to-Peer Telemetry (PANIKKAR *et al.*, 2015). The final version of such working paper was later released online as a report titled "Device democracy – Saving the future of the Internet of Things" (BRODY; PURESWARAN, 2015).

IBM recognizes the value of a blockchain-based decentralized approach to the IoT, to gain greater scalability, robustness, and security, as well as privacy-by-design. The result is "the Internet of Decentralized, Autonomous Things" (BRODY; PURESWARAN, 2015), a dynamic democracy of objects connected to a universal digital ledger, which provides users with secure identification and authentication. This concept, in IBM vision, is going to shape a brand new model of business in the very next future.

> IBM architectural approaches has evolved from an original viewpoint that all points in the network are equal towards one that recognizes some level of differentiation. In particular, IBM recognizes that many tiny devices may not have the full computational power and memory to manage the complete blockchain while others may be vital centers of commerce and interaction. Accordingly, our current architectural model has three levels of capability. (PANIKKAR *et al.*, 2015).

ADEPT architecture is based on TeleHash (as a messaging protocol), BitTorrent (as an efficient distribution layer) and Ethereum (as a platform for smart contracts and Decentralized Autonomous Organizations) (PANIKKAR *et al.*, 2015). Its main features can be summarized as follows. (PANIKKAR *et al.*, 2015; BRODY; PURESWARAN, 2015).

It is a business proposal using IoT. Developed by IBM in partnership with Samsung, the model aims to change the economic outlook by using the blockchain for the internet of things.

IBM believes this new model will achieve greater scalability, robustness, security, and privacy in projects that adopt this standard. The internet of things will be autonomous and decentralized.

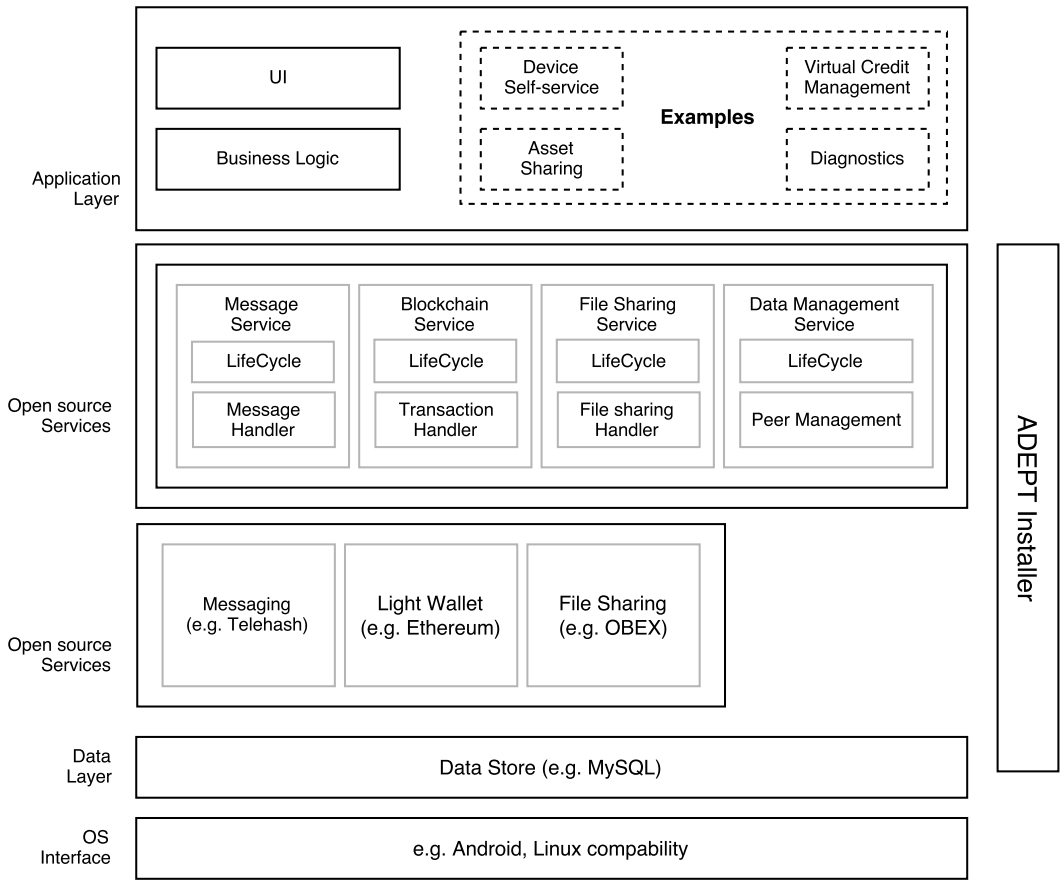
Not all IoT objects will have enough hardware to compute a blockchain; the architecture will be divided into three capacity levels, Light Peers, Standard Peers, and Exchange.

ADEPT will be a transparent and foolishly distributed system in which transactions are validated with the combination of proof of work and participation. Your architecture will be based on TeleHash, BitTorrent, and Ethereum.

- A transparent system and a fully distributed proof: transactions are validated through a combination of proof-of-work and proof-of-stake.
- An architecture suitable for different nodes, the nodes of the network can be distinguished according to their level of computational power and memory (PANIKKAR *et al.*, 2015; BRODY; PURESWARAN, 2015):
  - **Light Peers** are devices with low memory and storage capacity. It is assumed that light peers would not be able to store blockchains and would only retain their blockchain address within the device in what is described as a light wallet. For self-owned blockchain transactions, the light wallet would turn to another trusted pair. Light peer will execute messages, maintain a light wallet with their addresses and balance sheets, and make minimal file sharing. For example, receiving firmware updates or sending a summary of individual transactions to another pair based on a business or functional need. The reference architecture we have envisaged for a light peer is shown in Figure 13;
  - **Standard Peers** over the next few years, the processing power and storage capabilities of most products are expected to increase as the cost of manufacturing high-performance semiconductor chips decreases. The additional cost to the manufacturer or end consumer designing products to have this hardware would be minimal. Therefore, the washer or refrigerator of the future would be equipped with higher storage and processing capabilities that enable these products to meet blockchain requirements for a specified period, not only of themselves but also of peers who trust their customer's reliable products. The reference architecture we have envisaged for a standard peer is shown in Figure 14.
  - **Exchange (or ADEPT) Peers** are high-end devices with vast computing and storage capabilities. A market would require payment exchanges, analytical solutions, fraud detection, business and legal compliance packages, demand-supply matching solutions, and so on. Blockchain sizes can overgrow in a world where every city or community can have millions or hundreds of millions of IoT devices. However, with blockchain being the reliable source of information for all product transactions, it is essential to be able to access them at the regional or community level over time, in

some cases since the beginning of product life. We then need peers with significant processing and storage capabilities that can store the complete blockchain and make complex queries and analyses. The reference architecture is shown in Figure 15.

Figure 13 – ADEPT Light Peer Architecture - Logical View.



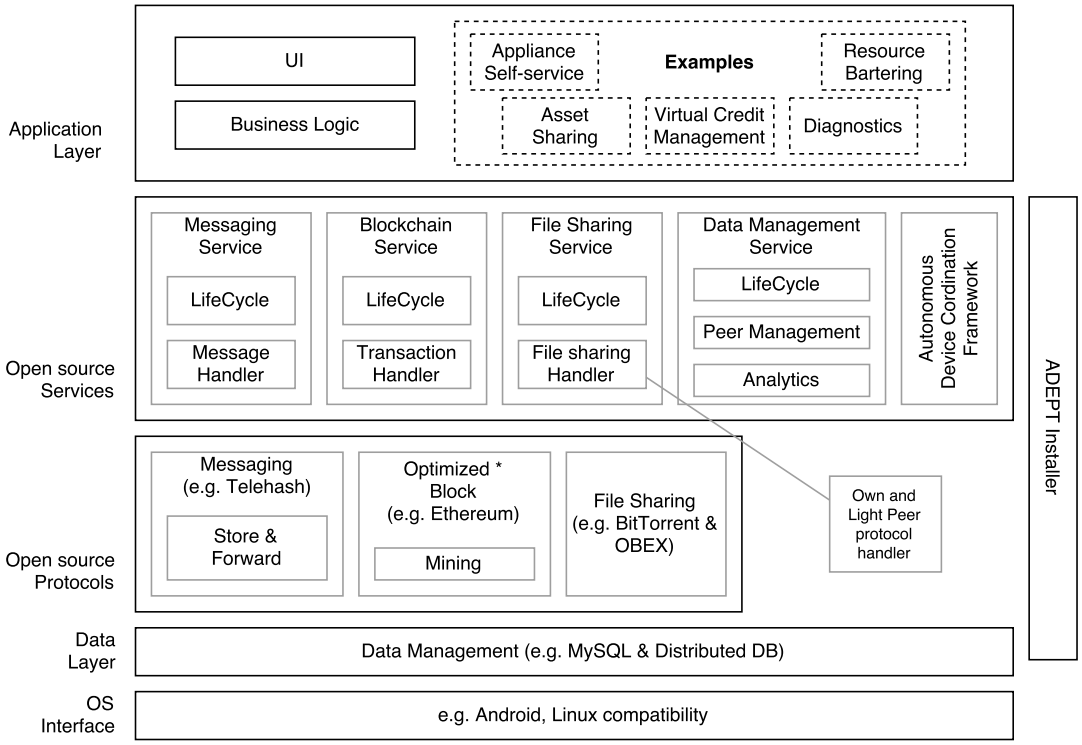
Source: Panikkar *et al.* (2015).

2.3.3.2 Foundational components for Proof of Concept

IBM selected the following open-source protocols to implement an ADEPT Proof of Concept (PoC): Telehash, BitTorrent, and Blockchain Technology (Ethereum).

1. **TeleHash**: Of the many messaging protocols it considered, TeleHash seemed the most promising in approach and ideological match to decentralized approach on IoT based its Kademlia protocol based Distributed Hash Table implementation.
2. **BitTorrent**: BitTorrent utilizes bandwidth efficiently while discouraging leeching. IBM envisions Torrent file sharing solutions being a critical part of the ADEPT architecture.
3. **Ethereum — Blockchain Technology**: Ethereum's improvements to the traditional blockchain approach of Bitcoin and the Turing complete scripting languages they introduced were extremely compelling. The ability to create binding contracts and potentially Decentralized Autonomous Organizations led us to pick Ethereum as our PoC's blockchain technology. (PANIKKAR *et al.*, 2015).

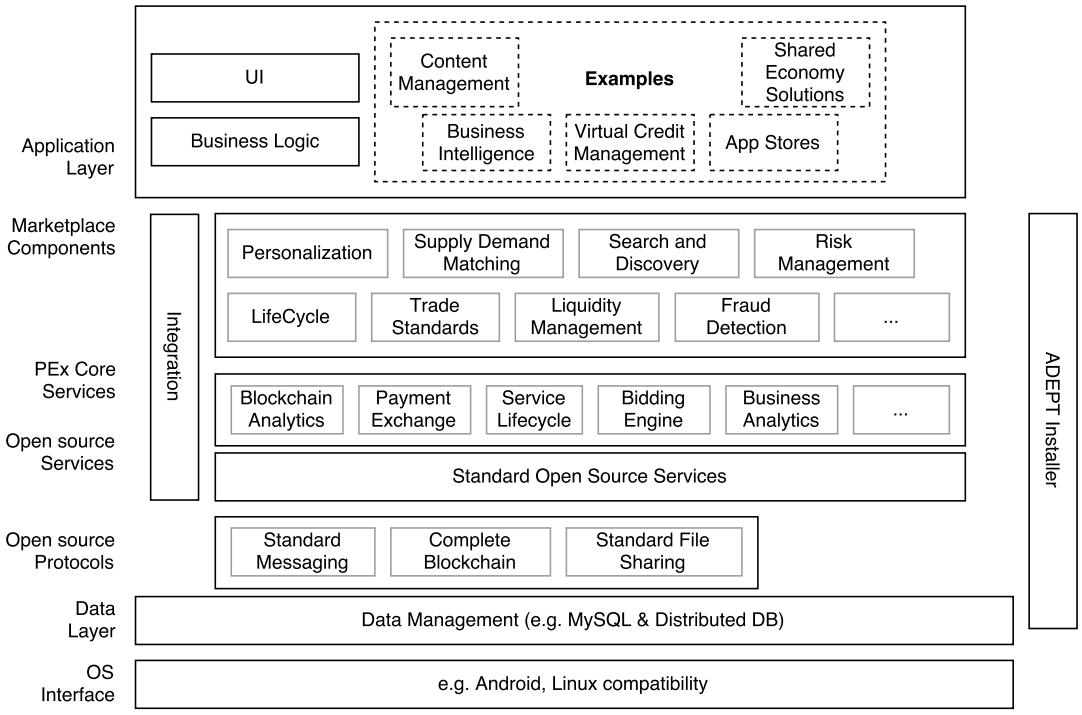
Figure 14 – ADEPT Standard Peer Architecture - Logical View.



* Could be optimized to hold the complete blockchain. Function of ADEPT Installer

Source: Panikkar *et al.* (2015).

Figure 15 – ADEPT Peer Exchange Architecture - Logical View.



Source: Panikkar *et al.* (2015).

2.3.4 Concluding remarks

We present a summary background about principles and applications to the Internet of Things and Blockchain technologies. We detect in related work the main features, challenges, and trends of some platforms for the IoT. We identify very much standards and a relevant project, as IEEE standards for the IoT and, W3C-based the Internet applications and, the key requirements to the Internet of Things. The Blockchain approach is trust-free, tamper-proof, auditable, and self-regulating system with no human intervention required to execute computation through a smart contract.

A smart contract can encode any set of rules. Finally, we summarize the core concepts about IBM ADEPT peer architecture and their open protocols to implement an ADEPT platform and concepts. In the next chapter, we present the preliminary studies results in Blockchain-based the Internet of Things: a systematic mapping. We describe the research questions, protocol, conduction, reporting, current trends, and challenges threats to validity. We were interested in answering the research questions: i) Has Blockchain-based IoT been constructed to stand on development processes? Which are those processes? ii) Which Blockchain-based IoT characteristics, principles, or requirements have been considered in Blockchain-based IoT development processes? We also present in current trends and challenges section the main discussions by the authors of the included papers about Blockchain-based the Internet of Things.

We presented several concepts that will be used to build this Blockchain-based the Internet of Things architecture. We described an overview of general usage scenarios that help and address our aim, objectives, and scope of this ontology proposal. We also presented a real scenario on the domain of the expertise water government provider to identify the main entities and their concepts. We summarize the DUL and SSN ontologies and their classes and properties. We intend to include them and also include the Blockchain and the Internet of Things ontologies. From the concepts found, we began to join these concepts to the creation of Blockchain-based the IoT architectural. Finally, we briefly describe our environment of building our knowledge domain through the Protégé application. In the next chapter, we discuss initial challenges and research opportunities to improve the understanding about an open problem in components cryptography, unifying logic, proof, trust as in the immature area and Blockchain-based the IoT development cycle.

3 MATERIAL AND METHODS

This chapter gives an overview of the research approach adopted in the thesis. It presents the path and strategies to achieve objectives. Also presents the preliminary results that point to the route of this thesis proposal. The results are reported in the systematic mapping, as follows.

3.1 INTRODUCTION

The science is a cooperative social activity, and scientific knowledge is the result of a cumulative process of this cooperation (BIOLCHINI *et al.*, 2005). Scientific knowledge is developed through several ways and approaches, actually defined as scientific methodologies.

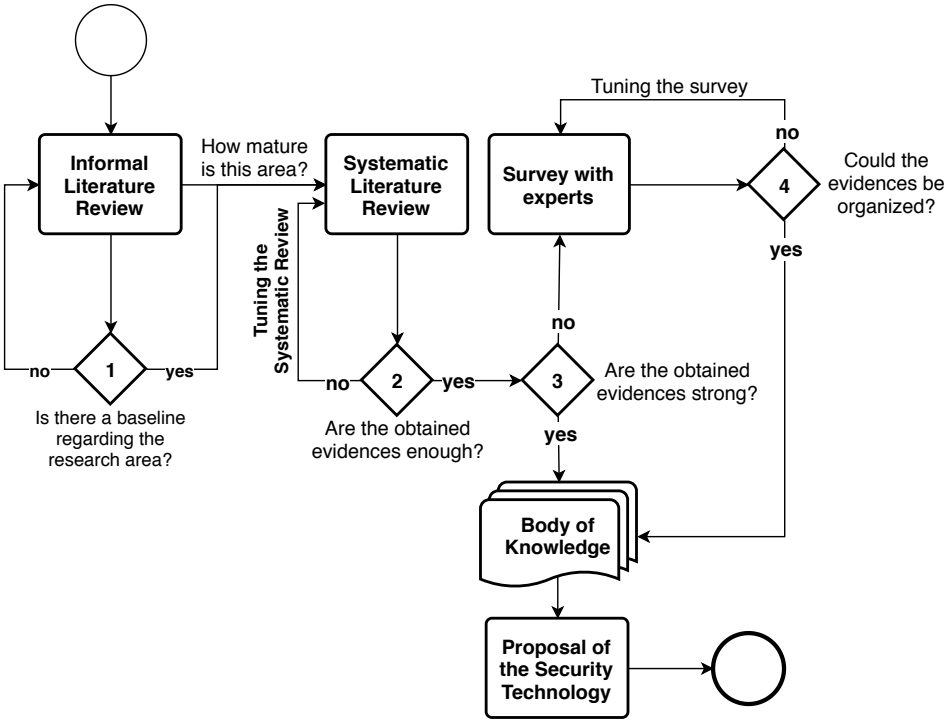
The scientific methodology is essential for academic studies. Initially, we present the etymological definition of the methodology term, Greek origin "*meta*" = after, besides, with, changed, different, beyond, adjacent, self; "*odos*" = a threshold, way, path, track, road, highway; "*logos*" = the word, a ground, a plea, an opinion, speech, study. (LIDDELL *et al.*, 1996; PRODANOV; FREITAS, 2013).

Method of approach. The method is the way, the form, the way of thinking. It is an approach to the level of abstraction of phenomena. It is the set of processes or mental operations employed in the research. In this research, we adopt the **hypothetical-deductive** method. The hypothetical-deductive method, as defined by Karl Popper from critiques of induction, expressed in The Logic of Scientific Research, a work first published in 1935 (PRODANOV; FREITAS, 2013). The hypothetical-deductive method begins with a problem or a gap in scientific knowledge, through the formulation of hypotheses and a process of deductive inference, which tests the prediction of the occurrence of phenomena covered by said hypothesis.

The research methodology (see Figure 16, based on research method by Dias-Neto, Spinola and Travassos (2010)) was divided into four steps to developing the proposed IoT-based unified logic layer model for embedded systems projects, (see Figure 17, based on research method by (DIAS-NETO; SPINOLA; TRAVASSOS, 2010)) and, as described in detail below.

First, we built an satellite view from isolated domain ontologies that do not interact with each other. Then, we built middleware to perform the interaction between the IoT devices to perform the interaction between devices and the proposed BIoT ontology. Next, we conducted a Survey with experts and semi-structured interviews to evaluate the new version of the solution. After that, security criteria have been validated through security strategies against possible attacks known in the literature. Finally, the proposal was validated in a controlled environment and with experts to confirm its effectiveness and efficiency, grouping levels, and resolution of vulnerability factors presented.

Figure 16 – Research strategy.



Source: The Author (2015).

The proposed ontology middleware and foundations called the Blockchain-based ontology for the Internet of Things Security (BIoT), are context-adaptive and service-based as an additional layer to IoT reference layers. BIoT is structured on a minimum basis, and on-demand services, based on the layers of perception (sensors), treatment and adequacy of the acquired data (from analog to digital), in the encryption and availability of symmetric keys between nodes, communication, and registration is allowed blockchain and controlled environment.

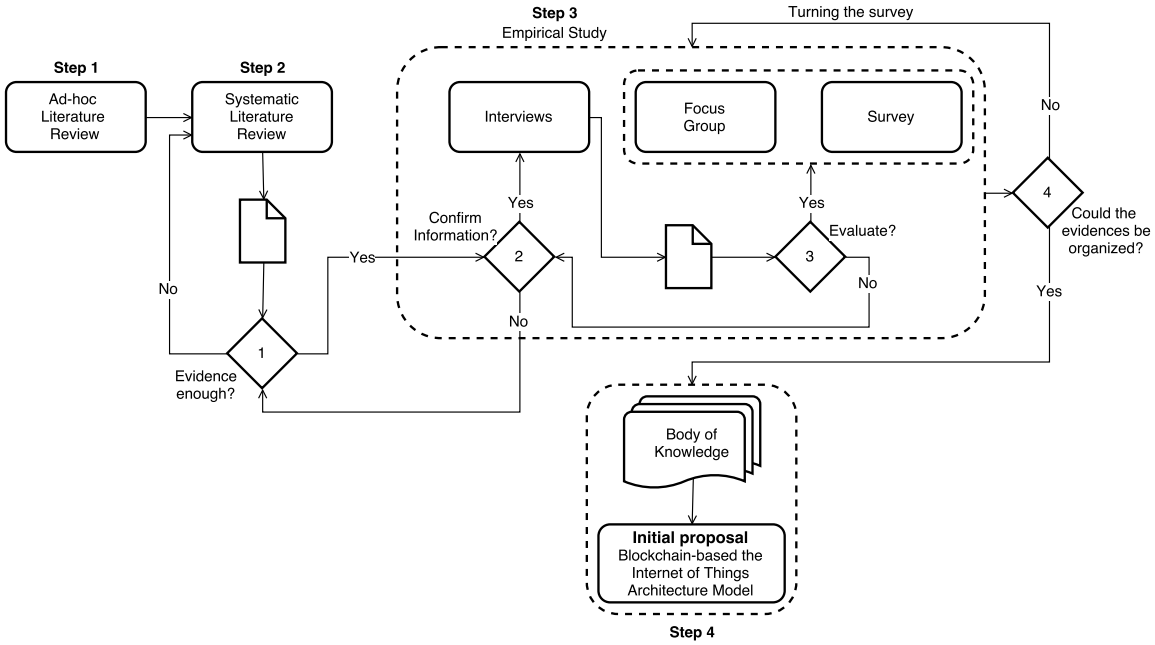
3.2 STEP 1 — AD-HOC LITERATURE (OR INFORMAL) REVIEW

The informal (unsystematic) review is usually an initial step in any research and development enterprise and was used to confirm the choice of a research area to be worked on and to identify evidence published in this study area. It also provides an initial conception of technologies in the research.

3.3 STEP 2 — SYSTEMATIC LITERATURE MAPPING

Systematic review was used to analyze of the previous findings, techniques, ideas, and ways to explore the topics in question, as well as their relevance to the issues of interest and synthesis and summarization of this information. Also, the snowball sampling technique was used to recruit experts in the research area in this thesis.

Figure 17 – Scientific methodology steps.



Source: The Author (2016).

Systematic mapping study is indicated during a first incursion into the topic discussed, before starting a systematic review, we came across a general question. To obtain an overview of this research topic and identify evidence to provide the best positions on the issues of research, we have established a systematic mapping. (KITCHENHAM; CHARTERS, 2007).

Kitchenham and Charters (2007, p. 5) still state that systematic mapping allows:

- mapping the evidence of a domain at a high level of granularity;
- the identification of clusters and voids of evidence to enable future systematic reviews;
- discover areas to conduct new primary studies.

This study aims at moving towards a consolidated knowledge about Internet of Things, Semantic Web, and Embedded Systems areas by developing a better understanding of which factors influence in IoT-based Embedded Systems projects. To achieve the goal of this study, we are conducting a systematic mapping of critical factors in IoT paradigms-based embedded systems building.

3.4 STEP 3 — EMPIRICAL STUDY

An empirical study aims to understand the context of a situation and creatively, to interpret and describe the complexity of a concrete case by exhaustively deepening a delimited object.

Coutinho (2005) says that the goal of this type of research is always holistic (systemic, broad, integrated). Researcher seeks to extract information from real context, in-depth, through

questionnaires, interviews, observations, written documents and records, field notes and journals, photos, audiovisual records, testimonies, Internet searches, among other methods.

Interviews

The qualitative research interview seeks to describe and the meanings of central themes in the life world of the subjects and, to cover both a factual and a meaning level, though it is usually more difficult to interview on a meaning level. The main task of interviewing is to understand the meaning of what the interviewees say (KVALE, 1996).

Interviews are particularly useful for getting the story behind a participant's experiences. The interviewer can pursue in-depth information around the topic. Interviews may be useful as follow-up to satisfied respondents to questionnaires,e.g., to further investigate their responses (QU; DUMAY, 2011).

Focus group

The focus group is a data collection method. The data is collected through a semi-structured group interview, and a group leader moderated it. Focus group is used to collect data on a specific topic (WILKINSON, 1999).

The focus group is a research technique that collects data through group interaction on a topic determined by the researcher. In essence, it is the researcher interest that provides the focus, whereas the data themselves come from the group interaction (MORGAN, 1997).

We arranged focus group sessions with the purpose to measure the preliminary version of Blockchain-based Internet of Things Security. The preliminary evaluation pointed out ways to build the Blockchain-based the Internet of Things Security. That also provided input to the design of a survey for final evaluation through expert opinion.

3.4.1 Test case ontology and performance metrics

Evaluation of BIoT ontology

IERC AC4 provides a set of best practices, a range of activities to support IoT project cooperation activities, such as workflows for well-defined technical activities. However, it does not provide: i) methodologies for reusing ontologies, ii) tools for validating ontologies, iii) explaining how to evaluate an ontology, and iv) how to develop a well-designed ontology.

Serrano and Gyrard (2016) provides a validation toolkit, *Hyperthing*, *Neon*, OWL validator, *OQuare*, *OntoClean*, *OnToology*, *vapor*, *OOPS!*, W3C RDF validator, *jena eyeball*, *ontoCheck*, *OntoAPI*, *ontoMetric*, *Prefix*. However, we did not find a benchmark or guide that guides the assessment, a suitable method for conducting the necessary validations that fits the needs of the project.

Gyrard, Datta and Bonnet (2018) say we must validate an ontology through the criteria defined below, and use some applications to satisfy the criteria. The criteria are as follows:

i. *Serialization*: The W3C Web Ontology Language (OWL) is a Semantic Web language designed to represent rich and complex knowledge about things, groups of things, and relationships between things;
ii. *Syntactic validation*: Activity required for compiling and executing ontology with libraries. We use the OWL Manchester tool because it is user-friendly to use on *Protégé*, particularly for entering and displaying descriptions associated with classes;
iii. *Interlinking*: Allows to integrate and navigate between ontology concepts and roles, ensuring interoperability;
iv. *Documentation*: The Parrot Tool for Automatic Ontology Documentation (hosted on Mondeca) was unavailable for tests.
v. *Improve Ontology Design*: The tool of choice was OOPS for detecting many early ontology pitfalls appearing within ontology developments. OOPS! warns us when: The domain or relationship is defined as the intersection of two or more classes. This warning avoided reasoning problems in case those classes could not share instances. A cycle between two classes in the hierarchy was included in the ontology. Detecting this situation avoided modeling and reasoning problems.

We chose tools based on some criteria: reusable; easy-to-integrate web services; open-source; extensive documentation or tutorials; and reactive community answering questions. At times, we find a server hosting for out-of-service web services or changes in tool updates. Something that happened during these assessments, servers were out of service on July 2017, November 2018, and recently, during last month, evaluation moments of this project with OOPS! and OWL Manchester.

During the process of dentistry development, some evaluation tools and reasoning tools were identified and tested; however, the ontology should be verified in both syntactic and semantic forms. Table 1 shows the tools used and a brief description of each. We use the *Protégé* and *Pellet Reasoner* tools tests, general activities, error checking, as they are widely cited and accepted by the semantic web community.

Sandbox (test case)

Jain (1990) state that for each performance study, we chosen a set of performance criteria based on commonly used performance analysis. To reach a quantitative performance analysis, we performed 15 interaction in each of the three test cases on our computer network offers the service of forwarding packets to the specified destinations on heterogeneous systems. These performance analysis may forward the packet correctly, it may forward wrong destination, or it may be down. The evaluation planning, performing, and analysis are described as follows.

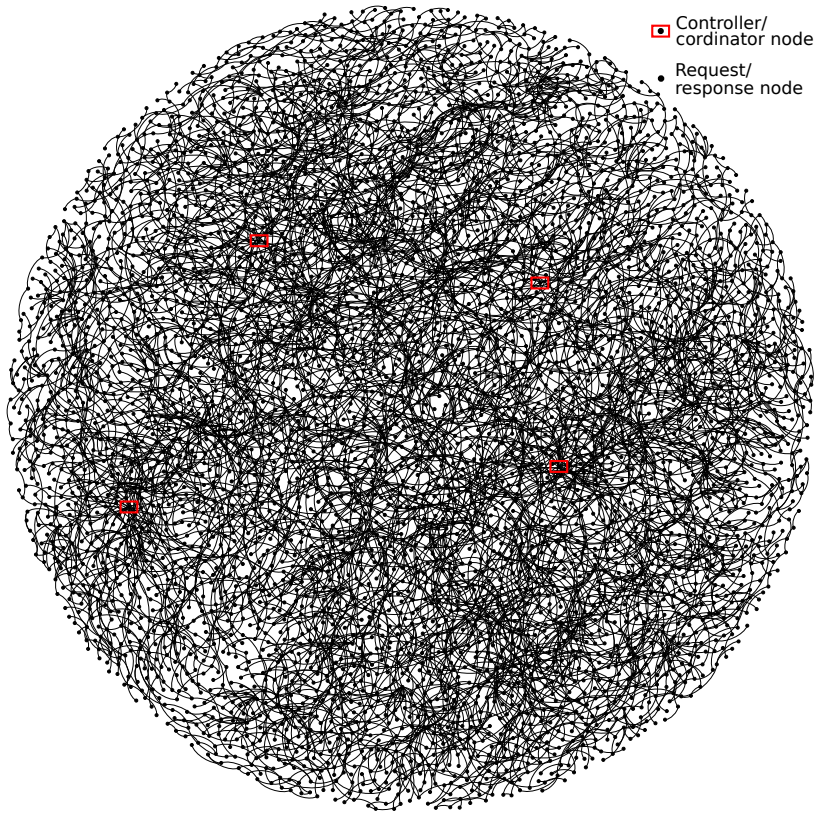
Selecting performance metrics

By performing the task correctly, we measure its performance by the time elapsed to accomplish the task, the rate at which the task is performed, and the resources consumed to accomplish the tasks. Three related metrics can be used.

- *responsiveness* (speed): is measured by its response time—the time interval between arrival of a packet and its successful delivery;
- *productivity* (reliability): is measured by its throughput—the number of packets forwarded per unit of time.
- *utilization* (availability): Performance optimizations at this resource offer the highest payoff—finding the utilization of various resources inside the system is thus an important part of performance evaluation.

The testbed consists of SDN and OpenFlow technologies. For the construction of our simulation infrastructure, we used a cluster of 4 AMD A8-5500 3.10GHz with 16GB RAM servers, we constructed a simulated blockchain network with four controllers and 4000 requests/responses nodes, as shown in Figure 18. The Docker platform was used to deploy the testbed by performing the following phases:

Figure 18 – Blockchain network structure with 4 controllers/coordinator and 4000 request/response nodes.



Source: The Author (2018).

- The test cases were compiled as a Docker image to create all the scenarios;
- Each node was created as a container instance of the compiled image;
- The Docker API SDK was used to manage the experiments.

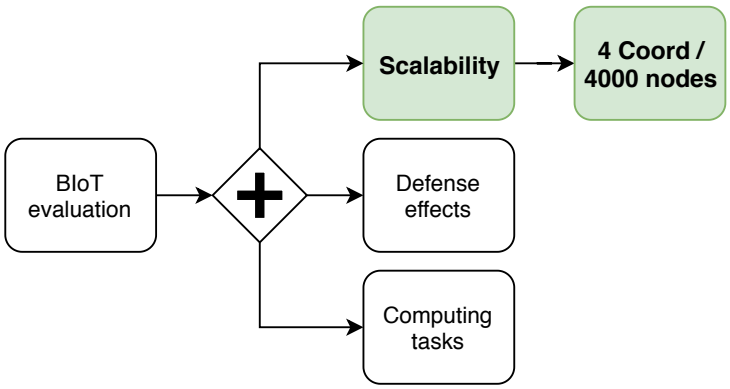
Scalability evaluation

We use OpenFlow software switch, we also use Mikrotik, with MikroFlow application, which is an OpenFlow based application, able to establish flow rule tables for our tests. To compare the performance of the flow rules table update scheme of our proposed BIoT in a large-scale network, we also built an SDN. We also are have compared against the DistBlockNet performance (see Figure 19).

Although a software-based flow rule table is not able to achieve a similar level of performance, we still noticed that BIoT conserves resources and provides significant protection.

We also noticed some limitations for sending encrypted data via TCP protocol, limitation of MikroTik device with OpenFlow protocol. In this scenario, we chose to evaluate packets using the UDP protocol.

Figure 19 – BIoT scalability evaluation overview.



Source: The Author (2018).

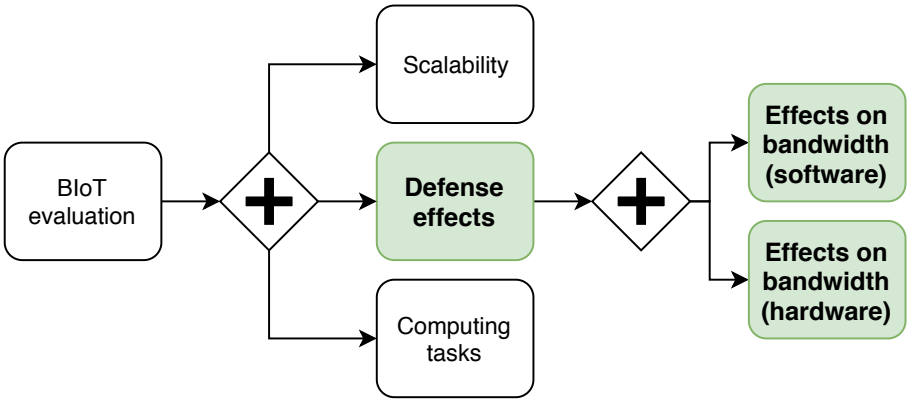
Defense effects

To evaluate the defense effects of our BIoT, we evaluated and compared it with an OpenFlow SDN MikroTik, called MikroFlow, and software environment Mininet SDN emulation tool. We implement clients and data plane caches in the MikroTik router with MikroFlow SDN. We assign the task to some clients of firing packets with incorrect access keys, as well as dispatch a UDP floating attack and 51% attack attempts on the BIoT network (see Figure 20).

Computing tasks

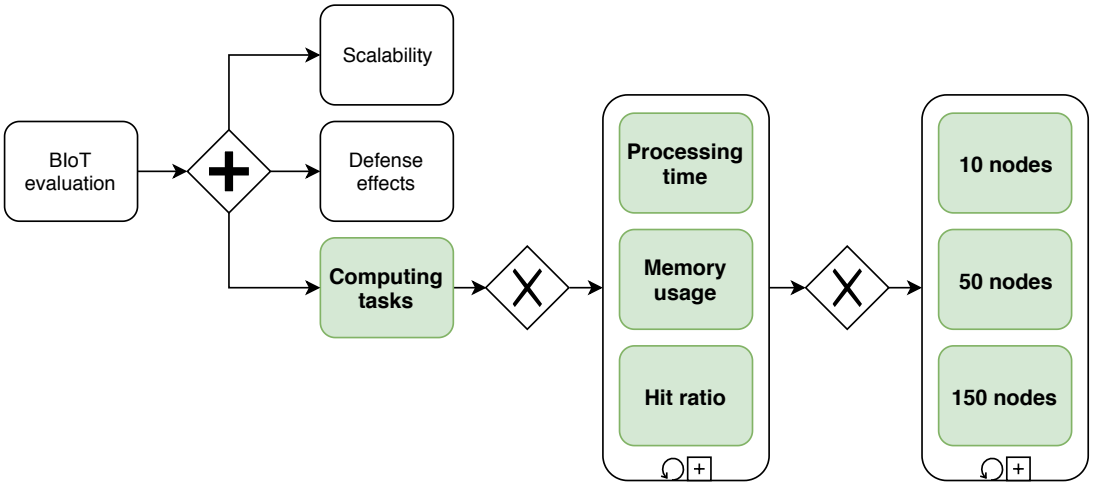
We established and evaluated some parameters based on the work of Ruta *et al.* (2017). This settings are to get a quantitative performance analysis. It was essential for further interpreta-

Figure 20 – BIoT evaluation defense effects overview.



Source: The Author (2018).

Figure 21 – BIoT evaluation computing tasks on processing time, memory and hit ratio with 10, 50, and 150 nodes overview.



Source: The Author (2018).

tion, possible comparability, and the evaluation of the obtained results.

We established and evaluated some parameters based on the work of Ruta *et al.* (2017). This settings are to get a quantitative performance analysis. It was essential for further interpretation, possible comparability, and the evaluation of the obtained results.

We built a small, medium, and large scale scenarios, with 10, 50 and, 150 nodes. In each scenario, we split two sets: the nodes into Coordinators, providers of annotated resources and semantic-based monitoring; and common nodes, resource requests that perform established methods to perform semantic-based agent tasks; all registered in the blockchain network (see Figure 21).

The following parameters were set:

i. processing time for tasks (duration was 300s);

ii. the Coordinator/common node ratio was 0.1;
iii. each Coordinator registered 20 generated annotations;
iv. each common node sent a new randomly-generated request every 10s;
v. each request could be forwarded to a subset of four nodes;
vi. a request was aborted if no match was found after the second hop;
vii. minimum threshold of semantic affinity was 0.9.

The following are the statistics collected:

i. average computing time processing tasks;
ii. average turnaround time for performing a request;
iii. average virtual machine memory usage per node;
iv. average hit ratio per node.

Memory

Usage of average and maximum RAM showed a decreasing trend for an increasing number of nodes among scenarios.

Hit ratio

We defined Hit ratio as the percentage of requests which retrieve on resource satisfying both node status and semantic relevance constraints within the given timeout

Survey

A survey is a data collection and measurement process that has the following characteristics (JR; FOWLER, 1995):

- the purpose of the survey is to produce statistics, that is, quantitative or numerical descriptions about some aspects of the study population.
- the main way of collecting information is by asking people questions; their answers constitute the data to be analyzed.
- generally, information is collected about only a fraction of the population, that is, a sample, rather than from every member of the population.

Hakim (1987) affirms that small examples can be used to develop, test, and even explain a particular problem, especially at the beginning of the research. In this sense, Beecham *et al.* (2005) states that the studies use small examples to obtain feedback from experts to evaluate the development of models that support the area of knowledge. Such as Dyba (2000) used 11 experts to conduct a review process on critical success factors in a software improvement process, based on data collection from 120 organizations. Emam and Madhavji (1996) interviewed 30 experts to develop a tool to evaluate success in requirements engineering.

There are three kinds of questions we can use in a survey: open, closed, and mixed questions. The open question allows the respondent to answer with its words, enabling the freedom of expression (AMARO; PÓVOA; MACEDO, 2005).

The value of a specialist's view or knowledge is also recognized in software quality assessment that suggests methods to capture expert opinion Rosqvist, Koskela and Harju (2003) formally. So we can state that the Blockchain and Internet of Things community have given more importance and credibility to studies that use techniques and expert opinions.

Other researchers have evidenced the relevance of this technique, as the researcher has done a precision analysis of many methods of estimating effort using expert opinion. This research revealed that through statistical analysis, a process of human estimation centered on the opinion of experts could overcome substantially simple models of function point analysis. The work of Minoli and Occhiogrosso (2018) also presented in its research with experts the evaluation of a maturity for Blockchain mechanisms for IoT security based in the context of a defense-in-depth approach.

Thus, researchers and practitioners adopt the expert opinion technique to evaluate their proposals, to obtain valuable feedback and practical evaluation.

Opinion of experts

The opinion of experts can be defined as a series of efforts widely used to interpret data, predict the behavior of a system, and may assess uncertainties (COOKE *et al.*, 1991). These expert opinions in one area of knowledge, considered as speculations, assumptions, and estimates, may support the acquisition of knowledge in any decision-making process (LI; SMIDTS, 2003; COOKE *et al.*, 1991).

The increased demand for expert opinion, especially in academic research, is justified by the fact that in many areas of knowledge, there are still immature decision-making processes under construction.

The research of Li and Smidts (2003) presented well-defined steps, which inspired the execution of our process to gather expert opinion, which consists of the following steps:

- *Problem state*. Prior knowledge and problem must be systematized and defined;
- *Selecting experts*. Several experts should be defined based on a set of criteria that may include the credibility, knowledge, skill, and reliability of experts.
- *Opinion elicitation*. This stage poses the right question and guarantees the driving conditions for an elicitation process;
- *Opinion aggregation*. The idea is to arrive at an aggregate opinion or consensus-based on which a decision can be made;
- *Decision making*. This last step makes the decision based on aggregate opinion.

It is worth to highlight that software engineering has used in its researches the opinion of the experts. However, there are some controversies (KITCHENHAM; CHARTERS, 2007) and skepticism in the scientific community about this matter, and until the present moment, there is no consensus. (KITCHENHAM; CHARTERS, 2007) affirms that the main problem is the dependence of informal proofs that might be influenced by the opinion of the experts. That is, depending on the specialist, its bias (cultural, origin, or self-confidence) can distort the information he passes.

Garcia (2010) says that if a specialist is perfect (that is, he has infinite knowledge about the theme and never makes mistakes), only one specialist is needed to the elicitation process. However, there is a tendency to follow up with as many specialists as possible, justified by a perception of security in the numbers (quantity of specialists). In this sense, Li and Smidts (2003) say that the objective of the opinion of experts in the acquisition of knowledge of the real world, the capture of the specialist's experience (challenges, lessons learned).

Finally, according to Li and Smidts (2003), the number of specialists needed in a study, the identification of the bias, and the technique adopted to the aggregation of the opinion of the experts are questions which must be planned and clarified at the beginning of the study. This way, the adoption of the evaluation of the research-based in the opinion of specialists will be more effective. In the next sections, these questions will be described in details.

In an ideal scenario, the specialists must be carefully chosen, taking into account some factors, like his knowledge in the research area according to NUREG-1150 by Hora and Iman (1989 apud GARCIA, 2010, p. 78). However, there are not a well-defined and known standard (LI; SMIDTS, 2003), for this selection or choice of specialists. In light of the above is relevant to formulate a criteria set which must be used to systematize the process of selection. NUREG-1150 presents a set of guidelines for this selection:

- The specialists must have experience proved by publications or services of consulting or management in the areas related to the theme of the study;
- Every specialist must be sufficiently versatile to be able to deal with questions concerning the studied theme, and wide experience to know how they will be put in practice;
- The specialists must represent a variety of experiences (e.g., academic knowledge, consulting, laboratories);
- The specialists must be willing to participate in the research and available to pass the requested information according to the method of data gathering to be used.

It should be pointed out that the specialists also are subject to some biases, particularly when forced to opine about subjects outside their knowledge domain (SLOVIC, 1987). For this reason, the specialists should be consulted only about events relative to their area of specialty; the experience and information relevant which contributed to their evaluation should be additionally required since even the specialists have a piece of wide knowledge, they might have difficulties to attribute probabilities (SKJONG; WENTWORTH, 2001).

Tversky and Kahneman (1974) divided the experts into two classes: those with biases originated from the local where they live or work and those with biases from the excess of confidence. An additional method to decrease the biases from the excess of confidence is to encourage the specialists to find reasons that contradict their initial opinions. However, the biases from the local or work may be corrected by the method of Bayesian aggregation (CHHIBBER; APOSTOLAKIS; OKRENT, 1992; LI; SMIDTS, 2003).

Li and Smidts (2003) say that when the aggregation methods had been used, they vary from easy-to-use methods, like the simple calculation of the arithmetic average of the specialists, to more complex techniques (CHIDAMBER; KEMERER, 1994) and the Bayesian aggregation (CHHIBBER; APOSTOLAKIS; OKRENT, 1992). Still about the aggregation of the opinion of the experts, (KEENEY; MCDANIELS, 1992) defined a process called Value-Focused Thinking (VFT) on which is searched the identification of the values and knowledge that the researcher will use as a guide to the process. The VFT approach is a way to identify desirable decision situations and so collect the benefits of these situations to solve them.

Number of experts

To this thesis, 52 potential candidates were selected, composing the group of specialists for this research. They represent a group of specialists with the capacity to evaluate and contribute to the improvement of the BIoT. Despite the reduced sample, 52 specialists are considered an adequate number. As notorious in the literature, the inquired sample is satisfactory to the current study, once it is qualified by renowned specialists, with an extensive experience in the Internet of Things, Embedded Systems and security process. In practice, from the selected sample, only 19 specialists participated in this study. The next session will describe the process of their selection.

Expert selection

The main requirements to participate in the study were: i) have a minimum of three years of experience (theoretical or practical) in the Internet of Things, Embedded Systems; ii) know Blockchain mechanism or technologies. Therefore, according to the recommendations of the NUREG-1150 (1989 apud Garcia, 2010, p 78), were selected specialists of the industry and academy, from different companies and universities, as well as from different places.

Expert biases

In this thesis, we defined the process for the elicitation of the opinion of specialists, such that the specialists provided clear and detailed explanations about the evaluation of the BIoT. Therefore, they also were contacted to clarifications about the evaluation. So, we had no evidence of bias or prejudices about participation in the research.

Experts opinion aggregation

In this doctorate research, we adopted the aggregation method based on the arithmetic average, based on the answers (analysis of the data) of the specialists. It is essential to highlight that no bias was perceived during the investigation. Furthermore, all the specialists were equally weighted during the aggregation; once there was not observed a significant difference in terms of their credibility and importance.

Research approach

The approach to this study was based on the recommendations from Li and Smidts (2003) and motivated by the work of Garcia (2010). It is organized into four phases. In the first phase, the objective was defined, evaluate, and validate the script of the guided interview. In the second phase, the specialists were selected and contracted according to the guidelines discussed in the Section 3.4.1. In the third phase, the invitation to participate in the interview was sent to the specialists, and after the acceptance, the data was collected. Finally, in the fourth phase, the data were analyzed aiming to characterize the BIoT in what concerns to its viability, based on the opinion of experts. The next sections discuss every phase in details.

In this work, we develop and apply open and closed questions based on literature reference with our adaptations. We have validated it by experts (see entire results in chapter 5). We have wanted to detect the adherence level this proposal in the real project through the expert opinion.

The Survey

The research was composed by an interview script developed after a wide literature review, with a strong influence of works related to the Internet of Things, Embedded Systems and Security area (LUU *et al.*, 2016) (LUU *et al.*, 2015), (VIGNA; CASEY, 2015), (BIRYUKOV; KHOVRATOVICH; PUSTOGAROV, 2014), (ZHANG; WEN, 2017), (ZHANG; WEN, 2015), (HARDJONO; SMITH, 2016), (ATZORI, 2017), (CONOSCENTI; VETRO; MARTIN, 2016), (CHRISTIDIS; DEVETSIKIOTIS, 2016), (HASHEMI *et al.*, 2016), (HUCKLE *et al.*, 2016), (UCKELMANN; HARRISON; MICHAHELLES, 2011), (PANIKKAR *et al.*, 2015), (NORTA, 2015), (ZYSKIND; NATHAN; PENTLAND, 2015), (ZYSKIND; NATHAN *et al.*, 2015) among others. Furthermore, we counted with the experience of some members of our research group.

The first version of the research was defined at the beginning of 2016 and was reviewed for four months. The review was performed together with three researchers (one with theoretical and practical experience in the Internet of Things, and the other two experienced in the embedded systems area and security).

During this review process, four versions of the research were generated. The main improvement about the first version was related to the order of the questions, the decrease of the script size, as well as adjusts in some clarifying points in the questions.

The Questions

The main aim of our study was to evaluate the viability of the BIoT, as well as its adaptation based on the opinion of specialists. In this sense, the questions were elaborated about the role of the specialist (academy or industry), the way that the maturity factors are distributed in the levels; the specification, description of the main objective of every maturity level; objectives related to the maturity factors; descriptions and objectives of every practice and, if it is possible to an organization to perceive and obtain in the incipient levels the benefits of the effective use of API, outside the highest levels proposed by the BIoT.

The final script was composed of 40 questions with open and closed questions and was projected to be concluded in approximately one hour, together with an expert researcher (see Appendix A).

Data collection and analysis

Before the interview, the script was sent by email to all the specialists in September 2018, and next to the interviews were appointed (some face-to-face, others through Hangouts). In November 2018, all the interviews were finished and collected to analysis. In some cases, we needed to contact the respondent to mitigate doubts or clarify some answers that could lead to diverse interpretations. In the study, four specialists were contacted to clarify some questions, mainly related to the objectives and practices of the BIoT.

3.5 STEP 4 — BODY OF KNOWLEDGE

A Body of Knowledge is the complete set of concepts, terms, and activities that make up a professional domain. We have developed a structured knowledge that is used by Blockchain-based the Internet of Things Security community to guide its practice. We have built a set of knowledge through the following: i) the first systematic mapping to identify factors and standards used in Blockchain-based the Internet of Things Security building, making it possible to run smart contracts between devices; ii) We have conducted an interview and a focus group to get new features and expert evaluation. Step 4 corresponds to building the “Body of Knowledge” (DIAS-NETO; SPINOLA; TRAVASSOS, 2010).

The Blockchain-based ontology for the Internet of Things Security defines a common vocabulary for researchers who need to share information in this domain, and it will include machine-interpretable definitions of basic concepts in this domain and relations among them (NOY; MCGUINNESS, 2001). In practical terms, we will develop an ontology that includes:

- defining classes in Blockchain-based ontology for the Internet of Things Security;
- defining slots and their allowed values;
- filling in the values for slots for instances.

Noy and McGuinness (2001) also say we can then create a knowledge base by defining individual instances of these classes filling in specific slot value information and additional slot restrictions.

3.6 BLOCKCHAIN-BASED THE INTERNET OF THINGS: A MAPPING

The systematic mapping was used to analyze of the previous findings, techniques, ideas, and ways to explore the topics in question, their relevance to the issues of interest and synthesis and summarization of this information. Also, the snowball sampling technique was used to recruit experts in the research area in this thesis.

The systematic mapping study (see a satellite view in Figure 22, based on systematic mapping study by Kitchenham and Charters (2007)) is indicated during a first incursion into the topic discussed, before starting a systematic review, we came across a comprehensive question. To obtain an overview of this research topic and identify evidence to provide the best positions on the issues of research, we have established a systematic mapping. (KITCHENHAM; CHARTERS, 2007).

Kitchenham and Charters (2007, p. 5) still state that the systematic mapping allows:

- mapping the evidence of a domain at a high level of granularity;
- the identification of clusters and voids of evidence to enable future systematic reviews;
- discover areas to conduct new primary studies.

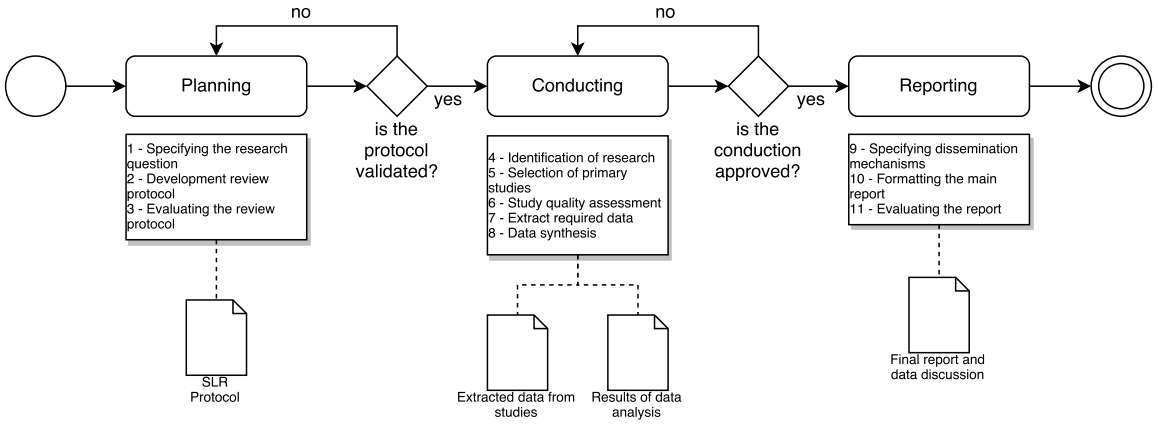
This study aims at moving towards a consolidated knowledge about Internet of Things, Semantic Web, and Embedded Systems areas by developing a better understanding of which factors influence in IoT-based Embedded Systems projects. To achieve the goal of this study, we are conducting a systematic mapping of critical factors in IoT paradigms-based embedded systems building.

To achieve the goal of our SLM, we formulated the following research questions (RQ).

RQ_1 Has Blockchain-based IoT been constructed to stand on development processes? Which are those processes? This RQ aims at identifying the processes, but including other initiatives (e.g., models, frameworks, architectures, methods, approaches, designs and procedures) proposed for building Blockchain-based IoT.

RQ_2 Which Blockchain-based IoT's characteristics, principles, or requirements have been considered in Blockchain-based IoT development processes? This RQ aims to investigate the main characteristics mentioned by the authors following as basis ten Uckelmann, Harrison and Michahelles' key requirements that need to be considered in the Internet of

Figure 22 – Systematic Literature Mapping.



Legend: Figure is presenting an overview of the Systematic Literature Mapping: planning, conducting and, reporting.

Source: Adapted from Kitchenham and Charters (2007).

Things (UCKELMANN; HARRISON; MICHAHELLES, 2011).

3.6.1 Protocol

We conducted this study based on Kitchenham and Charters (2007)'s conscious guidelines and procedures. This protocol specifies the basis for the study research questions, search strategy, selection criteria, and data extraction and synthesis. The protocol was mainly developed by one of the researchers and reviewed by two of the senior researchers aiming to mitigate any bias.

Search string. The standard version of search string was designed to include variations and synonyms terms related to "Internet of Things", "Blockchain" and their "Development Processes" (see Listing 3.1).

Listing 3.1 – Search string

```
1 (((model OR framework OR architecture OR process OR method OR
  ↪ approach OR design OR procedure) AND (development)) AND
  ↪ ((internet of things OR iot OR internet of everything OR
  ↪ web of things OR smarter planet))) AND (blockchain)
```

Search strategy. We selected the following search engines: ACM Digital Library[1], IEEE

[1] http://dl.acm.org/

Xplorer[2], ISI Web of Science[3], Science Direct[4], Scopus[5], Engineering Village[6]. It was considered the experts opinion, gray literature, and related works of the included studies.

Inclusion and exclusion criteria. The studies were selected according to the inclusion and exclusion criteria described below. To accordingly select the studies to answer our research questions, we established the following Inclusion (IC) and Exclusion Criteria (EC):

IC_1 The study discusses Blockchain-based IoT development processes.
IC_2 The study addresses Blockchain-based IoT characteristics, requirements, problems, or activities related to Blockchain-based IoT development processes.

EC_1 The study is not related to Blockchain-based IoT.
EC_2 The study does not discuss any Blockchain-based IoT development process.
EC_3 The full study is not available.

3.6.2 Conduction

Once the protocol has been agreed, the review proper can start. However, as noted previously, researchers should expect to try out each of the steps described in this section when they construct their research protocol (KITCHENHAM; CHARTERS, 2007).

Meline (2006) recommends the adoption of effective criteria for inclusion and exclusion of relevant studies to answer the research questions to be required. Some of the criteria are essential for the collection of a rigorous and defensible set of data for evaluation.

Therefore, we will apply the inclusion and exclusion criteria involved in the analysis of the parameters **1)** title and keywords of each article, and check which were available as full paper, it will be applied one **2)** summary of the analysis in the works identified in the previous phase, if there is a question, reading the introduction and conclusion; and **3)** the complete reading of the paper. Table 3 presents the primary studies included in this paper.

Table 4 shows the selection of studies by database (source studies). The initial search resulted in 25 works. In the first analysis, we excluded items 2, 23 remaining papers. Second selection, applying the criteria of inclusion and exclusion in the reading of the summary, the number of articles was reduced to 21. Upon complete reading from each of the other items, it was found that there two papers that had duplicate or same or similar content, resulting in their exclusion, leaving at the end a total of 17 papers with strong and relevance indications to the investigated area.

3.6.3 Reporting

Based on the analysis of the 17 primary studies included, we address the RQs.

RQ$_1$ *Has Blockchain-based IoT been constructed to stand on development processes?*

We identified 17 initiatives of Blockchain-based the IoT development as shown in the Table 4. (Three) of these initiatives are classified by the authors as frameworks, (4) as models, (6) as approaches, and (4) as other initiatives. For the other studies, we created the classification *Other Initiatives to Blockchain-based the IoT Development* that includes single initiative of methodology, description, [re]engineering, ontology or simulation platform to Blockchain-based the Internet of Things.

There are some methodologies to construct Ontologies, we recognizing the importance of formal methodologies to business adoption and success of the technical proposal. These methods are diverse in their scope, focus and approach, e.g., MaSE (DELOACH; KUMAR, 2005), GAIA (ZAMBONELLI; JENNINGS; WOOLDRIDGE, 2005), PROMETHEUS (PADGHAM; WINIKOFF, 2005), TROPOS (BRESCIANI *et al.*, 2004), MADEM (GIRARDI; LINDOSO, 2005), MOBMAS (TRAN; LOW, 2008). we have elected the MADEM and MOBMAS approaches because these are presented as well-known methodologies focused on the modeling phases and tasks.

Table 6 presents an overview of the initiatives regarding the domain type (i.e., general, specific or non-specify), modeling phases (domain analysis, domain design) and products (domain model, architectural model and, agent models), described in MADEM Methodology (GIRARDI; LINDOSO, 2005), summarized in Table 5.

RQ$_2$ *Which Blockchain-based IoT's characteristics, principles or requirements have been considered in Blockchain-based IoT development processes?*

We reported some frameworks, models, approaches, and other Blockchain-based the Internet of Things initiatives that present adherence to well-know development processes address to build an initial knowledge body. We detect key requirements in the Internet of Things and, we typed theirs as functionals and non-functionals requirements. Authors of the primary studies classified their works: (four) as frameworks, (4) as models, (2) as methods, (3) as approaches and, we classified (4) papers as other initiatives addressed in initial descriptions or superficial studies.

We have evaluated the adherence of each included study about the essential characteristics described in the UCKELMANN; HARRISON; MICHAHELLES' IoT key requirements and the GIRARDI; LINDOSO's MADEM methodology. In this analysis, we evaluate each initiative against the following 7 processes:

[2] http://ieeexplore.org/
[3] http://webofscience.com/
[4] http://www.sciencedirect.com/
[5] http://www.scopus.com/
[6] http://www.engineeringvillage.com/

- Modeling Phases:
  - Domain Analysis;
  - Domain Design:
    - Architectural Design;
    - Detailed Design;
- Products:
  - Domain Model;
  - Architectural Model;
  - Agent Models.

The following we have presented the Uckelmann, Harrison and Michahelles' key requirements (kR) that need to be considered in the Internet of Things.

We have identified that the requirements kR_1, kR_2, kR_3, kR_4 and kR_{10} are related to the Back-end Internet of Things Core Architecture oriented to IoT, it is the Scope of this research. While the requirements kR_5, kR_6, kR_7, kR_8, kR_9 were excluded from the scope because they point to a social context of IoT application.

We summarized the domain type: (six) as generic, (8) as specific and, (3) as non-specify. We aim to identify essential characteristics, processes, modeling phases, tasks, and products. We evaluated theses works to adherence to MADEM methodology (knowledge modeling). Most (16) of these works emphasized the domain analysis, but just (7) theses presented domain design, into (3) on architecture design and, (2) on presented detailed design. These papers addressed the product modeling: (ten) as domain model, (5) as an architectural model and, (1) as an agent model. The community still has no better support to the design, architecture, integration, and testing processes to build Blockchain-based the Internet of Things (see Table 6).

Considering UCKELMANN; HARRISON; MICHAHELLES' IoT key requirements, Table 7 depicts 100% of all studies addressed the kR_1, meet key societal needs for the Internet of Things including open governance, security, privacy and trustworthiness; followed by 70.6% both kR_2, bridge the gap between B2B, business-to-consumer (B2C) and machine-to-machine (M2M) requirements through a generic and open Internet of Things infrastructure; and kR_3, design an open, scalable, flexible and sustainable infrastructure for the Internet of Things; kR_4 develop migration paths for disruptive technological developments to the Internet of Things is covered by 64.7%; kR_5 excite and enable businesses and people to contribute to the Internet of Things is covered by 58.8%; followed by 52.9% both kR_6 Enable businesses across different industries to develop high added-value products and services and, kR_8 provide an open solution for sharing costs, benefits and revenue generation in the Internet of Things. The other IoT key requirements did not achieve at least 50%.

It has also been evaluated the adherence of each paper. In this analysis, the papers are evaluated against the 10 IoT key requirements. We highlight the importance of the studies S14, S15, S16, S8, and S1. They discuss some main activities or artifacts or modeling of design,

but they did not explicitly address theses activities or artifacts or modeling design in their propositions. However, they mention all the essential IoT characteristics processes. On the other hand, we considered that studies, S2, S5, S6, S11, S4, S7, S3, S9, S10, S17, S12 and S13 covered by 50% or less this did not explicitly address all the IoT fundamental processes.

3.6.4 Current trends and challenges

The main trends and challenges discussed by the authors of the included papers about Blockchain-based the Internet of Things are described in this section.

Atzori (2017) says that security flaws in the Internet of Things may lead, for instance, to malicious attacks on secrecy and authentication, silent attacks on service integrity, or attacks on network availability, such as the denial of service (DoS). Privacy and anonymity, on the other hand, are no less severe issues. The IoT objects are natural "collectors and distributors of information", so they represent a unique challenge to individual privacy.

In particular, the ubiquitous interaction of users with smart objects and groups of things; the invisible and automated collection of fine-grained data by third parties; and the uncontrolled concentration of such data on platforms lacking in transparency may systematically expose users to several threats, such as: identification, localization, monitoring, tracking, surveillance, manipulation, profiling, targeted advertising, data linkage, and even social engineering.

Conoscenti, Vetro and Martin (2016) investigated, which are the main factors that affect the levels of integrity, anonymity, and adaptability of the blockchain. They should further analyze what the security properties provided by the Proof of Work, which up to now is one of the key factors are allowing to achieve distributed consensus.

Ethereum platform supports the feature to encode rules or scripts for processing transactions through of smart contracts. Luu *et al.* (2016) investigate the security of running smart contracts based on Ethereum in an open distributed network. According to Luu *et al.* (2016), Luu *et al.* (2015), there are several new security problems. These bugs suggest subtle gaps in the understanding of the distributed semantics of the underlying platform. Authors propose ways to enhance the operational semantics to make contracts less vulnerable through a symbolic execution tool called Oyente. according to LUU *et al.*,

Blockchain has recently attracted the interest of stakeholders across a wide span of several industries, from finance and healthcare to utilities, real estate, and the government sector. That explosion of interest on the Blockchain-based applications has happened because we need applications that could previously run only through a trusted intermediary. Moreover, with adoption on Blockchain strategies, we can operate without the need for a central authority (CHRISTIDIS; DEVETSIKIOTIS, 2016).

A Blockchain-based the Internet of Things initial ontology will define a common vocabulary for researchers who need to share information in this domain, and it will include

machine-interpretable definitions of basic concepts in this domain and relations among them (NOY; MCGUINNESS, 2001). In practical terms, we will develop an ontology that includes:

- defining classes in Blockchain-based the Internet of Things initial ontology;
- arranging their classes in subclass and superclass (taxonomic hierarchy);
- defining slots and their allowed values;
- filling in the values for slots for instances.

Noy and McGuinness (2001) also say we can then create a knowledge base by defining individual instances of these classes filling in specific slot value information and additional slot restrictions.

3.6.5 Threats to validity

We have detected some threats to validity in this Systematic Literature Mapping:

- the specific group of interesting;
- the choice of search engines;
- choice of primary studies;
- placebo effects or courtesy bias or inadequate survey instrument;
- number of reviewers;
- study no available;
- data extraction doubts.

3.7 RELATED WORK

Pilkington (2016) proposes principles and applications to Blockchain technology, the core concepts and definition of the blockchain towards hybrid solutions, and the potential risks and drawbacks of public distributed ledgers.

Among the several critical concerns related to IoT, there are two concerns which deserve attention: *security* and *privacy*. Banerjee, Lee and Choo (2018) say that “in data-sensitive applications such as the Internet of Battlefield-Things (IoBT) and Internetofvehicles (IoV), ensuring the security of the data systems, and the devices, as well as the privacy of the data and data computations, is crucial”.

However, a threat to a system can be the result of a security measure that is not well-thought-out. In this way, considering the importance of maintaining security during the IoT applications, this research selected articles published since January 2016 about security techniques that are either designed for or apply to IoT. This study reported the following points:

1. there is a need to develop a standard for sharing IoT datasets among the research and practitioner communities and other relevant stakeholders;

2. the blockchain technology shows a potential to facilitate the secure sharing (confidentiality, integrity, availability, and immutability) of IoT datasets and the security of IoT systems.

> Thus, the authors highlighted the potential of blockchain in sharing and distributing such datasets in a research network; presented a conceptual blockchain-based compromised firmware detection and self-healing approach that can be deployed in an IoT environment; the use of blockchain as a collaborative security foundation to secure other IoT and related systems; suggested the optimization of blockchains and blockchain-based platforms to reduce energy consumption while offer more effective and efficient services; and proposed the necessity to designer an efficient and lightweight blockchain-based IoT security solutions with the aims to monitor the emerging threat landscape. (BANERJEE; LEE; CHOO, 2018).

Uckelmann, Harrison and Michahelles (2011) present in their book a list of 10 Key Requirements we need considering for Architecting the Internet of Things. They still say that these key requirements are not intended to provide a complete set of requirements and, they are meant to focus on specific aspects of the Internet of Things to start a rethinking process for future developments.

Weber (2010) says that the Internet-based technical architecture facilitating the exchange of goods and services in global supply chain networks has an impact on the security and privacy of the involved stakeholders. Measures ensuring the architecture resilience to attacks, data authentication, access control, and client privacy need to be established. Since business processes are concerned, a high degree of reliability is needed. In the literature, the following security and privacy requirements are described as *Resilience to attacks*: The system has to avoid single points of failure and should adjust itself to node failures. *Data authentication*: As a principle, retrieved address and object information must be authenticated. *Access control*: Information providers must be able to implement access control on the data provided. *Client privacy*: Measures need to be taken that only the information provider can infer from observing the use of the lookup system related to a specific customer; at least, the inference should be hard to conduct.

Atzori (2015) explores the main features of three blockchain-based platforms for the Internet of Things, Enigma, IOTA-Tangle, and ADEPT, as recently emerged in academia as well as in industry. The author says that if properly engineered, the blockchain technology offers a disruptive solution to the problem of security and privacy on Internet of Things environment, providing a new computational layer where data can be safely processed and analyzed, remaining private. The blockchain can also enable micro-payment functionality between digitally-enhanced devices, through ultra-light cryptocurrencies and smart contracts.

Another critical point is to explore practical applications of the blockchain, cryptocurrencies, and smart contracts in the Internet of Things context (ZYSKIND; NATHAN *et al.*, 2015). In the works of Kotis and Katasonov (2012) there is concern in identifying the challenges present in the Internet of Things, the possibility of a global connectivity between the real world and the virtual world of the entities subject-object, embedded devices, actuators, handles, software

applications-services, connecting things, not just places or people and “real-time delivery machine data-information for users”. Kotis and Katasonov (2012) also propose an ontology for automated deployment of applications in heterogeneous environments of the Internet of Things.

Agrawal and Vieira (2013), Xu, He and Li (2014) present the results of several questions, such as integration issues and communications technology, distributed intelligence, in addition to addressing the paradigm of the Internet of Things. Xu, He and Li (2014) also point out that the Internet of Things offers promising opportunities for the construction of systems and applications in the industry – briefly, the key concepts and a brief state of the art Internet of Things in the industry. In studies of Maksimovic, Vujovic and Omanovic-Miklicanin (2015) present a low-cost solution for traceability based on the Internet of Things paradigm and monitoring of food safety during transport.

The implementation of such features is expected to ensure a more efficient allocation of resources at the global level. However, it may also lead to undesirable consequences – such as a hyper-tokenization of society and a potentially dystopian concentration of power on big global platforms. Therefore, the overall benefits and drawbacks of the blockchain the deployment must necessarily take account of specific contexts of use, finding a balance between a need for innovation and social sustainability. (ATZORI, 2015).

There are numerous studies about security and privacy issues of blockchain networks, basically to the heterogeneous interconnected devices. In Moin *et al.* (2019) these issues are addressed. It is essential to consider some limitations of IoT, in particular, the limited storage and processed capacity of IoT devices. Thus, there is a need for separate data storage arises so that data can be utilized in the future; being these storage services provided by third-party at the cost of a user’s privacy. Moreover, the centralized database needs to store is more susceptible to attack due to its single point security breach chances. Furthermore, present IoT data is not trustworthy when applied in an external environment, as data manipulation is lacking when data is shared with other parties.

In this context, the Blockchain has been suggested to overcome the limitations mentioned above of IoT, being considered an emerging secure decentralized storage technology. Blockchain enables an enhancement of security and incorporates a large number of devices in today’s ecosystems. From Table 8 we highlight the works related to the largest, the works with the most significant interest and primary resources to this thesis, below.

Sharma *et al.* (2017) say that the DistBlockNet is an IoT platform that has the flexibility, efficiency, availability, security, and scalability of Internet-connected smart devices. Architecture for IoT uses blockchain technology, called DistBlockNet, to meet the principles required to design a secure, scalable, and efficient network architecture. Evaluation metrics: To assess scalability, defense effects, accuracy, and efficiency. Their work concerns with performance about blockchain network.

Hang and Kim (2019) describe that the IoT Blockchain Platform System Architecture is an integrated IoT platform using blockchain technology to guarantee to sense data integrity. This platform is to afford the device owner a practical application that provides a comprehensive, immutable log and allows easy access to their devices deployed in different domains. It allows for real-time monitoring and control between the end-user and device. The IoT devices are not included in the blockchain, and alternatively, a RESTful interface which handles requests from devices is defined to enable cross-platform communication between devices and the blockchain network.

Javaid, Aman and Sikdar (2018) state that the BlockPro provides and enforces data provenance and data integrity in IoT environments by using Physical Unclonable Functions (PUFs) and Ethereum, a blockchain variant with smart contracts. PUFs provide unique hardware fingerprints to establish data provenance while Ethereum provides a decentralized digital ledger which is able to withstand data tampering attacks. It is able to provide defense against data tampering attacks through a decentralized architecture. It also provides immunity from impersonation attacks.

3.8 CONCLUDING REMARKS

We conducted a Systematic Literature Mapping to investigate which are the primary development processes, and factors influence have been used in Blockchain-based the Internet of Things building. The ultimate goal of our research is to present the current panorama about best practices outlined in the literature to develop a Blockchain-based ontology for the Internet of Things projects. Blockchain-based the Internet of Things research area is so new and most of the papers and publications, as a book, technical report and others are concentrated in the last five years (i.e., 17 studies were considered as seen in Table 3).

We reported some frameworks, models, approaches, and other Blockchain-based the Internet of Things initiatives that present adherence to well-know development processes address to build an initial knowledge body. We detect key requirements in the Internet of Things and, we typed theirs as functionals and non-functionals requirements. Authors of the primary studies classified their works: (four) as frameworks, (4) as models, (2) as methods, (3) as approaches and, we classified (4) papers as other initiatives addressed in initial descriptions or superficial studies. We summarized the domain type: (six) as generic, (8) as specific and, (3) as non-specify. To identify essential characteristics, processes, modeling phases, tasks, and products. We evaluated theses works to adherence to MADEM methodology (knowledge modeling). Most (16) of these works emphasized the domain analysis, but just (7) theses presented domain design, into (3) on architecture design and, (2) on presented detailed design. These papers addressed the product modeling: (ten) as domain model, (5) as an architectural model and, (1) as an agent model. The community still has no better support to the design, architecture, integration, and testing processes to build Blockchain-based the Internet of Things.

Considering Uckelmann, Harrison and Michahelles' IoT key requirements, Table 7 depicts 100% of all studies addressed the kR_1, meet key societal needs for the Internet of Things including open governance, security, privacy and trustworthiness; followed by 70.6% both kR_2, bridge the gap between B2B, business-to-consumer (B2C) and machine-to-machine (M2M) requirements through a generic and open Internet of Things infrastructure; and kR_3, design an open, scalable, flexible and sustainable infrastructure for the Internet of Things; kR_4 develop migration paths for disruptive technological developments to the Internet of Things is covered by 64.7%; kR_5 excite and enable businesses and people to contribute to the Internet of Things is covered by 58.8%; followed by 52.9% both kR_6 Enable businesses across different industries to develop high added-value products and services and, kR_8 provide an open solution for sharing costs, benefits and revenue generation in the Internet of Things. The other IoT key requirements did not achieve at least 50%.

It has also been evaluated the adherence of each paper. In this analysis, the papers are evaluated against the 10 IoT key requirements.

We highlight the importance of the studies S14, S15, S16, S8, and S1. They discuss some main activities or artifacts or modeling of design, but they did not explicitly address theses activities or artifacts or modeling design in their propositions. However, they mention all the essential IoT characteristics processes. On the other hand, we considered that studies, S2, S5, S6, S11, S4, S7, S3, S9, S10, S17, S12 and S13 covered by 50% or less this did not explicitly address all the IoT fundamental processes.

Despite the efforts presented, we have identified the need to deploy and evaluate a solution that is closer to actual conditions. The idea is to address and measures the effectiveness of the security solution that solves problems with reliability, integrity, availability, and immutability in systems Blockchain-based IoT. Our work addressed the main characteristics, models and, tasks to integrate existing approaches and scalable blockchains and in designing an architecture for the Internet of Things applications to integrity, trust, and security issues. Moreover, we concentrated on building an initial Knowledge Body about Blockchain-based the Internet of Things devices.

Table 1 – Tools and GUIs used to evaluate and improve the BIoT ontology.

Tools	Validation criteria	Description
jena	Serialization and syntactic	Work with models, RDFS, and OWL to add additional semantics to your RDF data; Inference rules or built-in OWL and RDFS. Web service/API URL: https://jena.apache.org/
OOPS!	Ontology design	Knowledge catalog of most common problems. Web service/API URL: http://oops.linkeddata.es/
Pellet	OWL DL Reasoner	Support for Reasoning with individuals, user-defined data types, and debugging for ontologies. Web service/API URL: https://github.com/stardog-union/pellet/
WebVOWL	Visualization	Web application for interactive visualization of ontologies. Web service/API URL: http://vowl.visualdataweb.org/webvowl.html
Protégé	Syntactic; OWL DL and Functional Extensions	Authoring ontology and knowledge management system. Web service/API URL: https://protege.stanford.edu/
LOV	Discoverability	LOV stands for Linked Open Vocabularies, Definitions of a set of classes and properties. Describe specific types of things. Web service/API URL: http://lov.okfn.org/dataset/lov/
Parrot	Documentation	A RIF and OWL documentation service. Web service/API URL: https://labs.mondeca.com/parrot
OWL Manchester	Syntactic	The syntax used to write OWL ontologies. Web service/API URL: http://visualdataweb.de/validator
LogMap	Interlinking	Check satisfiability of each class of the integrated ontology: HermiT Reasoner; Lite (basic string similarities and no repair); With user interactivity and repair. Check the OWL 2 profile of your ontologies with the Manchester OWL 2 validator. Web service/API URL: https://www.cs.ox.ac.uk/isg/projects/LogMap

Source: The Author (2018).

Table 3 – List of included primary studies.

Study ID	Included Study	Year	Source
S1	LUU *et al.*	2016	ACM
S2	LUU *et al.*	2015	ACM
S3	VIGNA; CASEY	2015	ACM
S4	BIRYUKOV; KHOVRATOVICH; PUSTOGAROV	2014	ACM
S5	ZHANG; WEN	2017	IEEE
S6	ZHANG; WEN	2015	IEEE
S7	HARDJONO; SMITH	2016	Snowballing
S8	ATZORI	2017	Manually
S9	CONOSCENTI; VETRO; MARTIN	2016	Snowballing
S10	CHRISTIDIS; DEVETSIKIOTIS	2016	Manually
S11	HASHEMI *et al.*	2016	Manually
S12	HUCKLE *et al.*	2016	Manually
S13	UCKELMANN; HARRISON; MICHAHELLES	2011	Manually
S14	PANIKKAR *et al.*	2015	Snowballing
S15	NORTA	2015	Snowballing
S16	ZYSKIND; NATHAN; PENTLAND	2015	Snowballing
S17	ZYSKIND; NATHAN *et al.*	2015	Snowballing

Source: The Author (2016).

Table 4 – Primary studies inclueded for search strategy.

Source Studies	Retrieved	Duplicated	First Phase	Second Phase	Included
ACM Digital Library	6	-	6	5	5
IEEE Xplore	1	1	-	-	-
ISI Web of Science	1	1	-	-	-
Science Direct	3	-	1	-	-
Engineering Village	2	-	2	2	2
Manually	5	-	5	5	4
Snowballing	7	-	7	7	6
Total	25	2	21	19	17

Source: The Author (2016).

Table 5 – Summary of the modeling phases and tasks of the MADEM Methodology.

Phases	Tasks		Products	
Domain Analysis	Concept Modeling		Concept Model	Domain Model
	Goal Modeling		Goal Model	
	Role Modeling		Role Model	
	Variability Modeling		*In the models above*	
	Modeling of Role Interactions		Role Interaction Models	
Domain Design	Architectural Design	Mapping of the Role Model into a first draft of an Agent Society Model	Agent Society Model	Architectural Model
		Mapping of the Role Interaction Models into first drafts of the Agent Interaction Models	Agent Interaction Models	
		Reorganization of the agent society through cooperation and coordination mechanisms	Coordination and Cooperation Model	
	Detailed Design	Identification of a detailed design pattern	Agent Template	Agent Models
		Definition of the agent type		
		Modeling the agent behavior	Agent Behavior Model	
	Modeling the knowledge of the multi-agent society		Model of the Multi-agent Society Knowledge	

Source: Girardi and Lindoso (2005).

Table 6 – Caracterization of the included studies.

			Frameworks				Models				Methods		Approaches			Other Initiatives				Total
			S2	S14	S15	S16	S1	S5	S6	S11	S4	S7	S3	S9	S10	S17	S8	S12	S13	
Domain	Type	Generic				x			x		x	x				x		x		6
		Specific	x	x	x		x	x		x					x		x			8
		No specify											x	x					x	3
Modeling	Phases	Domain Analysis	x	x	x			x	x	x	x	x	x	x	x	x	x	x	x	16
		Domain Design	x	x	x	x	x			x		x				x				7
		- Architectural Design		x					x							x				3
		- Detailed Design	x				x													2
	Products	Domain Model	x	x	x	x	x	x		x				x		x		x		10
		Architectural Model		x			x		x			x				x				5
		Agent Models										x								1

Source: The Author (2016).

Table 7 – IoT key requirements adherence of the included studies.

Key Requirements	Frameworks				Models				Methods		Approaches			Other Initiatives				%
	S2	S14	S15	S16	S1	S5	S6	S11	S4	S7	S3	S9	S10	S17	S8	S12	S13	
1. Meet key societal needs for the Internet of Things including open governance, security, privacy and trustworthiness.	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	100,0
2. Bridge the gap between B2B, business-to-consumer (B2C) and machine-to-machine (M2M) requirements through a generic and open Internet of Things infrastructure.	x	x	x	x	x		x	x	x	x			x	x	x			70,6
3. Design an open, scalable, flexible and sustainable infrastructure for the Internet of Things.		x	x	x	x	x		x		x	x	x	x		x		x	70,6
4. Develop migration paths for disruptive technological developments to the Internet of Things.		x	x	x	x		x	x	x	x		x		x			x	64,7
5. Excite and enable businesses and people to contribute to the Internet of Things.	x	x	x	x	x	x					x		x		x		x	58,8
6. Enable businesses across different industries to develop high added value products and services.		x	x	x		x	x	x	x						x	x		52,9
7. Encourage new market entrants, such as third party service and information providers, to enter the Internet of Things.		x	x	x					x			x						29,4
8. Provide an open solution for sharing costs, benefits and revenue generation in the Internet of Things.	x	x			x	x	x			x			x	x	x			52,9
9. Public initiatives to support the usage of the Internet of Things for social relevant topics.		x	x															11,8

Source: The Author (2016).

Table 8 – List of related work.

Related work	PRO	DAT	ACC	PRI	INT	ANO	CON	AVA	ASY	SYM
(MENDONÇA; SILVA JÚNIOR; ALENCAR, 2017) (MENDONÇA *et al.*, 2016)	x	x	x	x	x		x		x	x
(SHARMA *et al.*, 2017)	x	x	x							
(HAMMI *et al.*, 2018)				x						
(WÜST; GERVAIS, 2018)	x	x		x		x				
(JAVAID; AMAN; SIKDAR, 2018)		x	x		x					
(LIU *et al.*, 2017)		x			x					
(BRAMBILLA; AMORETTI; ZANICHELLI, 2016)			x	x						
(HARDJONO; SMITH, 2016)				x		x				
(YUE *et al.*, 2016)				x	x		x			
(BASNET; SHAKYA, 2017)			x	x		x		x	x	x
(XU *et al.*, 2019)	x			x			x			
(DORRI *et al.*, 2017)								x		
(NGUYEN; PHAM; THAI, 2018)			x	x						
(STEICHEN; HOMMES; STATE, 2017)	x			x			x		x	x
(MENDEZ MENA; YANG, 2018)									x	x

Legend: PRO: Prevention/protection; DAT: Data protection; ACC: Access control; PRI: Privacy; INT: Integrity; ANO: anonymous access; CON: Confidentiality; AVA: Availability; ASY: Asymetric keys; and SYM: Symetric keys.

Source: The Author (2018).

4 BIOT ONTOLOGY

We exhibit in Chapter 2 the requirements to build the proposal of identification of entities and their interrelations. This strategy aimed at creating a solution that has the technologies associated with the Blockchain-based ontology for the Internet of Things Security (BIoT). These requirements support the BIoT proposition, influencing the determination of its application, the organization of services, and the set of functionalities identified. This Chapter presents the BIoT ontology, its profile, requirements of middleware, cryptographic strategy, principles of BIoT ontology, BIoT preliminaries, and concluding remarks.

4.1 REQUIREMENTS OF MIDDLEWARE FOR BIOT ARCHITECTURE

In considering the motivations for this work, the study of the aspects introduced by the Internet of Things and Blockchain, associated to the analysis of works related to BIoT Ontology, in addition to the scope of action of this proposal, are identified the requirements that should be met by BIoT.

Figure 26 presents a synthetic vision of the technologies that the BIoT proposes to integrate, being characterized the adaptation to the context as a central aspect for its convergence.

In this section, we present the requirements that are listed, analyzed when describing the services and procedures required.

4.1.1 From application

Considering the profile of the target application, and the corresponding characteristics, the Middleware to support it must meet the following requirements:

- distributed execution support;
- communication with decoupling;
- access and validation of blocks in a pervasive way;
- support logical and physical mobility;
- sensitive to context information;
- support for the semantic adaptation of functional and non-functional aspects;
- intermittent connectivity between LAN and WAN.

4.1.2 From middleware

To build Middleware, the following requirements are identified as necessary to meet the proposed architecture.

- Dynamic adaptation to context;

- Resource saving;
- Operation disconnected.

4.1.3 Middleware layer

Among the many traditional middleware options available for distributed systems can be classified into three broad groups or categories: object-oriented middlewares, message-oriented Middleware, and transaction-oriented Middleware. And their classifications are presented based on aspects of communication between components.

- *Object-Oriented Middlewares*: Their main goal is to manage requests between distributed objects. A client object can request execution of processing on a server object that can be stored on any node of the distributed system, handling those requests through Remote Procedure Calls (RPC) synchronously, which will be blocked until the object-server has returned a response. This proposal has been widely adopted, however, we must highlight its limitations in pervasive computing due to three factors: (i) computational cost; (ii) the synchronous request, presenting low scalability; and (iii) the principle of transparency, which presents difficulties for the construction of mechanisms that provide consistency of execution context.
- *Message Oriented Middlewares*: Provides communication between the application has distributed components through the exchange of messages. The client components send a message containing the request for a service and its parameters to a server-side component over the available network. Messaging-oriented Middlewares enable the implementation of asynchronous communication mechanisms, thereby enabling an operational decoupling between client and server (an important aspect in pervasive computing), allowing the client to continue processing as soon as Middleware accepts its message to send. On the other hand, the server component may return a response message (which is now managed by the Middleware), and the client component can decide the best computational moment to collect it. Also, it can lead to a need for increased memory capacity to store these messages until they are selected and sent for processing, which can be restrictive to mobile devices. Other aspects can also be taken into account from this Middleware, greater need for processing requires more robust hardware, lack of context awareness where the execution will take place, aspects and limitations involved in execution and management, as well as the messages being handled by the Middleware without involvement application.
- *Transaction Oriented Middlewares*: They are primarily aimed at applications that manipulate databases and have a high degree of reliability. Its main feature is to support transactions involving components that are running on different nodes. A client component groups a request into a transaction. Middleware is responsible for directing transactions between the server components that are transparently distributed to both clients and servers. It presents a guarantee of atomicity required in the management of transactions, but it produces a high computational cost, which can make projects of this nature impracticable

since the use of transactions in pervasive computing is not always indispensable. Another concern is that the physical mobility of the devices may imply non-permanent connections, which may increase the complexity of dealing with requests and transactions.

One factor to be considered in the design of sensor networks is the heterogeneity, the distinct or unequal nature, of the various interconnected devices. It brings us to the challenge of building interoperable networks that integrate the most varied devices, technologies, purposes and performance under the same network structure, as well as allowing a flexible dynamic for the increment of new devices to the network. In this context, middleware management platforms, named Middleware, are developed that mediate the communication between various software and devices. In general, in order to meet the application needs, Cavalcante *et al.* (2015) suggests that a basic middleware structure should present as requirements: a) interoperability; b) device discovery and management; c) scalability; d) context science; e) managing large volumes of data; and, f) security.

Interoperability is an essential property in wireless sensor networks as they integrate a variety of devices from a variety of building, transmission, and data technologies. In this regard, the middleware layer plays a significant role, since it focuses on the development models and protocols that allow the necessary abstraction for the construction of more flexible sensing structures.

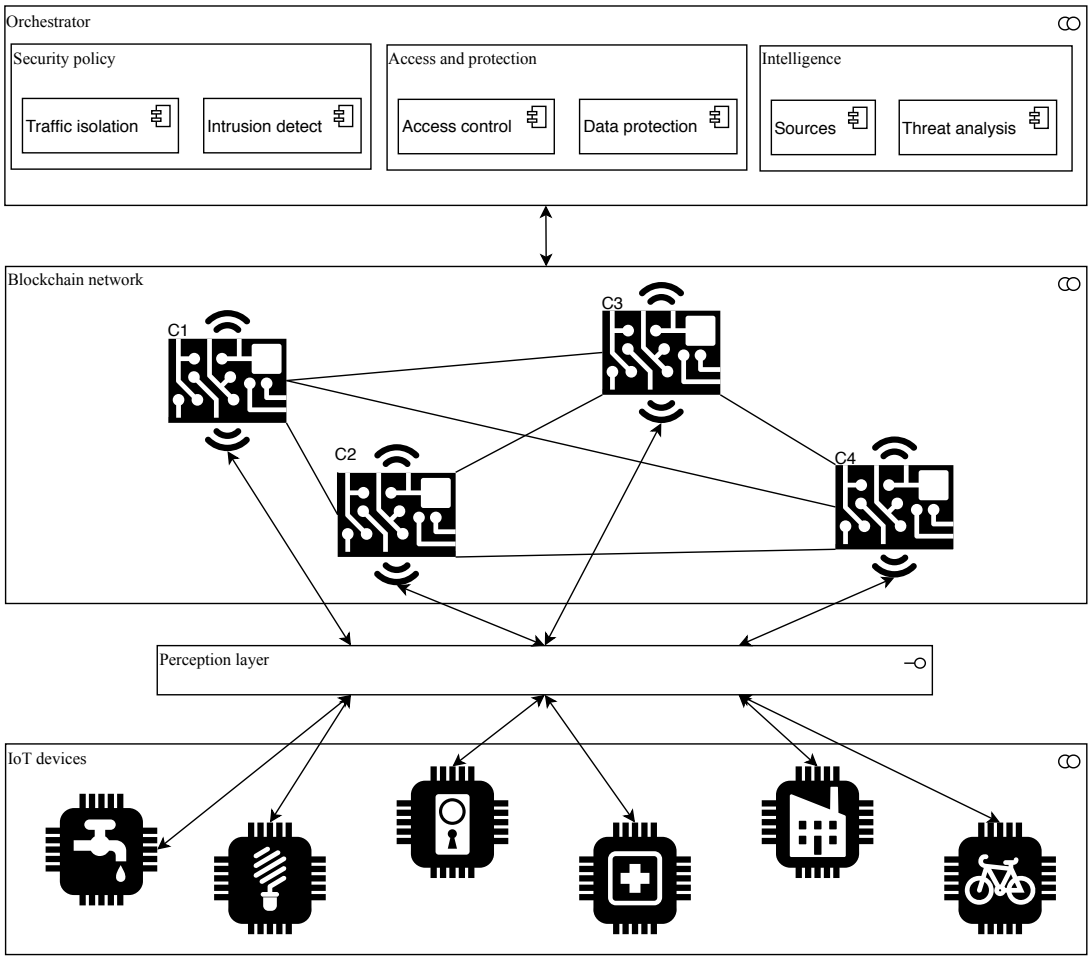
The BIoT adopts secure network based on Blockchain network for IoT communication framework to improve security, scalability, privacy, and access control without the need for a central controller. Figure 23 shows the views of the IoT device structuring layers, perception layer, blockchain network, and its orchestrator. In the Orchestrator, we looked at security policy, access control and data protection, and threat intelligence modules.

Another essential property in wireless sensor networks is reflected in the flexibility of the network that must be built to allow the dynamic inclusion of new devices without the need to stop the entire structure; this property is called discovery and management of devices. This property is closely tied to scalability, which is the ability to dynamically assimilate the growing number of new devices without compromising network functionality.

Contextual science refers to metadata that identifies, among other things, state properties, connectivity, and device location. The management of large volumes of data permeates the whole solution of a wireless sensor network because while a scalable network incorporates new sensor devices, they will provide more data that should be properly: a) transmitted; b) stored; c) recovered; and, d) processed.

No less important is the data security that can be obtained from techniques of masking and prevention of data corruption, such as cryptography, which must maintain a: a) integrity, data must travel throughout the structure without undergoing modification in content; (b) privacy, data shall be transmitted and stored so that it is not legible by unauthorized means; c) availability,

Figure 23 – Overview of architecture of BIoT.



Source: The Author (2018).

access to the data must be available whenever requested by those who are duly authorized; and, d) reliability, the data must represent exactly what was initially collected at its origin.

The middleware layer, particularly considering the WLC architecture, is composed of: a) collector module, responsible for receiving data from the various sensors; and send them to one, b) database manager system (DBMS), responsible for managing the persistence of application data.

The MQTT protocol is an industry-standard that suggests a model for managing to message between devices. It was developed by IBM to manage data transmission over intermittent or low-bandwidth data networks. It has the following main properties: a) low implementation complexity; b) optimized for application in both TCP / IP and non-TCP / IP networks; c) guarantee of high rate of package delivery; d) message management through the sending of control packets; and, e) allows the construction of application with low electrical consumption.

Using the application for MQTT protocol, a broker application was introduced in the data collector module, acting as an intermediary in capturing the data transmitted by the sensor

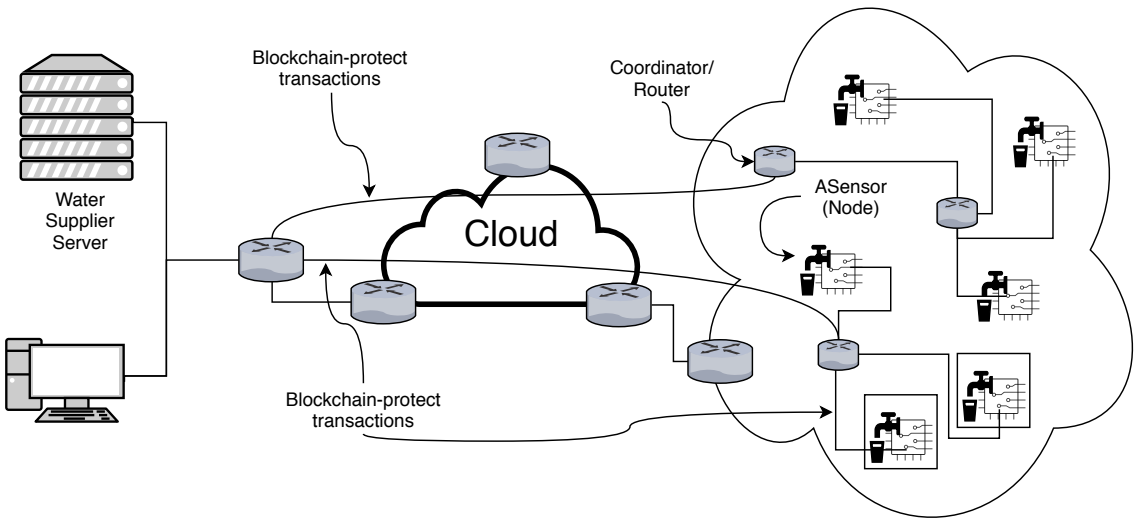
modules, which play the role of publisher, and in storing the data in a DBMS. The implementation of the MQTT protocol chosen for the development of this project was Mosquitto4.

Appendix A contains the source code for the sensor module developed for this project, followed by the source code used to control the remote monitoring module, in Appendix B. It was necessary to use the PubSubClient library to aid in communication through the MQTT protocol. The source codes that make up the library are listed in Annex A, where the user license is initially presented in Algorithm A.1, followed by the source code of the PubSubClient.cpp component, in Algorithm A.2, and finally, the source code of the PubSubClient.h component, in Algorithm A.3. The DBMS chosen was MariaDB, due to its free license and for reasons of compatibility with the SQL standard, MariaDB also presents a low hardware requirement and good speed compared to other standard DBMSs on the market.

For the data persistence, a data model composed of two entities was used, one is responsible for identifying the sensor module and the other responsible for storing the data read. The first entity, called sensors, stores the sensor ID and its inheritance relationship, and the second entity, called the datastream, stores the data stream containing the source sensor ID and a time signature. Figure 26 illustrates the data model used in the middleware layer of the WLC architecture proposed in this work.

In our scenario, Mendonça *et al.* (2016) have built a solution to improve the water distribution, allowing better planning and regional strategies, and enhance the domestic consumption of the treated water, by applying the IoT concepts, an overview can be seen on Figure 24.

Figure 24 – Blockchain-based the Internet of Things Security application to Water Flow Controller overview.



Source: The Author (2017).

We proposed the use of smart devices to perform, autonomously (or with lower power

consumption). For this, we have searched commercial solutions, performing research with systematic methods related to the control of the demands and offers and the supply of water.

4.1.4 Public key management to BIoT

Cryptographic key management and distribution are complex, involving cryptographic, protocol, and management considerations. The problems involved and a broad study of the various aspects of key management and distribution are of great concern. The starting point is NIST SP 800-57.

Due to the inefficiency of public-key cryptosystems, they are rarely used for direct encryption of the data block of considerable size, and they are limited to relatively small blocks. We use symmetric key distribution resources asymmetric encryption as a basis, one of the most appropriate uses of a public key cryptosystem is to encrypt symmetric keys for distribution through a hybrid scheme.

A hybrid technique, already used on IBM mainframes, shares a secret master key with each user, and distributes secret session keys encrypted with the master key through a public key distribution scheme in which we have two major concerns:

- Performance: Several systems have the need to modify session keys very frequently, which could decrease system performance due to the high computational cost in encrypting and decrypting these public keys. The central idea is to keep public keys cached after use and to occasionally update them, between WLC and Coordinators or Routers and the Key Distribution Center.
- Compatibility: The hybrid scheme is easily covered in an existing Key Distribution Center (KDC) scheme, with minimal disruption or software changes.

Some techniques have been presented for the distribution of public keys. These proposals can be grouped as follows:

- Public announcement: Displays the differential of public-key encryption, which is public. However, it has a weakness. Anyone can fake the public announcement.
- Publicly available directory: A higher degree of security can be achieved by maintaining a publicly available dynamic directory with public keys. It presents the disadvantage of having to be under the responsibility of a responsible entity or organization, which would prevent the distributed application present in the Blockchain approach.
- Public key authority: We apply this public key distribution strategy by presenting the strongest security strategy for public key distribution through strict control. Even though this approach has some disadvantages, for example, nodes need to request, with some frequency, to communicate with other nodes of the network. This behavior is mitigated by the technique that saves the key, known as caching, which reduces. More details of the implementation can be seen below.

- Public-key certificate: An alternative technique, initially suggested by Kohnfelder, is to use certificates that can be used by the network nodes to exchange keys without contacting an even more reliable public-key authority against public keys obtained from an authority of public key. However, this approach would also require a trusted public-key entity signature. In addition to some difficulties encountered of compatibility of the generation of certificates and compatibility between devices involved in BIoT.

Public key authority for the BIoT

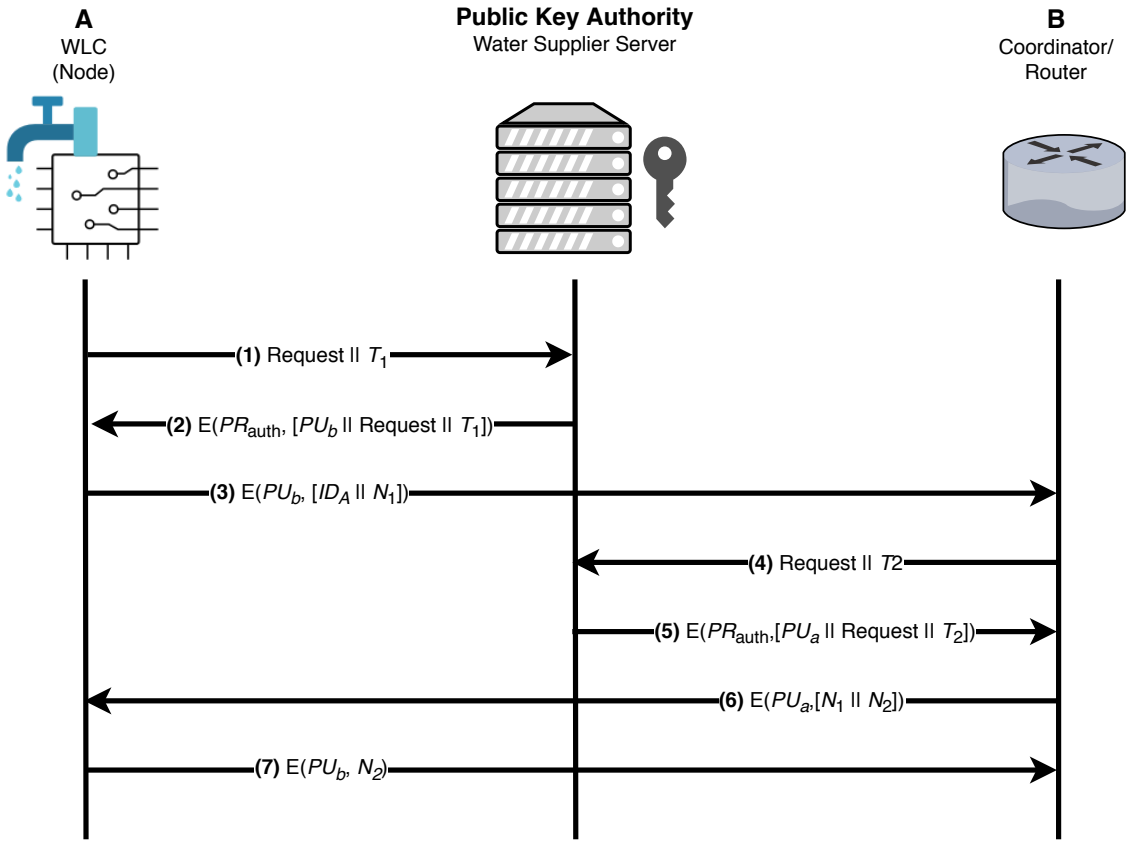
Stronger security for public key distribution can be obtained by offering tighter control over directory distribution of public keys. We use a typical scenario (POPEK; KLINE, 1979). In our scenario, a central authority maintains a dynamic public key directory of all WLCs and Coordinators or Routers. Each WLC knows a public key of each Coordinator or Router, being able to verify and validate transactions from each associated WLC. The following steps (combined by number with Figure 25) occur:

1. *A* sends a timestamped message to the public key authority, containing a request for the current public key of *B*.
2. The authority responds with a message that is encrypted using the authority's private key, PR_{auth}. Thus, *A* can decrypt the message using the authority's public key. Therefore, *A* has assurances that the message originated by the authority. The message includes the following:
   - The public key of *B*, PU_b, which *A* can use to encrypt messages destined for *B*.
   - The original request to allow *A* to compare this response with the previous request corresponds to and that the original request was not changed before receipt by the authority.
   - The original timestamp so that *A* can determine that this is not an old message from the authority, containing a key different from the current public key of *B*.
3. It stores the public key of *B* and also uses it to encrypt a message to *B*, containing an identifier of *A* (ID_A) and a nonce (N_1), which is used to identify that transmission exclusively.
4. *B* obtains the public key of *A* in authority;
5. In the same way as *A* obtained the public key of *B*.

At that point, the public keys were safely delivered to *A* and *B*, and they can start trading protected. However, two additional steps are desirable:

6. *B* sends a message to *A* encrypted with PU_A and containing the nonce (N_1) of *A*, in addition to a new nonce generated by *B* (N_2). As only B could have decrypted the message (3), the presence of N_1 in the message (6) assures *A* that the correspondent is *B*.
7. *A* returns encrypted N_2, using the public key of *B*, to guarantee *B* that its correspondent is *A*.

Figure 25 – Public-key distribution scenario to BIoT.



Source: The Author (2017).

Key Decentralized Control

The use of a Key Decentralized Control (KDC) requires that it be trusted and protected against criminal acts. This risk can be avoided if the distribution of keys is decentralized. One of the problems of using key decentralization is that its application is not so simple for large networks that use only symmetric cryptography, but it has advantages in local or even hybrid context. The hybrid approach is adopted in BIoT, in Tree, Tree Cluster, Star, or Mesh Network Topology, described in the work of Mendonça *et al.* (2016).

In the BIoT, each node of the system can communicate with other nodes safely, distributing session keys. For system security and stability, we restrict the limit of the number of master keys, based on the work of (REARDON, 2016), to $[n(n-1)]/2$ for a configuration of n WLC (Node), Coordinator, Router or other end systems. The short lifetime of the session keys favors system protection, as shown in Figure 24, p. 83.

1. *A* issues a request to a *B* for a session key and includes a nonce, N_1.
2. *B* responds with a message that is encrypted using the shared master key. The response includes the session key selected by *B*, an identifier of *B*, the value $f(N_1)$ and another nonce, N_2.

3. Using the new session key, *A* returns $f(N_2)$ to *B*.

4.2 THE PROFILE FOR BIOT

According to the literature, the term ontology was first used in the area of Computer Science in the works of Mealy (1967), Another Look at Data, in which three different approaches are presented for the area of data processing: a) the real world; b) that which exists in the human mind; c) symbologies stored on paper or some other medium.

The ontologies have been presented in several initiatives and researches in the areas of Database, Software Engineering, and Artificial Intelligence (SMITH, WELTY, 2001).

4.2.1 Architecture for BIoT

Figure 26 provides an overview of the software architecture highlighting interest area of the actuation for BIoT Ontology, in Panikkar *et al.* (2015)'s overview. The ontology representation in this figure as a virtual module aims to emphasize its importance in architecture and characterize the presence in the design of other components.

With the objective of minimizing the costs of specifying the aspects necessary for the semantic distribution and adaptation treatment offered by the RDF or OWL language, already integrated into the execution environment and modeled in this perspective, it presents a logical organization through the reference layers, and more specifically in front of the Open Source Protocols layer.

At the top layer (or application layer) are the user interface and business logic directed to the Internet applications of Blockchain-based Stuff.

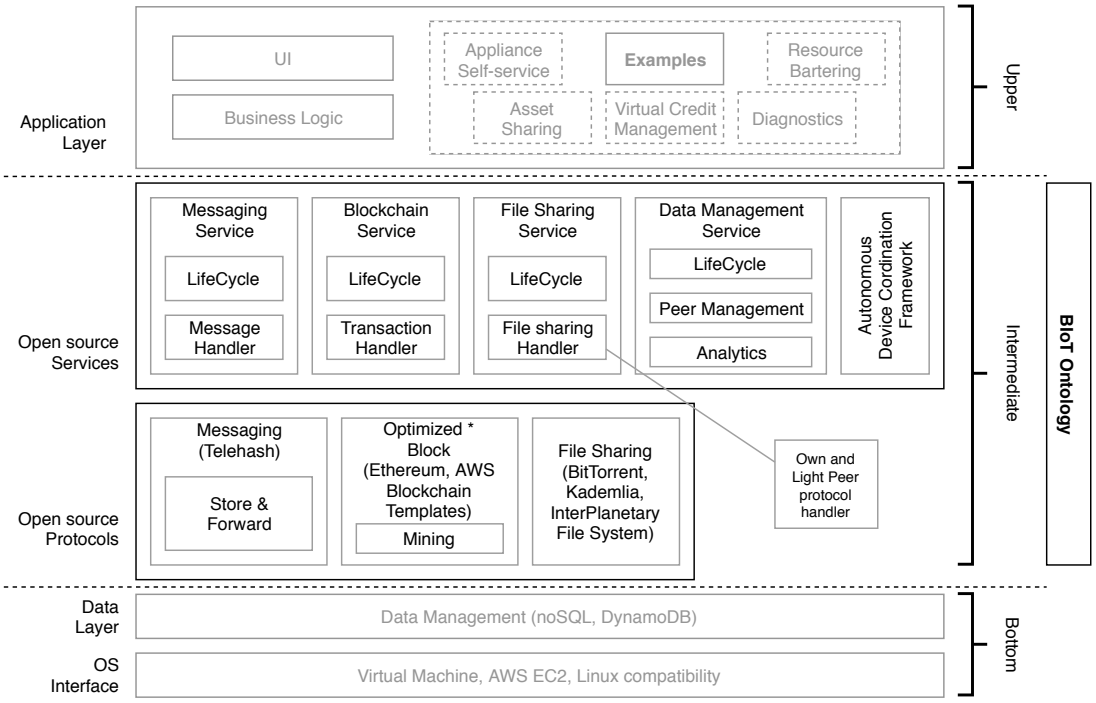
The middleware layer (BIoT Ontology) are the mechanisms to support the execution of the Internet of Things based on Blockchain and to the strategies of adaptation. This layer is composed of two levels:

The first level described in the reference model as Open Source Services consists of five modules, Messaging Service, Blockchain Service, File Sharing Service, Data Management Service, and Autonomous Device Coordination Framework.

The second level of the middle tier is the Open Source Protocols, the basic BIoT Ontology protocols, which provide the entities and functionalities needed for the first level and cover various aspects such as (WANG *et al.*, 2018):

Messaging (Telehash) mechanisms to check the status of devices, through requests to protocols that basically provide four messages: `ping`, used to verify that a node is still alive; `store`, stores a (key, value) pair in one node; `find_node`, the recipient of the request will return the *k* nodes in his buckets that are the closest to the requested key; `find_value`, same

Figure 26 – Blockchain-based Internet of Things Security Architecture – Logical View.



Legend: Figure is highlighting the area of interest of the BIoT Ontology, intermediate.

Source: Adapted from Panikkar *et al.* (2015).

as `find_node`, but if the recipient of the request has the requested key in its store, it will return the corresponding value;

Optimized Block (Ethereum) Ethereum is a distributed software platform based on a public and open-source Blockchain that allows developers to create and deploy decentralized applications, which means that it uses a peer-to-peer approach. Each interaction occurs and is supported between and only by users participating in it, with no controlling authority involved. A global system of so-called nodes supports all the Ethereum system. Each node lowers all Blockchain and applies consensus rules. These consensus rules, as well as countless other aspects of the network, are dictated by "smart contracts", designed to automatically perform transactions and other specific actions on the network with parts that do not necessarily trust. The terms for both parties comply are pre-programmed in the contract. The completion of these terms then triggers a transaction or any other specific action. The system also provides its users with the Ethereum Virtual Machine (EVM), which essentially serves as a runtime environment for Ethereum-based smart contracts. It provides users with security to run untrusted code, ensuring that programs do not interfere with each other. EVM is completely isolated from Ethereum's core network, making it a perfect sandbox tool for testing and improving smart contracts;

File Sharing (BitTorrent) each file must be associated with a torrent, a small file that contains information needed for sharing. The information represents data that confirms the integrity

of the torrent, addresses of trackers (servers that guide communication). Some of the main fields found in a torrent file: a) announce, describes the file distributor tracker; b) announce-list: any auxiliary trackers; c) comment: comments inserted by the creator; d) created by torrent creator software; e) info: file data, name, size, hash code. Other concepts involved are: a) seed: each machine that has the complete file to be shared; b) peer: each computer that receives or shares files (or part of files); c) leecher: nodes that have already downloaded files (or their parts) but are not sharing; d) tracker: server that keeps track of the communication between all the seeds and peers, and which machines to connect to; e) swarm: set of computers that are sharing the same file.

The lower layer of the architecture is made up of the lower levels of the operating system, native languages, and physical medium of execution. To provide portability, we base the deployment through virtual machines, or even maintain the platform in the cloud environment, in services such as Amazon Web Services.

The BIoT is organized in logical layers, with differentiated levels of abstraction, and is directed to the maintenance of the quality of the services offered to the user, by concept of adaptation. The system adapts to provide quality to the services provided, while the application adapts to meet the expectations of encapsulation and transport, safely while maintaining the functionality of the application.

4.2.2 Scalability issues for IoT based on blockchain

Applying blockchain knowledge to IoT resources enables data transmission strength, traceability, and accountability, providing greater security when compared to the classic resource-network connection model.

In practice, the reliability of the connection is built by comparing the language adopted for communication of interest (network resource) and the nature of this communication that occurs in 4 steps (RUTA *et al.*, 2017):

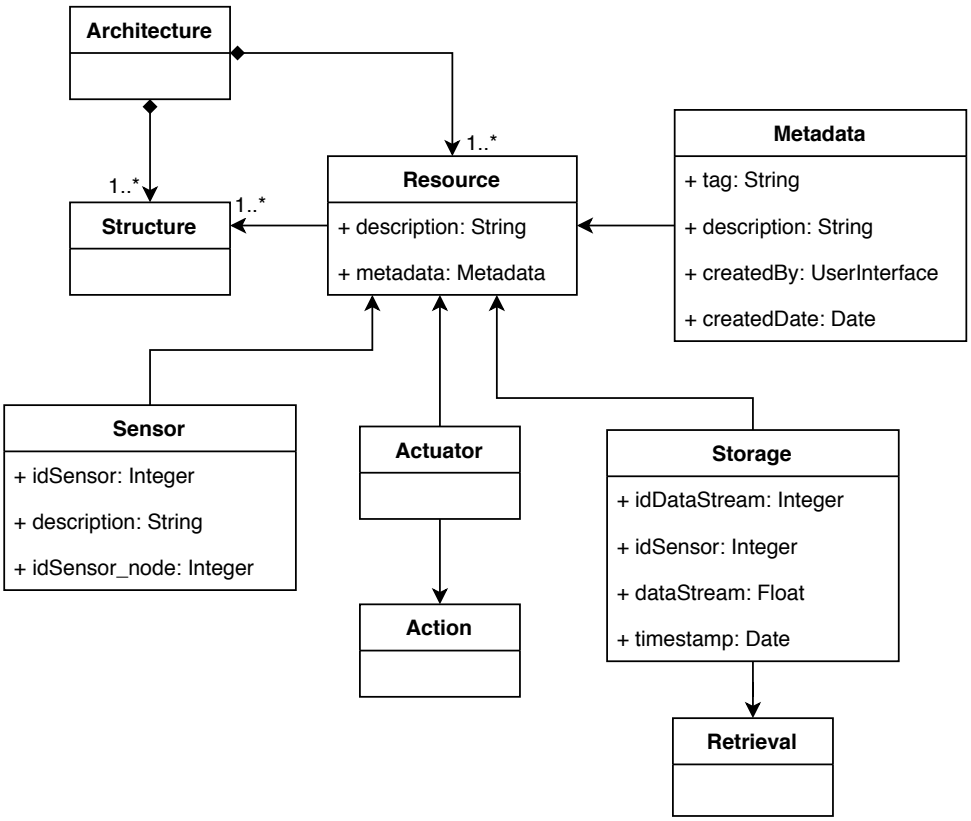
- *Registration*: Through a URI is recognized the uniqueness of the resource according to its nature; This way, it is possible to quantify the effort required for its registration in the network.
- *Discovery*: Through the language adopted by the parties interact, it is possible to evaluate if the communication is effective, ie, if the "answers" match the requests.
- *Explanation*: If necessary, compatibility between the parties can be justified by the result obtained, which in turn can be stored and used to recognize future connections.
- *Selection*: At this stage, the connection would be established because it was recognized as secure.

4.3 PRINCIPLES OF BIOT ONTOLOGY

The premise of integrating the scenarios (i) of Blockchain technologies, (ii) the Internet of Things, (iii) Telehash and (iv) BitTorrent, is mapped in an organization composed by the aggregation of execution cells, shown in Figure 26, p. 88.

The physical environment where BIoT is defined consists of a standard network infrastructure, whose composition can be modified by the dynamic aggregation of new nodes, characterized by the Internet systems of Things, with distributed and hybrid devices and well-defined parameters of admission on the network.

Figure 27 – The Internet of Things (Local Water Control) class diagram – overview.



Source: The Author (2016).

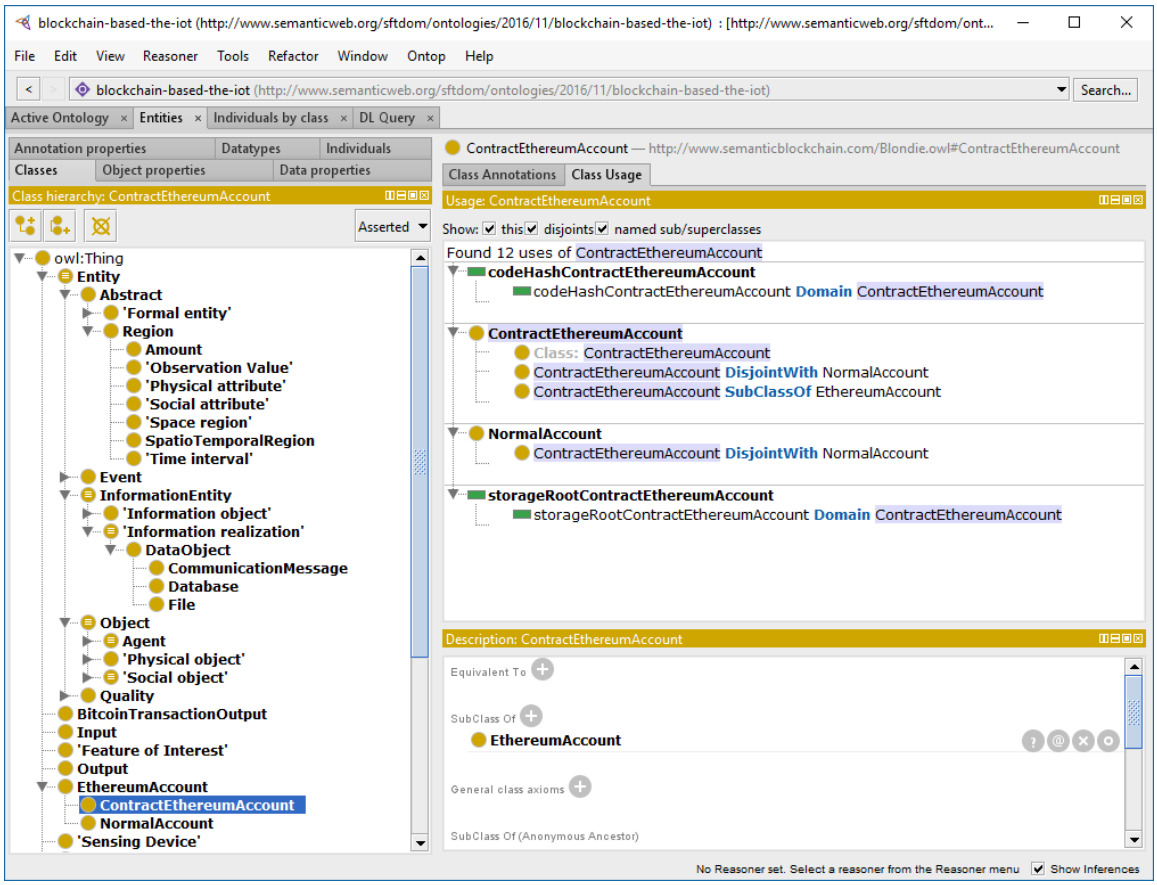
4.3.1 Layers of BIoT overview

The proposed ontology is engineered using a layered approach. Conceptualizations related to sensing also, observations were reused from the SSN ontology (*ssn* prefix for namespace of the SSN layer[1]), high-level generic ones from the DUL[2] ontology namespace of the DUL

[1] http://purl.oclc.org/NET/ssnx/ssn#
[2] https://www.w3.org/2005/Incubator/ssn/wiki/DUL_ssn

layer), BLONDiE[3] – Blockchain Ontology with Dynamic Extensibility, high-level generic ones from the IoT Ontology[4], and IoT-lite Ontology[5].

Figure 28 – Overview of Blockchain-based the IoT Security entities definition in Protégé 5.1.0 notation showing the entities and classes previously identified.



Source: The Author (2016).

On the limitation of existing ontologies to fully meet the requirements for the proposed Blockchain-based ontology for the Internet of Things Security, this project introduces the definition of new concepts and properties:

1. representation of higher-level entities of the Blockchain-based the IoT Security world, and
2. for the representation of Blockchain-based the IoT Security entities alignment.

These all based on the main requirements of the IoT and security, privacy, and trust environment of the Blockchain technology.

4.3.1.1 biot:BIoT_Entity

The highest (top-level) concept of the Blockchain-based the IoT Security (biot) entities layer is the concept *biot:BIoT_Entity* (as seen in Figure 28), specified as a Blockchain-based

[3] https://github.com/hedugaro/Blondie
[4] https://archive.org/services/purl/purl/IoT/iot
[5] http://iot.ee.surrey.ac.uk/fiware/ontologies/iot-lite

Internet of Things entity that has a distinct, separate situation, existence or view and is consistent with ('satisfying') a description of a set of entities. It is a subclass of *ssn:Situation* (a social object that satisfies some description) and of *ssn:FeatureOfInterest* (a relation between an observation and the entity whose quality was observed) restriction to some *dul:PhysicalObject* (any object that has a proper space region).

4.3.1.2 biot:SmartEntity

A *biot:SmartEntity* is a *biot:BIoT_Entity* that represents the association between exactly one physical object (*dul:PhysicalObject*) observed and some other objects (*dul:includesObject*) such as a *ssn:Sensor*, an *iot:Actuator*, an *iot:EmbeddedDevice*, or an *iot:Identifier*. The observation is represented using the feature of interest property restriction to physical object.

4.3.1.3 Bottom

Things layer is comprised of things that are subject to the automation offered by the IoT. This is a large domain, including (for example) people (with wearables, e/m-health medical monitoring devices, etcetera), smartphones, appliances (e.g., refrigerators, washing machines, air conditioners, etcetera), homes and buildings (including HVAC and lighting systems), surveillance cameras, vehicles (cars, trucks, planes, construction machinery), utility grid elements. In-layer security, we have into device-level blockchain guarantees authorization and authentication; encryption, key management; trust and identity management.

Data acquisition layer covers "data acquisition" features. It is physically made up of sensors (appropriate to the thing and the layer of the high layer), embedded, embedded, sensors and others. Layer 1 and layer two may appear to be warranted worldwide, to sensors, voice, video, multimedia, location, as shown Table 9.

Listing C.5 instances the definition of a context element named "`analogToDigital-Acquisition`", to receive the analog data from the sensor and convert to digital data.

Fog networking layer supports "fog networking", that is, the localized (location- or neighborhood-specific) network that is the first hop of the IoT client ('device cloud') connectivity. Typically, fog networking is optimized to the IoT clients operating environment and may use specialized protocols. It could be a wired or a wireless link.

ContextManager service (see Figure 29) is responsible for handling the raw information produced by the monitoring and submission to the service for registration.

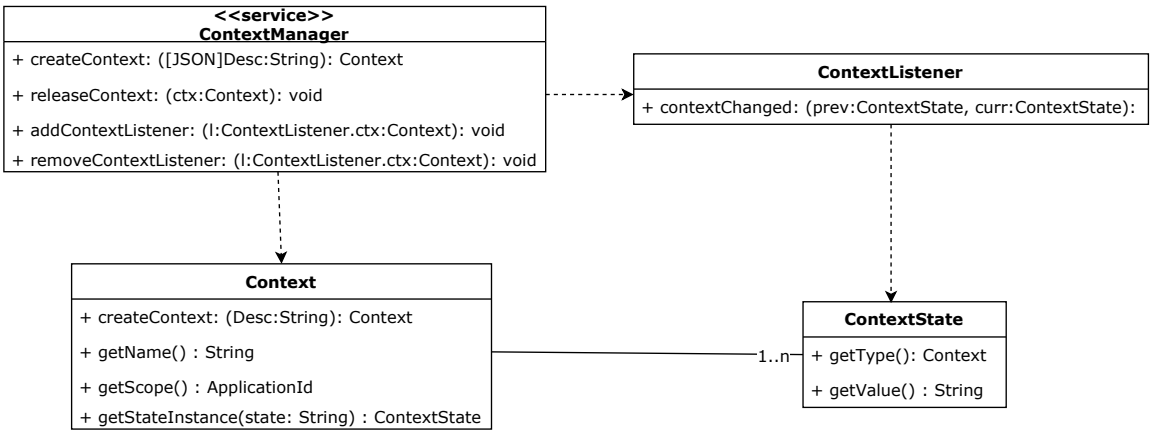
Context elements are defined by the `createContext()` method, and it returns a `Context` object, which refers to the context element, which defines all its `ContextState` possible states for the context element. The `createContext` method receives as parameter a JSON description of the data relating to that context element and must be produced through the information coming by monitoring.

Table 9 – BIoT ontology, *data acquisition by sensors* use case.

Name of BIoT	**Value proposition**
Use case	Sensors capturing data
Local use case	The sensors are the soldiers of the “Internet of Things”, the pieces of hardware that do the critical work of the processes of monitoring, measurements and data collection. They are often one of the first things people think while imagining IoT. When talking about sensors and their processes, there is a network of meanings that implies a dependence roles and functions (or tasks) of obtaining the data. The intended meanings include possible functional roles played by certain sensors, such as, for example, 1. proximity sensors, 2. accelerometer and gyroscope, 3. temperature, 4. humidity, 5. pressure, and 6. level. Therefore, the class and instance variables are present at the maximum ratio for this standard.
Logic addressed	OWL(DL) and OWL Lite
Reference ontologies	SSN, DUL, IoT Lite, BIoT ontologies

Source: The Author (2016).

Figure 29 – BIoT ContextManager service classes diagram.



Source: The Author (2018).

Open source services layer supports the “data centralization” function. It corresponds to the core networking functions of modern networks. It includes the functionality of typically found in institutionally-owned (core) networks, industry-specific extranets, public/private/hybrid cloud-oriented connectivity, and Internet tunnels. These networks achieve their functionality utilizing carrier-provided connectivity services and infrastructure and utilize wireline and wireless links.

Open source protocols layer supports the “data aggregation” function. This function may entail come kind of data summarization or protocol conversion, for example, mapping from a thin, low complexity protocol used by the IoT clients in consideration of low-power predicaments,

to a more standard networking protocol, as well as the edge networking capabilities. The data aggregation function is typically handled in a "gateway" device. Edge networking represents the outer tier of traditional network infrastructure, the access tier, employing well-known networking protocols.

4.3.1.4 Upper

Applications layer encompasses a vast array of horizontal or vertical applications or "application domains" (Use Cases). The list of applications is 'unlimited': applications include e/m-health, smart cities, smart building, smart grid, intelligent transport, surveillance, sensing, crowd-sensing, intelligent production, and logistics. In-layer security, we can highlight authorization and authentication; encryption and key management, trust, and identity management.

Data analytic and storage layer encompasses the data analytic and storage functions. It guarantees authorization and authentication; encryption and key management; trust and identity management.

4.4 BIOT PRELIMINARIES

Description logics are a family of logical languages for representing knowledge in a decidable fragment in First-Order Logic (RUTA *et al.*, 2017). Basic DL syntax elements are:

- Concept (class): names, standing for sets of objects, e.g., medicine, shape, sweeting_agent.
- Role (object property): names, linking pairs of objects in different concepts, such as hasDosage, hasShape;
- Individuals (instances): special named elements were belonging to concepts, e.g., Acetylsalicylic_Acid_Regular, Coated_Caplet.

Logical constructors combine the elements to compose concept and role expressions. Each DL has a set of constructors. The conjunction of concepts, represented as $\sqcap$, is available in all DLs; Some DLs also use disjunction $\sqcup$ compliment and complement $\neg$.

In this thesis, we consider the description logic $\mathcal{ALCI}$ and the particular domain ontologies used are Σ-inseparable in $\mathcal{ALC}$ because they link the same $\mathcal{ALC}$-concept inclusions in Σ between them and BIoT ontology. BIoT ontology has approximately 40 classes, 550 axioms, 180 logical axioms, 220 declaration axioms, and 45 object property count. In other words, we work with to handle the data captured by the sensors; Normalizing data for blockchain registration; Collecting data from blockchain; Monitoring blockchain status (security). However, in this work, we present only the main classes about Blockchain-based IoT Security.

- Asset
- Feature
- Attack
- Agent

Table 10 – $\mathcal{ALN}$ constructors.

Name	DL syntax	Manchester syntax
Top	$\top$	`owl:Thing`
Bottom	$\bot$	`owl:Nothing`
Concept	C	`C`
Role	R	`R`
Inverse role	R^-	`R`$^-$
Conjunction	$C \sqcap D$	`C` **`and`** `D`
Disjunction	$C \sqcup D$	`C` **`or`** `D`
Atomic negation	$\neg A$	`not A`
Unqualified existential restriction	$\exists R$	`R some owl:Thing`
Universal restriction	$\forall R.C$	`R only C`
Unqualified number restriction	$\geq nR$ $\leq nR$	`R min n` `R max n`
Definition axiom	$A \equiv C$	`Class:A EquivalentTo:C`
Inclusion axiom	$A \sqsubseteq C$	`Class:A SubClassOf:C`

Source: Ruta *et al.* (2017).

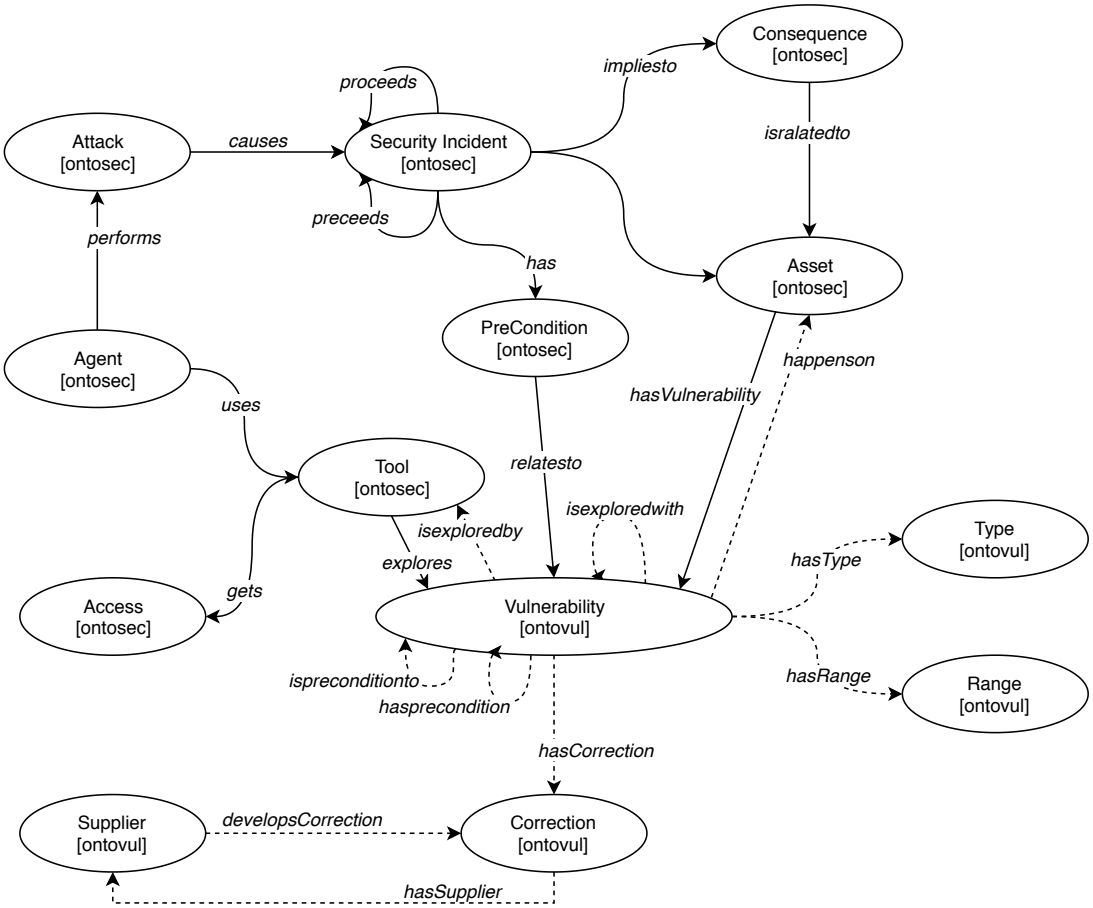
- Vulnerability
- SecurityIncident
- Consequence
- PreCondition
- Tool

For the construction of the proposed BIoT ontology, we considered related works in the Linked Open Vocabularies for the Internet of Things (LOV4IoT[6]) project. LOV4IoT references almost 500 ontologies that relate to an IoT applicable domain such as sensors and semantic web technologies and classifies these ontologies according to best practices as well. We also consider the work of Gyrard, Bonnet and Boudaoud (2014a) OntoSec (see Figure 30, highlighting main entities, concepts and relations of the OntoSec, from Martimiano (2006)), a Security ontology, Mozzaquatro, Jardim-Goncalves and Agostinho (2015), Mozzaquatro *et al.* (2016), a work based on OntoSec and extended to IoTSec, identifying many object concepts and properties for the proposal BIoT.

This work also has been inspired for an ontology of attacks and countermeasures for M2M communications by "The iotsec Ontology (Security Toolbox : Attacks and Countermeasures)" (GYRARD *et al.*, 2014; GYRARD; BONNET; BOUDAOUD, 2014b), cryptographic concepts and

[6] https://lov4iot.appspot.com/

Figure 30 – OntoSec: security ontology main concepts and relations.



Legend: Figure summarized is highlighting main entities, concepts and relations of the OntoSec.

Source: Adapted from Martimiano (2006).

security properties by "An ontology of Information Security" (HERZOG; SHAHMEHRI; DUMA, 2007), "Security in the Semantic Web using OWL" (DENKER; KAGAL; FININ, 2005), and "Security Ontology for Annotating resources" (KIM; LUO; KANG, 2005).

4.4.1 Classes

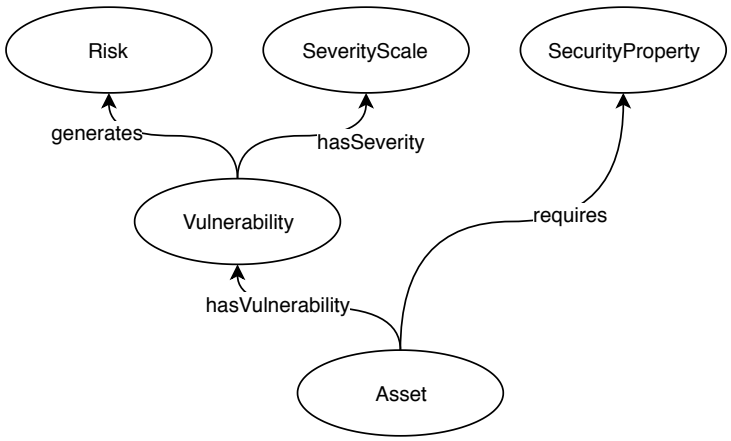
In this section, we describe some relevant classes, their relations, and axioms to ensure correct execution of relationship rules.

Asset

Figure 31 depicts highlighting the relationships between `Asset`, `Vulnerability`, `Risk`, `SeverityScale`, and `SecurityProperty` classes, from Gyrard *et al.* (2014), Gyrard, Bonnet and Boudaoud (2014b).

The Asset has Vulnerability that generates a Risk and has a SeverityScale. The Asset

Figure 31 – Asset class and its relations.



Legend: Figure summarized is highlighting main concepts and relations of the Asset class.

Source: Adapted from Gyrard *et al.* (2014), Gyrard, Bonnet and Boudaoud (2014b).

class is characterized by Architecture or type of operating technology, such as Zigbee, Bluetooth, Mesh, Ethernet, or RFID Technologies. All of these are SubClassOf Asset.

The core of the ontology represents the relationship that directly surrounds the Asset class and its relationships to Vulnerability and critical concepts such as Associated Risk and SeverityScale impacts.

Table 11 – Asset class and its axioms.

DL syntax	**Manchester syntax**
Asset $\sqsubseteq \exists$ hasVulnerability Vulnerability	Asset SubClassOf hasVulnerability some Vulnerability
Asset $\sqsubseteq \exists$ requires SecurityProperty	Asset SubClassOf requires some SecurityProperty
Asset $\sqsubseteq \neg$ SecurityProperty	Asset DisjointWith SecurityProperty
Asset $\sqsubseteq \neg$ SecurityMechanism	Asset DisjointWith SecurityMechanism
Asset $\sqsubseteq \neg$ Threat	Asset DisjointWith Threat
Asset $\sqsubseteq \neg$ Vulnerability	Asset DisjointWith Vulnerability

Source: The Author (2018).

Table 11 presents DL and Manchester syntax. In axioms for Asset means that Class Asset has some Vulnerability; Asset can require some SecurityProperty, Asset is DisjointWith SecurityPropert, SecurityMechanism, Threat and, Vulnerability classes.

From the above explanation of how to read DL and Manchester syntax, we can consider the same reading for some classes as follows:

4.4.2 Rules

All the main concepts used in the below axioms refer to the fundamental classes of BIoT. The inference machine makes deductions when using the ontological code. In this case, OWL DL has a relationship rule described through the axioms. So some of the following axioms make sure that the deductions were correct or valid.

The inference rules are processing in Semantic Web Rule Language (SWRL) with Protégé editor using the reasoner Pellet. The reasoner manipulates the logic of ontology to reason with individuals (see Table 12).

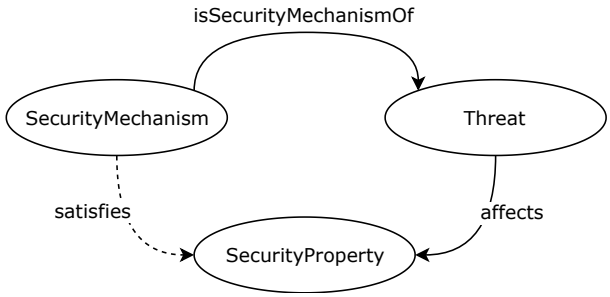
Table 12 – Inference rules.

R#	Inference rule
R1:	`hasPart(?x, ?z), hasPart(?z, ?y)` → `hasPart(?x, ?y)`
R2:	`isSecurityMechanismOf(?sm, ?t), threatens(?t, ?v)` → `mitigates(?sm, ?v)`
R3:	`SecurityMechanism(?sm), SecurityProperties(?sp), Threat(?t), affects(?t, ?sp), isSecurityMechanismOf(?sm, ?t)` → `satisfies(?sm, ?sp)`

Source: The Author (2018).

Figure 32 presents a graphical representation of the inference rule R3. The object property `isSecurityMechanismOf(?sm, ?t)` provides the ability to link `SecurityMechanism(?sm)` and `Threat(?t)` and `affects(?t, ?sp)` for linking between - `Threat(?t)` and `SecurityProperty(?sp)`, respectively. Then, this association enables one to discover implicit facts from structured knowledge in the object property `satisfies (?sm, ?sp)`.

Figure 32 – Representation of the inference rule R3.



Source: The Author (2018).

The reasoning provided the satisfactory result by providing a security mechanism against threats to the Zigbee protocol that addresses some instances of SecurityProperty class, ensuring Data Integrity, Confidentiality, and Authentication.

The proposed system consults the Knowledge Base Protocol through RDF Query (SPARQL) to identify attributes and individuals that refer to the appropriate security mechanism for the scenario of interest.

Listing 4.1 – Query to identify possible attacks on zigbee vulnerabilities.

```
SELECT ?ASSET ?VULNERABILITY ?THREAT ?SECURITYPROPERTY
?SECURITYMECHANISM ?FEATURE
WHERE
{ ?VULNERABILITY biot:isVulnerabilityOf ?ASSET .
  ?VULNERABILITY biot:isThreatensBy ?THREAT .
  ?THREAT biot:affects ?SECURITYPROPERTY .
  ?SECURITYMECHANISM biot:isSecurityMechanismOf ?THREAT .
  ?SECURITYMECHANISM biot:hasFeature ?FEATURE .
  ?SECURITYMECHANISM rdfs:label ?SMLabel .
  FILTER regex (?SMLabel, 'ZIGBEE') }
```

Listing 4.1 shows a SPARQL query used to perform class associations ASSET, VULNERABILITY, THREAT, SECURITYPROPERTY, SECURITYMECHANISM, FEATURE. As soon as a security incident is identified, the system queries the ontology knowledge base and reports the vulnerability as Unauthorized Access, which affects some SecurityProperty. Thus, many vulnerabilities can be exploited and incorporated over time into the knowledge base.

The proposed security middleware uses IoTSec and OntoSec ontology concepts. It one can find better solutions and services according to identified security alerts. The reasoning capabilities determine data instance correctness and assertiveness using rules. This process derives from implicit facts of existing knowledge within a context – the reasoner based on rules or inductive and deductive reasoning. Some ontology verification processes occur as a result of reasoning:

- Check ontology consistency and knowledge base;
- Also, check the relations of intention between the classes;
- Sort instances in classes.

The reasoner manipulates ontological logic using inference rules for reasoning with individuals. Also, it shapes the user-defined data types and debugging support for BIoT ontology.

4.4.3 Architectural Design Pattern

Nowadays, we can not ignore other technologies for description that favor consistency, maintenance, and best practices for all IoT applications and devices. Web APIs aim to balance a RESTful API interface with positive developer experience.

With the increase in the availability of embedded devices, especially with the growth of sensors that support the IP protocol, in addition to the growth in the availability of Internet networks (3G, 4G, 4.5G . . .), and even operators dedicated to IoT, continuous increase of Internet bandwidth and reduction in prices favor the development of large-scale devices.

The many data interfaces between sensors is a problem that has arisen, in sensor capture, processing and transfer between services, with storage in several databases demonstrate a need and attempt to unify through a RESTful API.

Currently, there are two streams of Web Services development: SOAP (Simple Object Access Protocol) part of the use of XML to transfer data or objects between applications and REpresentational State Transfer (REST), which can currently operate through the most popular JSON or XML.

We highlight SOAP in the early 2000s, the SOAP protocol was considered as a W3C recommendation for Web Services development, considered the standard widely implemented at the time, leaving legacy systems and integrations that persist to this day.

REST was developed from the HTTP 1.1 protocol and, unlike SOAP, which aims to establish a protocol for communication between objects and services, proposed the proper use of HTTP verbs (GET, POST, PUT, HEAD, OPTIONS and DELETE) to create services that could be accessed by any legacy or current device or system.

Therefore, REST is not just a communication protocol but is considered as an Architectural Design Pattern, for any services or devices that may be exposed through HTTP protocol.

Many sensors are IP-enabled, which enables the ability of these sensors to be programmed to provide RESTful communication.

4.4.4 Pragmatic REST

These guidelines aim to support a truly RESTful API. Here are a few exceptions:

Put the version number of the API in the URL (see examples below). Do not accept any requests that do not specify a version number.

Allow users to request formats like JSON or XML like this:

http://app-example.com/api/v1/biot.json

http://app-example.com/api/v1/biot.xml

4.4.5 RESTful URLs

General guidelines for RESTful URLs

- A URL identifies a resource.
- URLs should include nouns, not verbs.
- Use plural nouns only for consistency (no singular nouns).
- Use HTTP verbs (GET, POST, PUT, DELETE) to operate on the collections and elements.
- You should not need to go deeper than resource/identifier/resource.

- Put the version number at the base of URL, for example, http://app-example.com/v1/path/to/resource.
- URL v. header:
  - If it changes the logic you write to handle the response, put it in the URL.
  - If it does not change the logic for each response, like OAuth info, put it in the header.
- Specify optional fields in a comma-separated list.
- Formats should be in the form of api/v2/resource/id.json

We compiled the blockchain structure as a Docker image to create the scenarios for conducting the experiments; We executed each node as a container of the composed image; We used the Docker API SDK to manage all test execution, detailed in Section 5.2.

4.5 BIOT TESTBED ENVIRONMENT

We have broken the built environment and evaluation by concomitant moments of the device in operation to identify and extract concepts and properties, and we have used *Protégé* to formalize these extracted conceptualizations and check inconsistencies during the construction of ontology proposal.

4.5.1 Model of context for BIoT

We describe an overview of general usage scenarios. To better the realize we assume that in the whole life-cycle of the development of an ontology or their parts, there are different moments or phases. These moments are called offline and online phases, the former refers to ontology design maintenance, and the latter implicates the use of the ontology for problem-solving tasks.

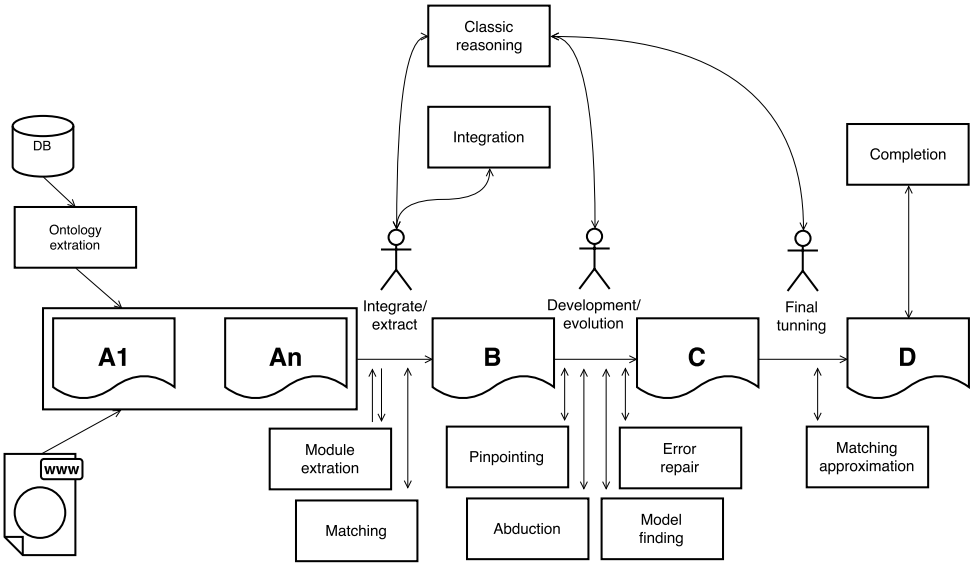
First, we present a scenario that illustrates the interplay of various online and offline tasks. Tones-d *et al.* (2007), Hart *et al.* (2004), Noy and McGuinness (2001) present in a structured and systematic way some associated techniques mentioned in the literature about the design of ontologies, as shown in Figure 33.

4.5.1.1 Offline usage scenarios

Tones-d *et al.* (2007) describes how to ontology design can be performed using different sources:

1. existing ontologies;
2. existing databases schemas and the databases instances;
3. additional sources from the Web;
4. building ontologies from scratch;

Figure 33 – General usage scenario for ontology design.



Source: Tones-d *et al.* (2007).

4.5.1.2 Online usage scenarios

Online usage scenarios to ontology building process are oriented towards through the view of “runtime ontology” or “ontology at work”, and for online usage scenarios, runtimes or the ontology works on study field are even more critical than for offline scenarios.

The terms ABox and TBox are used to describe two different kinds of statements in ontologies. ABOX is an *assertion component* — a fact associated with terminological vocabulary within a knowledge base. TBOX describes a system regarding controlled vocabularies, e.g., a set of classes and properties (see Figure 34).

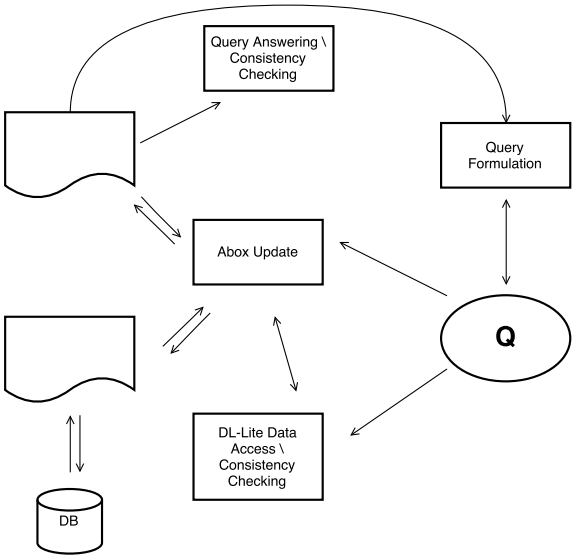
4.5.2 Device runtime environment

Figure 28, p. 91, presents the main concerns of the project, to maintain the integrity of the smart entities and control entities, as well as between the physical entities associations and features, software agents, applications, and services. The layer of IoT-ontology and namespace are listed and described in the page directories, used to organized and prevented conflicts with other pages with the same name or same conceptualization.

The proposed architecture was built considering four layers, as shown in Figure 35:

1. **the perception layer**: responsible for data collection of flow sensors and, it sends data to wireless sensor network (WSN) nodes;
2. **the communication layer**: receive the data from the sensors and, transmit them through a network of WSN nodes, using the ZigBee modules, to the concentrators that will send the data further to a data collector;
3. **the middleware layer**: the data received from the communication layer are then checked

Figure 34 – General usage scenario for ontology access.



Source: Tones-d *et al.* (2007).

and, stored data into a database;

4. **the application layer**: business rules are applied, after all.

We built formerly for identification of the involved entities in sensing and data collection concerning the water consumption we develop the forerunner prototype Local Water Control (LWC) has just two layers: *i)* the perception layer and *ii)* the communication layer, as well as specifications of the XBee network access control topologies: point-to-point, star, tree, cluster and mesh.

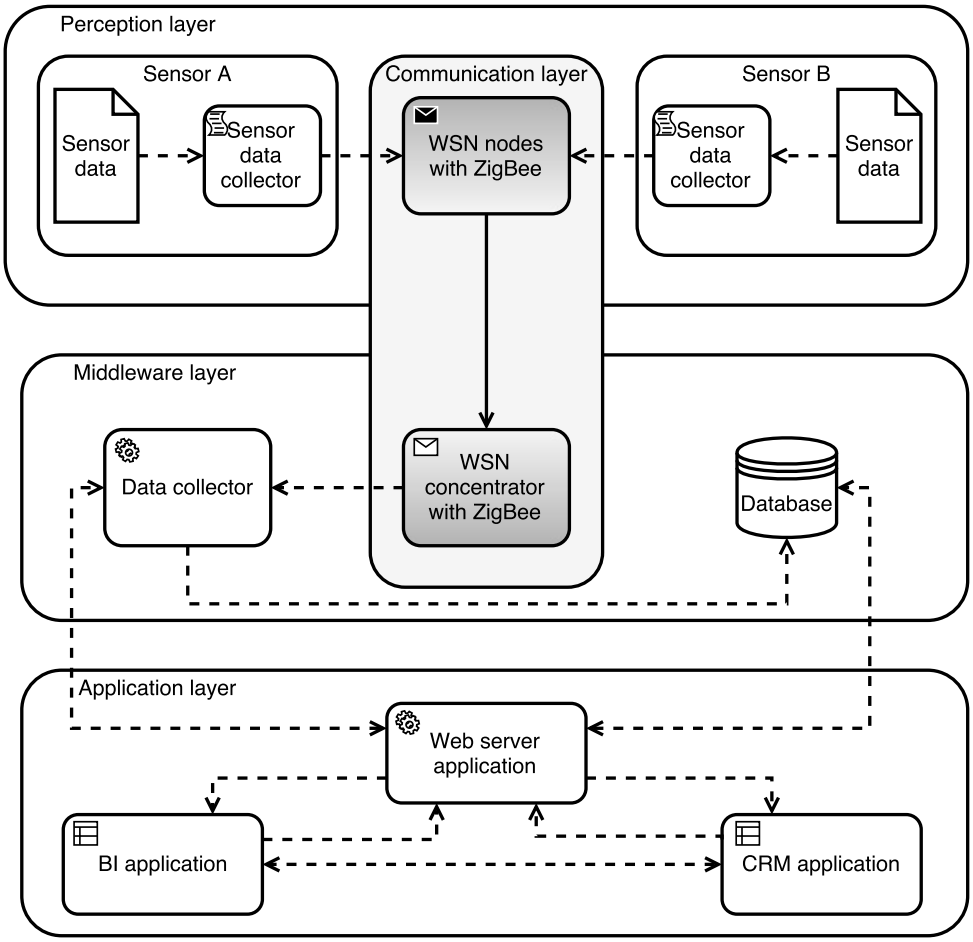
The Local Water Control application has developed and tested with Arduino Mega 2560, in charge of the communication between the water flow sensor and data communication.

4.5.2.1 Perception layer

The perception layer is in charge of collecting data related to water consumption by a water flow sensor. The water flow sensor is a plastic body, a rotor, and a hall-effect sensor. The water flows through the rotor that rotates and generates frequency pulses detected by the half-effect sensor (flow rate range: 1∼30L/min), with this structure it is possible to know the flow rate and, therefore, the average of the domestic water consumption (real-time) in each residence. This layer captures data by flow sensors. The proposed study do aim to consider the flow rate analytics.

Data acquisition can be described as a process for the perception of physical phenomena and their transformation into electric-digital signals for later transmission and processing. The basic constitution of an acquisition layer is given by a) sensor module; b) module translator and data collector; and, c) transmission module. In this work the acquisition layer is composed, par-

Figure 35 – Local Water Control Architecture has four layer. They are 1) perception; 2) communication; 3) middleware; and 4) application layer.



Source: Mendonça *et al.* (2016).

ticularly considering the LWC architecture, by: a) sensor module, responsible for capturing water flow data; b) micro-controller module, which translates the sensed data into digital information and performs basic local processing; to finally send to the, c) ZigBee module, responsible for transmitting (sending and receiving) the data. Within the acquisition layer, the data collected by the sensors are loaded into the communication layer.

For the scenario considered in this work, the sensors are responsible for measuring the flow of water. The acquisition layer is responsible for capturing consumption data, using a water flow sensor. The water flow sensor consists of a plastic body, a rotor, and a Hall effect sensor. Hall effect sensor: it is a transducer that when applied under a magnetic field, responds with a change in the output voltage (see Figure 36).

The flowing water, by rotating the rotor produces a frequency of pulses that are delivered by the Hall effect sensor, it is then possible to know the flow, and consequently the water consumption, as can seen in Figure 36, specifications in Table 13, and a sample connection on Listing 4.1. Mendonça *et al.* (2016) present the Listing 4.2 was installed on an Arduino Mega

Figure 36 – Water Flow Module with Hall Effect Sensor.

Source: The Author

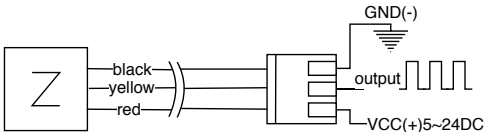
Table 13 – Water Flow Module Specification.

Characteristic	Value
Model	YF-S201
Working Voltage	5V∼24V
Water Pressure	≤2.0MPa
Flow Rate Range	1∼60L/min
Storage Temperature	$-25 \sim +80\,^{\circ}\mathrm{C}$

Source: Documentation of Water Flow Sensor. Available online: http://wiki.seeed.cc. Access date: 3 nov. 2015.

2560, and it was used to control the sensor module can be found as follows:

Figure 37 – Water Flow Sensor Wiring Diagram.



Source: Documentation of Water Flow Sensor. Available online: http://wiki.seeed.cc. Access date: 3 nov. 2015.

4.5.2.2 Communication layer

Communication layer, in the LWC architecture, is composed of data transmission modules via a wireless network. ZigBee communication modules implanted in the sensor modules, and the data collection module were used.

Zigbee technology is a low-cost, two-way wireless communication standard, meaning that data can travel from sensors to the application, and actuation commands can be sent back to the sensors. It has a communication range that varies from 75 meters up to a few kilometers and can build networks formed by up to 65,536 devices in a star, tree, or mesh network topology. ZigBee devices can have different functions in the network communication model, can assume the role of coordinator, network node, or end-device. In assuming the role of coordinator, the device will have the function of starting and maintaining the network in operation, besides acting

Listing 4.1 – Reading liquid flow rate using ZigBee and Water Flow Sensor (sample code).

```
// Reading liquid flow rate using ZigBee and Water Flow Sensor
// Code adapted by Silva Júnior, J. F.

#include "SoftwareSerial.h"
volatile int NbTopsFan; //measuring the rising edges of the signal
const int hallsensor = 2; //The pin location of the sensor
float waterFlow = 0;

void rpm () { //This is the function that the interupt calls
  NbTopsFan++; //This function measures the rising and falling edge of the
  hall effect sensors signal
}
// The setup() method runs once, when the sketch starts
void setup() { //
  pinMode(hallsensor, INPUT); //initializes digital pin 2 as an input
  initialised,
  attachInterrupt(0, rpm, RISING); //and the interrupt is attached
  Serial.begin(9600); //Setup function where the serial port is
}
// the loop() method runs over and over again,
// as long as the Arduino has power
void loop () {
  NbTopsFan = 0; //Set NbTops to 0 ready for calculations
  sei(); //Enables interrupts
  delay (1000); //Wait 1 second
  cli(); //Disable interrupts
  //(Pulse frequency x 60) / 5.5Q, = flow rate
  waterFlow = (NbTopsFan * 60 / 5.5);
  in L/hour
  Serial.print (waterFlow, DEC); //Prints the number calculated above
  Serial.print (" L/hour\r\n"); //Prints "L/hour" and returns a new line
}
```

as a bridge between distinct networks assuming the role of the router, and the role of the terminal device is represented by the sensor devices (RAMOS, 2010).

Zheng and Zhang (2011) present some advantages of the IoT wireless sensor network, where various sensors collect data. The data collected is then transferred to a server by GPRS-

Listing 4.2 – BIoT ASensor (Node) on Arduino Mega 2560 (MENDONÇA *et al.*, 2016).

```
# Algorithm ``ASensor''

var
    signalSensor, pinSensor, RX, TX: integer
    waterFlow: real

begin
    # Pin setup
    pinSensor <- 2 # Sensor reading Pin
    RX <- 2 # XBee Pin RX
    TX <- 3 # XBee Pin TX
    # Starting variables
    waterFlow <- 0 # Storage the water flow
    # Randon value for simulation
    SignalSensor <- 55
    # Receives the sensor signal calculates the flow
    waterFlow <- signalSensor * 60 / 5.5
    # Print the output of the XBee gross
    # Print the output of the XBee cubic footage
    print(``ASensor: '', waterFlow)
end
```

DTU using a GPRS communication network and stored in a remote terminal unit (RTU). The ZigBee is a low-cost, two-way, wireless communication standard. Its operating rate varies from usual 75m to even a few kilometers, presents low power consumption requirements. The ZigBee communication module, in miniaturization and low power consumption, provides the basis for the Internet of Things, with an open communication protocol IEEE 802.15.4 built for industrial, scientific and medical environments. It meet all required features for this design, as excellent immunity to interference, high reliability (packet delivery guarantee, even in the event of data corruption), low power consumption, good range (even without sight), native encryption of 128-bit, up to 65,536 nodes in the network, and cost/benefit. The ZigBee module allows the availability of network topologies in a tree, star or meshes topologies. Note that the choice of topology can suit the operation and architecture requirements.

One of the characteristics of the ZigBee devices is the low electrical consumption as against other devices, in part, made possible by the operating mode, being able to switch from idle, or idle, to act in less than 30ms. The ZigBee device remains idle until it identifies an available data packet for transmission. To minimize electrical consumption and increase communication efficiency, the ZigBee device can assume 6 states in a cyclic model:

1. inactive;
2. active; which can assume the sub-states of:

- transmission;
- reception
- command; to return to the state,

3. idle.

In the idle state, the device performs only a signal check on the antenna to know if there is a request for transmission; at that moment, the device does not transmit or receive data. Unlike the idle state, in the idle state, the ZigBee device is not performing any actions. It is the most energy-saving mode and is the advantage presented by this device over other wireless data transmission modules such as WiFi or Bluetooth. In ZigBee, the state change occurs in less than 30ms.

Mendonça *et al.* (2016) present the Coordinator that is responsible for initiating and maintaining the network connections, and also to act as a bridge between different networks. The Router works as a node and as a bridge between other network nodes. Finally, the device hosts receive data from the sensors and send the data for processing and transmitting directly to the actuators. The Listing 4.3 was installed on an Arduino Mega 2560, and can be observed as follows:

When the device detects radio signals on the antenna, it enters the active state and checks for the validity of the packet and address for communication, enabling modes of transmission, reception or command. When switching to the active mode the device can assume the sub-states of a) transmission; b) reception; and, c) command.

The transmission mode starts with the verification of the destination address, if the address is not known, a search will be done for a valid address, which when discovered will be used in search of the destination route if the ZigBee device can not find an address valid or a communication route the packet will be discarded, and it will return to the idle state, otherwise the transmission of the data will be made and again the device returns to the idle state waiting for new data transmission. This strategy allows saving electricity on the device, as there will only be data transmission for valid address and route.

The receive mode is assumed when valid data is perceived, which are then transferred to the transmission. Command mode is for the sole purpose of sending and receiving configuration commands between ZigBee devices.

4.5.2.3 Application layer

The application layer consists of a Web Server Application running the role of a subscriber, or listener, of the MQTT protocol. This module is also responsible for retrieving the data stored in the DBMS to be presented to the application modules for analysis and decision making, considering the LWC architecture, presenting the data in a Business Intelligence (BI) application and CRM Application. Figure 24, p. 83, shows a representation of the application layer of the

Listing 4.3 – BIoT Coordinator/Router on Arduino Mega 2560 (MENDONÇA *et al.*, 2016).

```
Algorithm ``Coordinator/Router''

var
    mac, IPAddress: character
    analogInPin, analogOutPin, sensorValue, EthernetServer,
    I: integer

begin
    # Network setup
    mac <- ``0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED''
    IPAddress <- ``192.168.25.201''
    EthernetServer <- 80
    # Pin setup
    analogInPin <- 0
    analogOutPin <- 9
    # Starting variables
    sensorValue <- 55
    i <- 0
    # Print data
    while sensorValue <> 0 do
        clear
        # Starting Ethernet service
        print()
        print()
        timer 500
        repeat
            # Print data received by XBee
            print(``ASensor: '', sensorValue,``  m^3/hour'')
            i <- i + 1
        until i = 10
    endwhile
end
```

LWC architecture.

The layer is composed of an intelligent agent, incorporated into the Web Server Application, responsible for monitoring data from the middleware layer, enabling the issuance of pre-configured alerts in case of the occurrence of disturbances in the water distribution network.

The BI module was developed using the QlikView Personal Edition platform, a free, personal-oriented version of the QlikView data visualization platform developed by Qlik which provides for free, and for personal use, a tool aimed at the rapid creation of guided analytical applications and data display in the scorecard. In the BI module, a panel of indicators is displayed, containing the data collected by all the sensors, showing quickly and clearly the variation in water demand and consumption. The panels presented in module BI of the developed prototype are:

- chart of measurements per day;
- chart of measurements per hour;
- sensor selector;
- selector for the month, day and time; and,
- a grouper of the selected options.

4.6 CONCLUDING REMARKS

IoT networks must be data-centric, data is sent across a large number of devices, these devices are usually not configured with adequate security level or updated, a motivation for potential attacks on the IoT network.

Blockchains are designed to run on heterogeneous P2P networks. We must note that IOT end devices have minimal features compared to more robust devices such as high-performance application servers. We present below some of these limitations, observed in this work.

The data are recorded in Blockchain, through persistent and unchanged data, thus ensuring integrity; and with data persisted by encrypted data (to be applied) by symmetric keys (shorter validation time) or asymmetric keys (longer validation time).

Our proposal is focused on the addition of the Middleware layer, which may belong to the application layer since the network layer does not present significant physical constraints. The structure of the BIoT ontology aims to establish a body of knowledge about the basic principles of the operation of a Blockchain network, to implement secure data recording from the IoT devices, data encryption rules (or policies) and Blockchain integrated with a typical IoT architecture.

Requirements of the Middleware for BIoT architecture were divided into two viewpoints, the application point of view and the middleware point of view. Next, we establish the Middleware layer with its key features, object-oriented, message, and transactions, because we consider a

heterogeneous and distributed network environment. Features were developed to solve security issues, about (a) the integrity, data must travel throughout the structure without undergoing modification in content; (b) privacy, data shall be transmitted and stored so that it is not legible by unauthorized means; (c) availability, access to the data must be requested by those who are duly authorized; and, (d) reliability, the data must represent what was initially collected exactly at its origin.

To validate the proposal, we apply testbed concepts with application in the level of controlled simulation in bench, as can be observed in Section 4.5. Particularly considering the WLC architecture, it is composed of a) collector module, responsible for receiving data from the various sensors; and send them to one, b) database manager system (DBMS), responsible for managing the persistence of application data.

Finally, we have established some key strengths and key elements of BIoT, describing their fundamental entities, class diagrams (overview), pseudo-codes and general entities, as well as some value propositions and their respective slots, see Table 9, p. 93. We also propose a Web API based on REST, using the main HTTP verbs (GET, POST, PUT, DELETE) to access features and verify in the browser, confirmation of transactions, obtaining performance reports and critical validation.

5 EVALUATION OF THE BIOT

This chapter presents evaluation (step 3 of the research approach — see Chapter 3, Figure 17, p. 17) of BIoT ontology. Next, we describe the of context for usage scenarios. Then, we present performance evaluations to scalability, defense effects, accuracy, and efficiency of BIoT. Finally, we present the survey planning and expert evaluation of the characteristics and elements of the proposed BIoT and the applicability in projects.

5.1 EVALUATION OF BIOT ONTOLOGY

The tool that contributed the most was the *Pellet Reasoner*. At the end of the tests, we can confirm the criteria through the automated tests available at the *Dr. PerfectO*[1] project. Some tools, like *OOPS!* enable integration with our development and tools through RESTFul Web Service by just including XML code as on Listing 5.1 and using an HTTP POST[2] request.

Listing 5.1 – *OOPS!* XML code in BIoT API test tool integration.

```
<?xml version="1.0" encoding="UTF-8"?>
<OOPSRequest>
        <OntologyURI>http://www.cc.uah.es/ie/ont/learning-
           resources.owl</OntologyURI>
        <OntologyContent></OntologyContent>
        <Pitfalls>10</Pitfalls>
        <OutputFormat></OutputFormat>
</OOPSRequest>
```

When testing with the tools presented, most of the relevant bug fixes were made. The *Pellet* and *OOPS!* tools returned significant results. The tool that received the most attention in the final analysis due to fault identification and pointed improvements was the *OOPS!*. The mechanisms identified 15 cases of developments, affecting 48 classes one or more times, about 120 corrections in the proposed ontology. After syntactic validation, some proposed recommendations to clarify the meaning of “Vocabulary” in this ontology context.

In this context, a vocabulary is synonymous 2 of ontology. However, we differentiate vocabulary from an ontology by characteristics enabling reuse and integration by other vocabularies:

- Small size
- Low formal constraints (basically RDFS and a fistful of OWL)

[1] http://perfectsemanticweb.appspot.com
[2] http://oops-ws.oeg-upm.net/rest

- Few instances except for examples
- Rich user documentation (Labels, comments, definition, description, etc.)

By linking and reusing each other, vocabularies contribute to the growth of an awesome ecosystem: “The Linked Open Vocabularies”, by following the summarized recommendations below:

- Vocabulary Metadata
  - Identification
  - Title and description
  - Version and modification
  - Rights and property
- Vocabulary Elements (Classes and Properties)
- Documentation

Our considerations. We have fixed about 80% critical and relevant repairing needs. After corrections, the reasoner was able to infer many cases by uncovering the relationship. These relations were between Assets, Vulnerabilities, Threats, Security Properties, and Security Mechanisms. For this release, We ignored other suggestions because these proposals were for aesthetic repairs or partially addressed knowledge. As a result, the ontology provides a reliable security mechanism. It prevents threats to sensor access protocols. It also satisfies the monitoring of instances of the SecurityProperty class. Admittedly, it ensures the protection of authentication, confidentiality, and data integrity. The inference rules classify any attempted change on data in the blockchain network. Succeeding, the Security Mechanism resists attempts to tamper with or alter data.

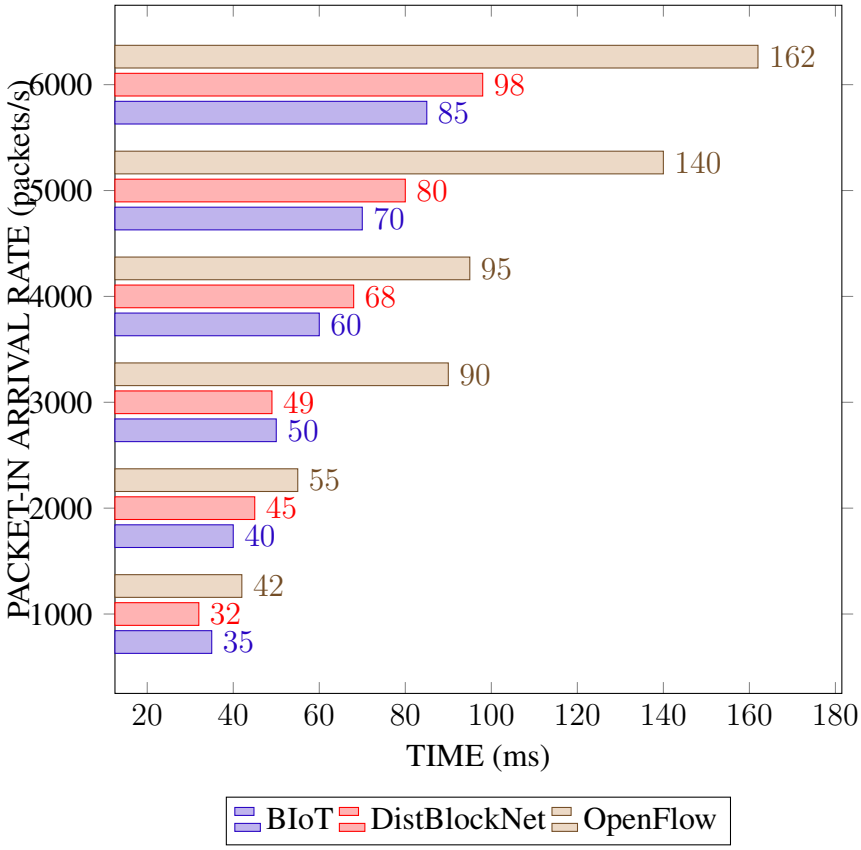
5.2 SANDBOX FOR PERFORMANCE TESTS

In this section, we describe the statistics collected from the sample mean—sum of all observations and dividing this sum by the number of views (this test case: 15 observations) in the sample. We performed a performance evaluations to evaluate scalability, defense effects, accuracy, and efficiency of our BIoT. BIoT evaluation were performed against OpenFlow SDN and DistBlockNet performance evaluation . (SHARMA *et al.*, 2017; RUTA *et al.*, 2017).

5.2.1 Scalability evaluation

Figure 38 shows the result of the flow rules table update time about to the packet-in arrival rate in both the BIoT and OpenFlow SDN, and DistBlockNet. In this experimental result, we observed that our proposed BIoT regularly performed superior to the distributed OpenFlow SDN, and DistBlockNet networks as the rate of the packet-in arrival increased, considered for time (abscissa axis) as “Lower is Better (LB)”. (JAIN, 1990).

Figure 38 – Flow table update time vs. packet-in arrival rate BIoT against OpenFlow SDN, and DistBlockNet.



Source: The Author (2018).

5.2.2 Defense effects

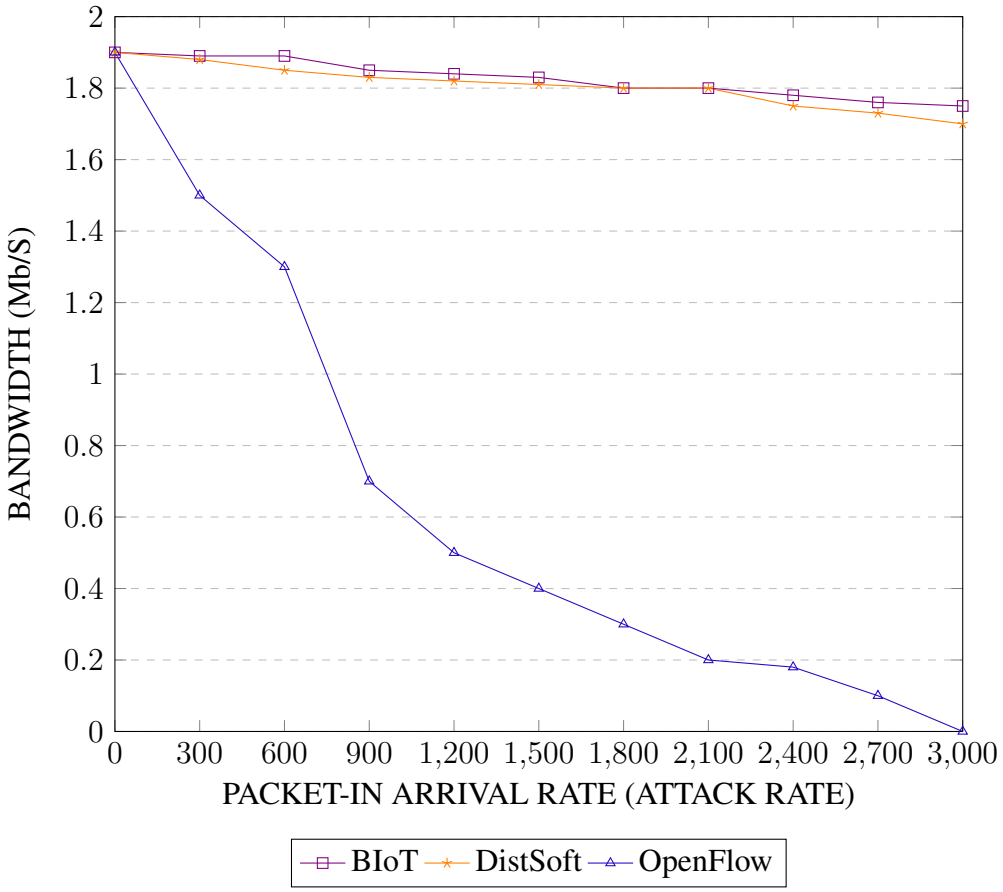
We evaluated the BIoT on both OpenFlow and DistBlockNet and found that the built networks proved to be quite unstable, increasing the bandwidth consumption of the tested networks (both OpenFlow SDN and DistBlockNet evaluated test objectives), as seen in Figure 39, considered for bandwidth availability as “Higher is Better” (HB). (JAIN, 1990).

BIoT presented stability due to the proposed middleware architecture, which identifies the node at the beginning of the packet transactions through symmetric cryptographic key, each node keeping its private key; controller/coordinator nodes keep the public key of each other client node, hashing hash sums on controller/coordinator nodes, as seen in Figure 40.

5.2.3 Computing tasks

In this section, we describe the statistics collected (this test case: 15 observations) from the metrics: average processing time for computing tasks on a given request; average turnaround time for accomplishing requests; average Docker container memory usage per node; and average hit ratio per node.

Figure 39 – Effects on bandwidth during different attack rate in the software environment (simulated environment)



Source: The Author (2018).

5.2.3.1 Time

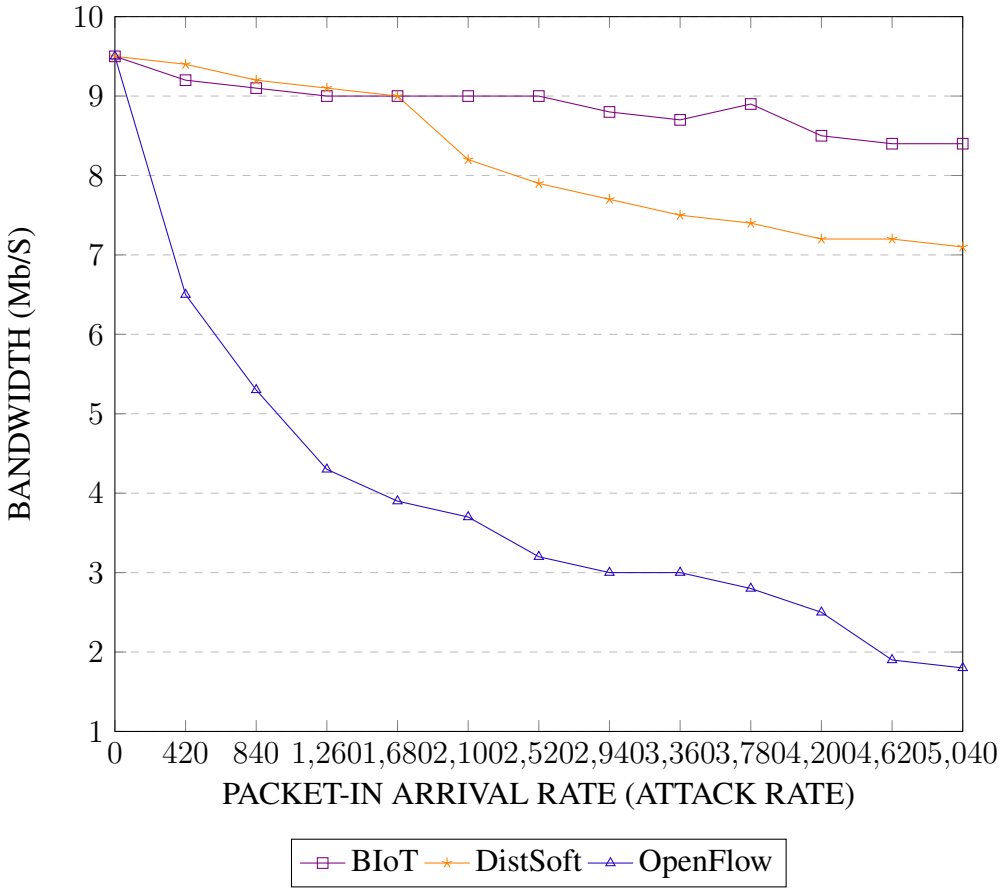
Figure 41 and 42 show lower differences in the turnaround time averages between 10 and 50 nodes. The values obtained can be considered as very low with small and medium scenarios, while the scenario of 150 nodes turnaround time reached the timeouts at 2 and 6 seconds. However, explanation mode on, there was an increase in the average processing time for the task by ten nodes.

Figure 41 and 42 present the same proportionality in other scenarios except, in 150 nodes scenario. It ones was a significant difference in processing for the offline explanations, considered for processing time as “Lower is Better” (LB). (JAIN, 1990).

The increasing order of timeout processing occurred in order, 2, 10, and 6s (see Figure 43). Discovery task for explanations mode on was proportional for increasing timeout.

Processing times tend to increase at higher scales due to the needed consensus about computation results among a more significant number of nodes. Furthermore, in all the experiments, the time of discovery process dominates explanation and selection. This result was expected, as

Figure 40 – Effects on bandwidth during different attack rate in hardware environment (OpenFlow SDN, MikroTik with MikroFlow environment).



Source: The Author (2018).

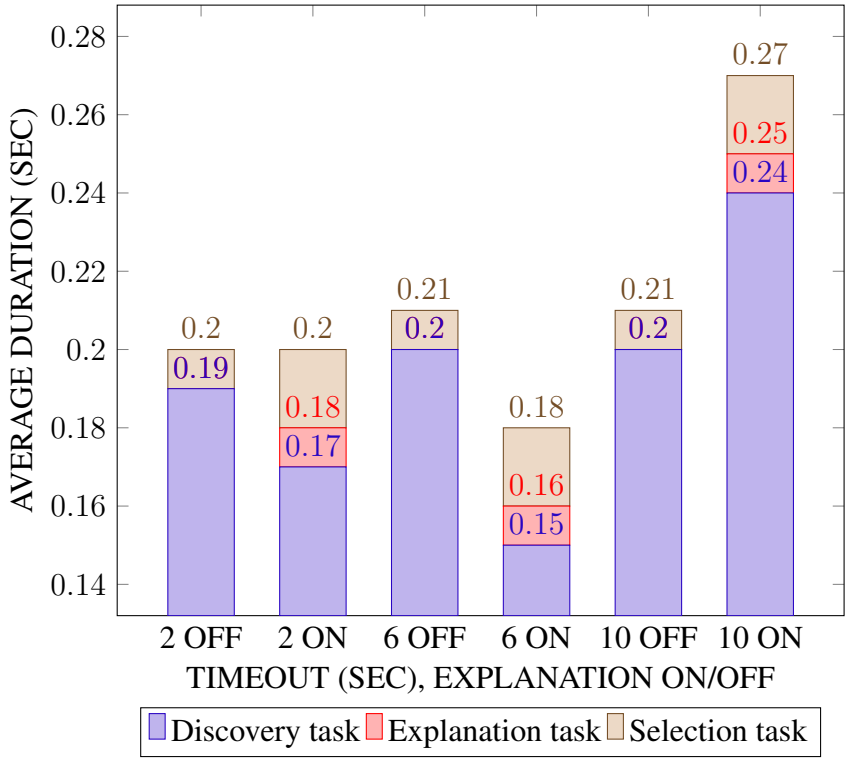
semantic matchmaking is the most computationally intensive task, even if performed with an optimized reasoning engine. (LAURENCE, 2019).

5.2.3.2 Memory

Usage of average and maximum RAM showed a decreasing trend for an increasing number of nodes among scenarios. It tends to stabilize with more significant amounts of nodes, and memory demand shows a slight increase in usage (see Figure 44 and 45), considered for memory use as "Lower is Better" (LB). (JAIN, 1990).

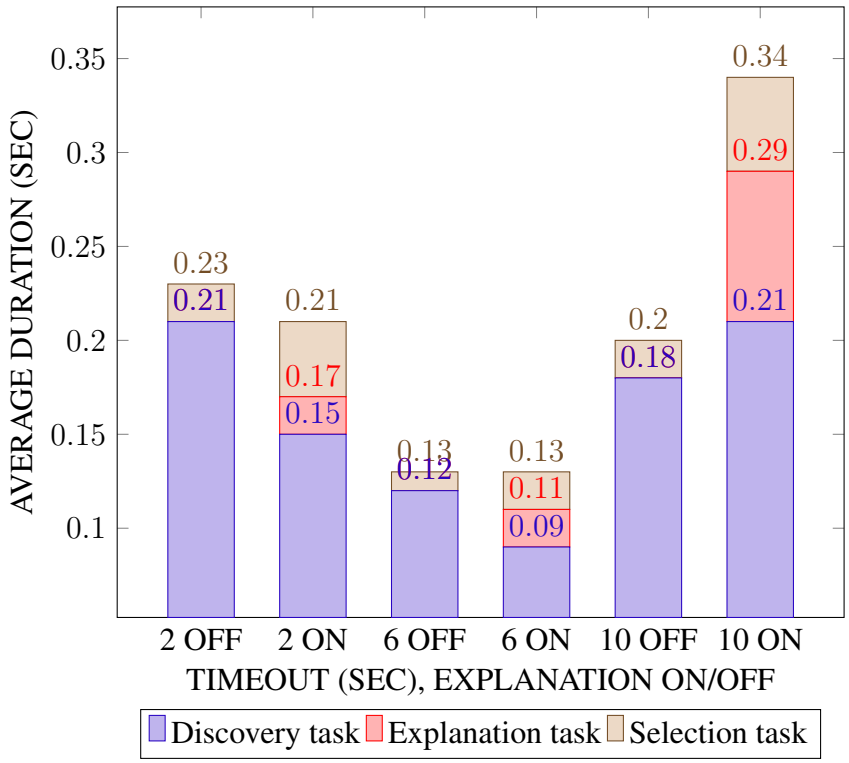
These results are expected due to Containers being a form of operating-system-level virtualization that allows you to run multiple isolated systems on a single real operating system. These isolated systems can be effectively separated from the protection of containers and limited in both disk usage, RAM and CPU, and only use the host's required, significantly reducing disk space usage. (CHAE; LEE; LEE, 2019).

Figure 41 – Processing time for tasks on 10 nodes.



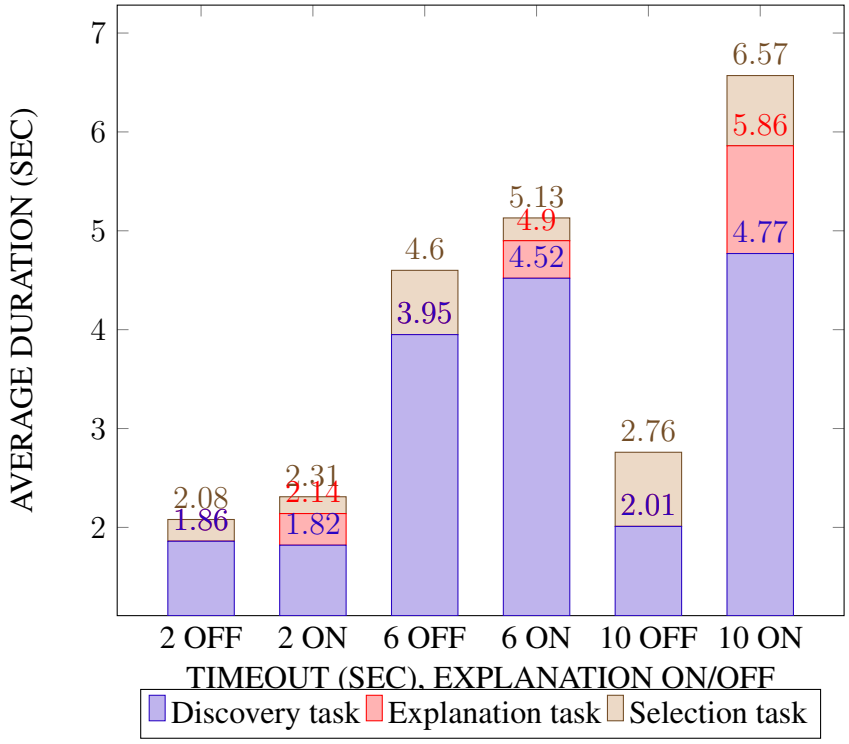
Source: The Author (2018).

Figure 42 – Processing time for tasks on 50 nodes.



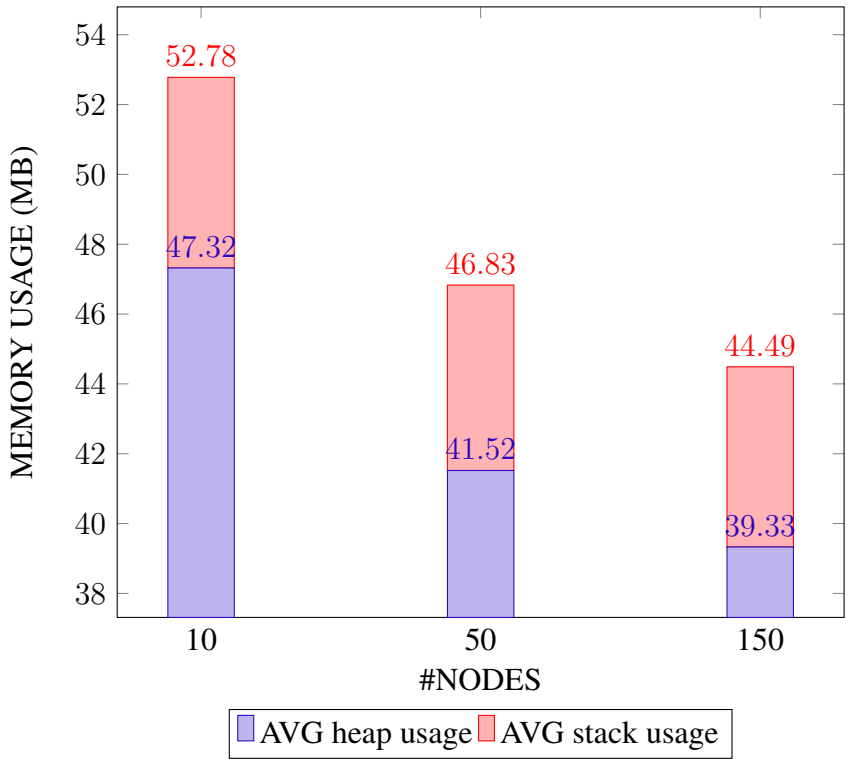
Source: The Author (2018).

Figure 43 – Processing time for tasks on 150 nodes.



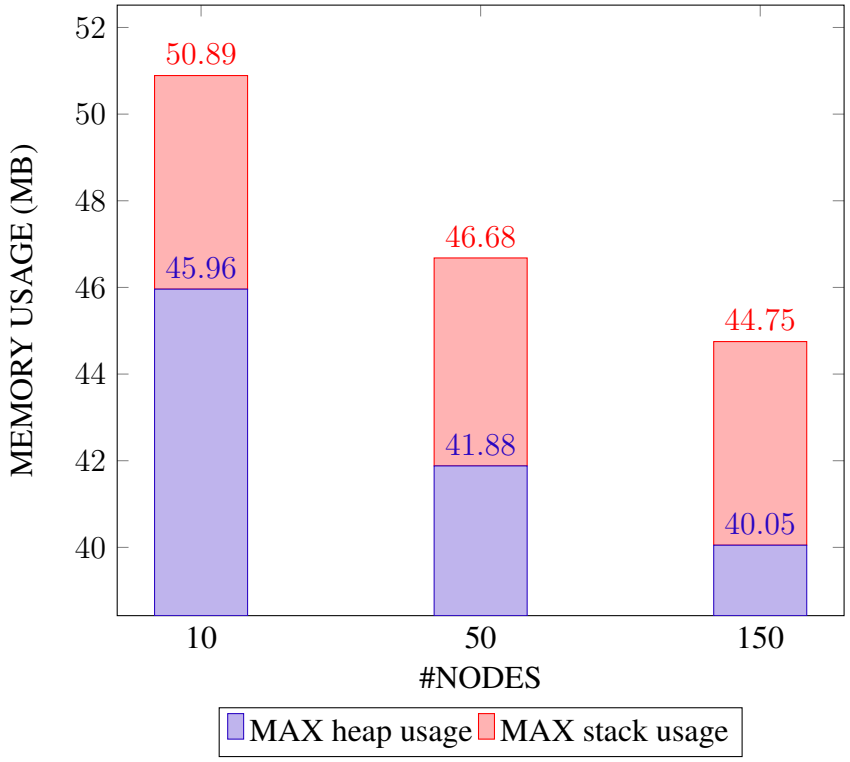
Source: The Author (2018).

Figure 44 – Virtual machine average memory usage per node.



Source: The Author (2018).

Figure 45 – Virtual machine maximum memory usage per node.



Source: The Author (2018).

5.2.3.3 Hit ratio

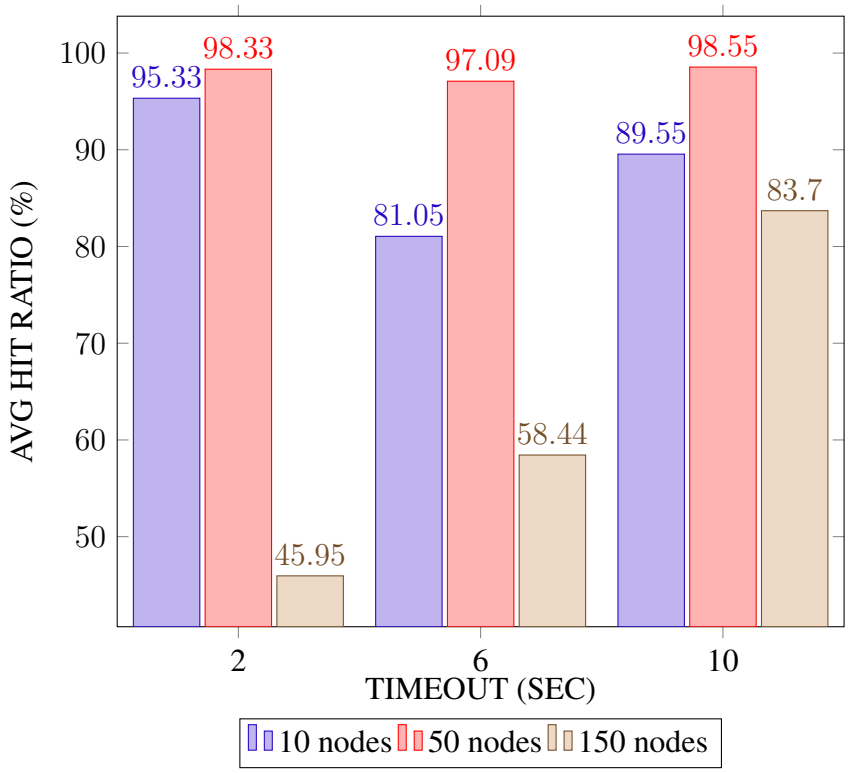
We defined Hit ratio as the percentage of requests which retrieve on resource satisfying both node status and semantic relevance constraints within the given timeout, considered for average hit ratio timeout as “Lower is Better” (LB). (JAIN, 1990).

Figure 46 presents average hit ratio directly related to the number of nodes. Scenario 10 nodes had a lower average hit ratio than 50 nodes. The 150 node scenario has the best performance.

In part, we get a better handle on Docker’s orchestration. It can also be related to the best path issues chosen in the graph, due to the correct ontology inference. Other factors that can be associated with the computational architecture infrastructure.

These results compounded overhead with the inherent complexity of consensus algorithms, leading to the turnaround time issues already discussed. Experimental results show that the approach of small and medium scenarios is more effective for applying permitted blockchain, and this provides better sizing of solutions.

Figure 46 – Average hit ratio depending to timeout.



Source: The Author (2018).

5.3 SURVEY BASED ON EXPERT VIEW

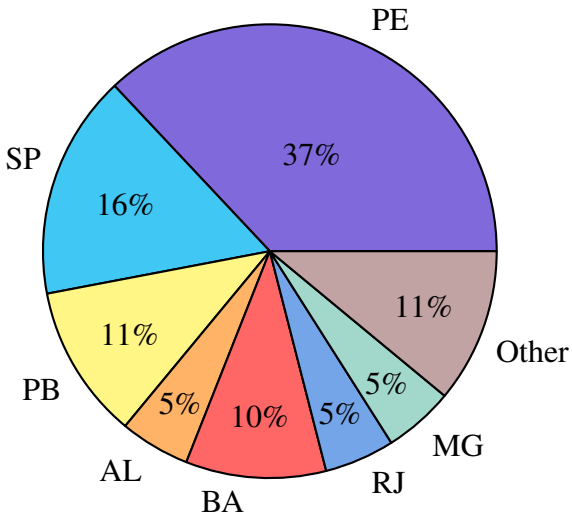
5.3.1 Results of the survey

This section presents the analysis of the data collected in the research, discussing in details the main questions and pointing some correlation points that must be taken into consideration.

5.3.2 Characterization of the experts

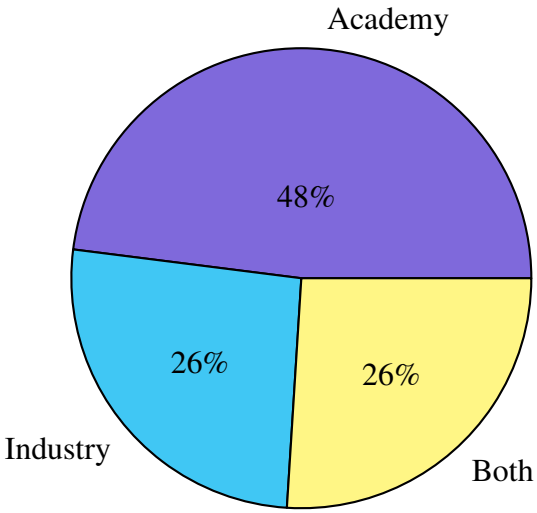
Most of the nineteen specialists that participated in the study are resident in Brazil, with predominance on the state of Pernambuco (see Figure 47). Two participants were out of the country (the United Kingdom and Canada), labeled Others. Still, about the characterization of the participants, five work exclusively in the industry, nine work exclusively in the academy, while five work in both Academia and Industry (see Figure 48). Under the research participants list, eight have the basic formation in computing, four in electrical/electronics engineering, three in management and four in other academic degrees (see Figure 50). Furthermore, there are undergraduate and graduated, three are undergraduate/bachelor degree; twelve participants have a master's degree, two have doctor's degree, and two have a post-doctor (see Figure 51). Every one of the participants acted actively in recent years in the Embedded Systems, IoT or Blockchain area.

Figure 47 – Location of the respondents.



Source: The Author (2018).

Figure 48 – What type of expertise do you have in security requirements for IoT design?



Source: The Author (2018).

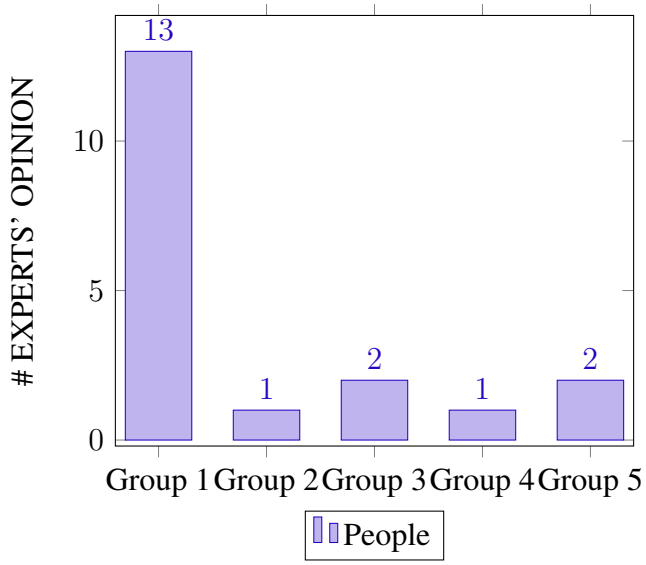
5.3.3 BIoT structure

In this question, we asked if the quantity and the organization of the levels in the BIoT are proper to evaluate the coverage of the Blockchain-based the Internet of Things Security descriptions. The objective was to identify if the descriptions levels are well-defined and understandable (Figure 49).

We realize there are convergent positions about the potentialities of the BIoT to the analysis and improvement of the development of Blockchain-based the Internet of Things Security solution, being this an innovator solution that aims the improvement of the development of this devices, seeking to minimize the challenges of the security, in particular, the absence of

concerns about confidentiality, integrity, availability and authenticity in solutions to the Internet of Things, supported by blockchain technology. The BIoT can be used as a polyvalent tool to support the development process of these devices.

Figure 49 – BIoT organization is suitable for the absence of concerns about confidentiality, integrity, availability, and authenticity.



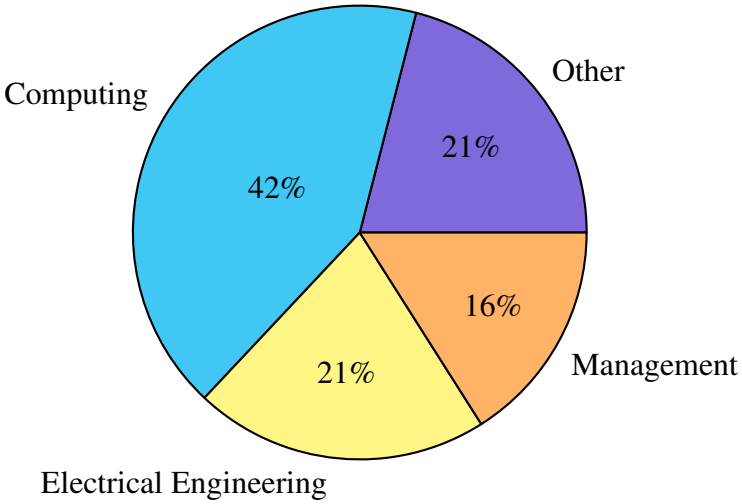
Source: The Author (2018).

Label – Group 1: Yes, no changes are required; **Group 2**: No, one or more descriptions must be included; **Group 3**: No, one or more descriptions must be included; **Group 4**: No, one or more descriptions must be grouped; **Group 5**: No, one or more descriptions must be updated.

In this question, interviewed affirmed that the solution is well organized and the number of levels is satisfactory. However, six respondents pointed out that the BIoT should have five levels, to be in line with the IBM ADEPT architectures. Still, in their viewpoint, this change in the BIoT would ease its acceptance by the organizations. Corroborating this idea, the Expert 2 affirms that "BIoT would have a greater acceptance if adapted or adherent to the (IBM ADEPT architectures with five levels)". The Expert 6, said that "is necessary to make a parallel of the BIoT with the IBM ADEPT architecture, from the IBM ADEPT implementation, know where it can treat elements of the BIoT, to a greater diffusion of the proposed solution".

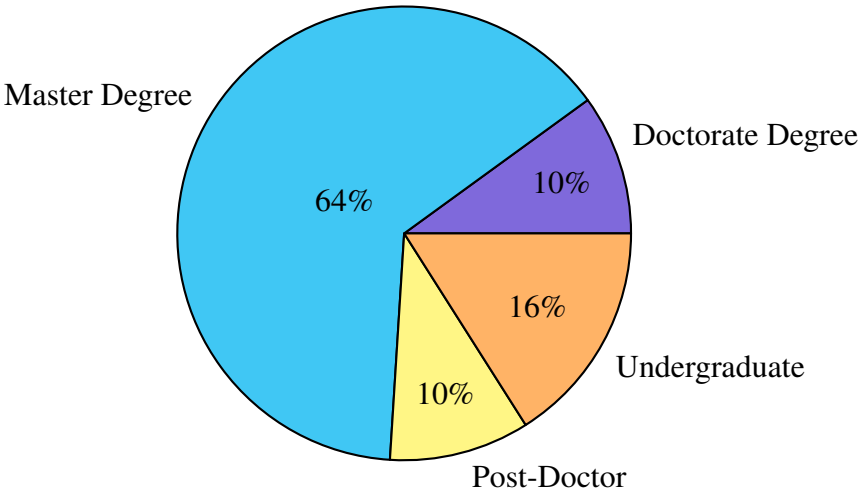
In minor instance, we observed some disagreeing positions about its potentialities. We had two experts that disagreed with level 1 of the BIoT. In their viewpoint, if the level does not evolve, there is no reason to present it in the solution. The Expert 12 affirms that an interesting approach is the used by the IBM ADEPT, because since the initial phase already has a defined structure, at least in an overview. Supporting this affirmative, Expert 19 says that there is a tendency of the researchers and universities in following the already existent models. However, models like the IBM ADEPT may not be the best alternative. "I think that a solution needs to be lean to be used by the companies and their developers. Other experts suggest improving nomenclature on some concepts and reanalyzing some groupings and levels".

Figure 50 – What is your area of expertise?



Source: The Author (2018).

Figure 51 – What is your schooling?



Source: The Author (2018).

Respondent 1 related problems to the perception layer, the sensors in the nodes have great varieties, and these are heterogeneous, usually have simple structures, and may not offer protection features. At the network layer, we discuss issues of malicious attacks or even DoS attacks. At the application layer, depending on the application field and the development team. In addition to costs, the cheaper may contain fewer security features; all of these factors can cause numerous security problems.

Expert 2 says “One of the major concerns in transactions between network nodes through sensors is *confidentiality*, that is, only those with the necessary rights and privileges will be able to access the information, whether it is stored, in processing or transit. Transactions, depending

on business rules, can be compromised, if intercepted by other nodes or unauthorized people".

Expert 2 says that "making sure that the packets received (or stored) have not been altered in an unauthorized manner (making sure unauthorized agents do not modify the data)".

Expert 12 says that "we must guard against improper modification or destruction of information, including the no retractability and authenticity of it". Other problems are in "IoT endpoints in different (vertical) applications often use different addressing applications and addressing formats, creating complexity".

Expert 6 says that "an approach that follows the standards established by the industries and segments of the IoT and Blockchain area presents the greater possibility of acceptance".

Expert 2 says "at the deepest level of this solution; it is normal to have rejections or not a complete understanding about proposed solution, compared to the more basic levels that have great acceptance". In this question "Other eight respondents say there are other important concerns is IoT systems generally employ low-complexity platforms with limited computing power and memory, preventing or limiting the use of a built-in firewall, as well as electrical limitations on end devices".

To address some of these challenges, we can be point some security considerations, as shown in Table 14:

Table 14 – Some challenges in the communication pattern, inside the lower layers (sensors) beside Blockchain for sensor data, as a trusted execution environment.

Actual status quo	Blockchain-based the IoT Security
no (or weak) encryption	strong end-to-end encryption
no (or weak) passwords	use of Transport Layer Security (TLS)
no (or weak) access control	strong identification

Source: The Auhor.

5.3.4 Benefits of BIoT according to the experts

After the interviews, experts were inquired about what benefits the BIoT brings to its IoT projects.

IoT applications risk being compromised in security attacks and revealing user behavioral patterns and personal lifestyles. IoT needs security as sensors, and integrated devices will transmit information to each other over the Internet, and for the time being, there is no clearer way to secure such communications than employing Blockchain technology. The BIoT presents some advantages:

- The devices showed a reduced risk of information tampering and the applications tested;

- Do not need a central agent. No need for anyone approving and validating all transactions and security rules. Instead, each node participates in the validation of all transactions on the blockchain network;
- The blockchain network looks like a meshed network; the BIoT allows remote access to devices to control nodes, from viewing maintenance data to routing information. An advantage of BIoT is that the blockchain network has an encrypted and unchanging table of security credentials, allowing devices to connect to other devices to validate operations;
- The low local level BIoT compute units are node and coordinator, which are units with migration and may have attributes assigned externally, due to the adoption of RESTful, interface that handles requests from devices is defined to enable cross-platform communication between devices and the blockchain network and established domain ontology;
- IoT device information is stored on the blockchain network for validation and access control;
- It is not clear; the context information is produced at a high level by the ContextManager service, extracted by the context detector. The BIoT seems like a good idea.

5.4 CONCLUDING REMARKS

This chapter presented the definition, planning, operation, analysis, and interpretation of the study based on expert opinion that evaluated the feasibility, completeness, and adequacy of the BIoT maturity.

For 68% (13) of the experts said it would adopt this solution and no changes are required; 32% (6) of the experts said they possibly adopt, but some changes would be included, excluded, grouped or updated. This result motivates us to invest heavily in continuous improvement of BIoT so that it can assist organizations in organizational processes effectively.

The study also identified some directions for improvements that will be analyzed and subsequently generated a new version of the proposed BIoT. This activity is already planned for our future work.

6 CONCLUSIONS AND PERSPECTIVES

This chapter revisits the research issues considered in the development of this thesis, highlights the main contributions of the and characterizes the job opportunities ahead of relevant BIoT research.

6.1 CONCLUSION

Today's computer network infrastructure does not guarantee adequate levels of security for the growing number of devices connected to the Internet. Given the many security incidents reported, we hypothesized how to structure a solution that could autonomously ensure device security (authenticity, privacy, data integrity, and confidentiality).

To this end, we investigate possible solutions within the context of IoT, Blockchain, and Security Ontologies. We propose a solution that mitigates/minimizes the effects of threats and attacks through constant monitoring and performance of the devices themselves. Finally, we list the conclusions in the following items:

- The BIoT proposal provided a unified technical structure for monitoring IoT ecosystem assets from ontology and knowledge base;
- At this point, some contributions can be highlighted to improve monitoring of specific asset security threats within various scenarios, with prospects for adoption in small and large technology parks;
- One such contribution is the identification of security ontologies for IoT and Blockchain. Knowledge about alerts and potential threats can be related to vulnerabilities and security properties. Moreover, these can be applied to ensure data integrity, confidentiality and privacy in blockchain networks;
- The ontology engineering reasoning capabilities provided discovery of scenario data and inferred security properties to verify vulnerabilities;
- Security alerts triggered from different categories at runtime can be integrated into the knowledge base;
- Another contribution was the mitigation of problems with data integrity, reliability, and privacy in heterogeneous environments, evaluated through performance testing (e.g., device testbed, Docker orchestration, semantic ontology assessment), and validated by experts.

Thus, by building the model presented, we identified that it is possible to build IoT solutions that allow greater resilience to attacks, authentication and data control, ensuring privacy through the development of services based on IBM ADEPT and Blockchain Domains as we can observe in the performance evaluations presented in Chapter 5.

As an additional contribution, we also matured a body of the knowledge of security area,

privacy, and reliability of the Internet of Things. This study is useful for both academia and business from different sectors. They can benefit from this consolidated knowledge and use it to guide the definition of their development processes geared to the paradigms of the Internet of Things.

6.2 PROPOSAL DISCUSSIONS

We have found out another difficulty transactions are visible to all nodes, opposite a need for devices in restricted environments. We solved some of these difficulties by adopting the consensus protocol for the Blockchain network, which guarantees execution without a bottleneck in time for registration.

This subject can be explored in other perspectives moving towards a consolidated knowledge about Blockchain-based the Internet of Things Security by development a better understanding of which factors influence in Blockchain-based the Internet of Things Security development processes, architectures, frameworks, approaches, models, methods, designs.

The work was supported in Blockchain networks, e.g., in closed environments and simulated nodes, performance measurements may present differences against a deployed environment.Other areas can be explored, as follows:

Computation: we were able to perform transactions typically, within the expected time, possibly due to the change of the PoW to PoC protocol. We would recommend test other protocols. Some tests performed in the 8-bit device presented an excellent resolution in hashes of size 15 characters. For 128 characters, it took between 5 and 6ms, representing 166 and 200 hashes per second that can mean low performance in this context.

Storage: Despite the concern, Blockchain storage of the application did not present any significant storage problems, considering the proposed BIoT to perform storage in an aggregate device, restricted in character and limited only to the application.

Energy: At this point, we did not carry out tests to measure energy consumption, to perform all the tests being power supplied by wire. In the works of (LAURIDSEN *et al.*, 2016) it was measured which IoT device is designed to consume 0.3mWh per day, and can run for at least five years using a CR2032 batt with the capacity of 600mWh. It is by energy-saving strategies, e.g., sleep mode and high-efficiency communication technologies.

Compare with previous works (Section 3.7), the BIoT ontology we propose has the following advantages:

- It is more reliable. We use the blockchain network to test attacks on the proposed BIoT. Unsuccessful attempts to change packages. Network availability remained stable even as packet traffic increased;
- Data Integrity Verification efficiency can be enhanced with an increasing number of nodes;

- However, we only implemented the fundamental function of our proposed solution. We also can validate IoT devices in our test yet have low efficiency in writing to a smart contract for use in deploy environment. The test environment is still of small scale.

6.3 PUBLICATIONS

- The Blockchain-based Internet of Things development: initiatives and challenges. In: International Conference on Software Engineering Advances, 2017, Athens. The Twelfth International Conference on Software Engineering Advances (MENDONÇA; SILVA JÚNIOR; ALENCAR, 2017).
- Perdas em sistemas de distribuição de água: estudo baseado em revisão sistemática. In: Congresso Nacional de Saneamento e Meio Ambiente, São Paulo. Congresso Nacional de Saneamento e Meio Ambiente (SILVA JÚNIOR *et al.*, 2017).
- A local water control architecture based on Internet of Things to water supply crisis. In: 31st South Symposium on Microelectronics, Porto Alegre. 31st South Symposium on Microelectronics, 2016 (MENDONÇA *et al.*, 2016).

6.4 PERSPECTIVES

There are numerous studies about security and privacy issues of blockchain networks, basically to the heterogeneous interconnected devices.

In Moin *et al.* (2019) are addressing these issues and should be studied because they can affect the availability services of the distributed network in the IoT environment: i) Scalable data management; ii) Complexity; iii) Network speed; iv) Interoperability; v) Privacy; vi) Standards; vii) Fork; viii) Regulations and governance; ix) Unavoidable security flaw; and x) Politics.

Although some authors report possible implications for the use of Blockchain in IoT, we can not measure, for this work, issues related above, such as, for example, *network speed*, *fork*, *regulations and governance*, *unavoidable security flaw* and *politics*.

Other security factors to be addressed in future work are those discussed through Top 10 IoT vulnerabilities.

In this context, the OWASP related the top 10 IoT vulnerabilities that include: i) Weak, guessable, or hardcoded passwords; ii) Insecure network services; iii) Insecure ecosystem interfaces; iv) Lack of secure update mechanisms; v) Use of insecure or outdated components; vi) Insufficient privacy protection; vii) Insecure data transfer and storage; viii) Lack of device management; ix) Insecure default settings; and x) Lack of physical hardening.

We can notice that in the OWASP collaborators paper, about 10 IoT vulnerabilities, there are factors which impact directly on human decision (it is not reported in this study, as such as topics *Weak, guessable, or hardcoded passwords*, *Insecure default settings* and *Lack of physical*

hardening); and other technical factors (it is goals this study, as such as topics *Insecure network services*, *Insecure ecosystem interfaces*, *Lack of secure update mechanisms*, *Use of insecure or outdated components Insufficient privacy protection*, *Insecure data transfer, and storage*, *Lack of device management*).

We conducted blockchain network stability and security testing cases implemented to support the Internet of Things. We even tested in environments with different approaches to verifying the latest blockchain record, through the Proof of Work (PoW), which required more computational and time-consuming power and the Proof of Consensus (PoC), the latter being more efficient, matching our study proposal, assigning trusted coordinating nodes to the network, without the need for a trusted third-party. Other validation methods can be tested to validate better response times, data availability. In future work, other validation methods can be tested to validate better response times and data availability, security, and privacy.

When conducting blockchain network security and stability assessments to support the Internet of Things, we are faced with new knowledge currently very relevant. The Software-Defined Networking (SDN) area presents us with new possibilities to IoT and Blockchain networks aimed at home environments, with ease of establishing tables of security policies and rules, which considerably facilitated the construction of our testbed cases. It is noteworthy that these new studies can be take in into future work of Blockchain-based the Internet of Things Security.

During the evaluation of the BIoT by experts, everyone felt the need to establish new security policies and rules. What could be facilitated by the integration of OpenFlow, more specifically, MikroTik OpenFlow, called MikroFlow. However, Mikrotik does not recommend using this environment for production, it recommends using it only as a testing and learning environment.

REFERENCES

AGARWAL, R. *et al.* Unified IoT ontology to enable interoperability and federation of testbeds. In: WORLD FORUM ON INTERNET OF THINGS (WF-IOT), 3., 2016, Reston. **Proceedings** [...]. New Jersey: IEEE, 2016. p. 70–75. Available from Internet: https://ieeexplore.ieee.org/abstract/document/7845470/. Accessed on: 17 jan. 2017.

AGRAWAL, S.; VIEIRA, D. A survey on Internet of Things. **Abakós**, Belo Horizonte, v. 1, n. 2, p. 78–95, 2013. Available from Internet: http://periodicos.pucminas.br/index.php/abakos/article/view/5372. Accessed on: 2 jan. 2016.

ALAYA, M. B. *et al.* Toward semantic interoperability in oneM2M architecture. **IEEE Communications Magazine**, IEEE, v. 53, n. 12, p. 35–41, 2015. Available from Internet: https://ieeexplore.ieee.org/abstract/document/7355582/. Accessed on: 2 jan. 2016.

ALAYA, M. B. *et al.* Towards semantic data interoperability in oneM2M standard. **IEEE Communications Magazine**, Institute of Electrical and Electronics Engineers, v. 53, n. 12, p. 35–41, 2015. Available from Internet: https://hal.archives-ouvertes.fr/hal-01228327/. Accessed on: 2 jan. 2016.

ALI, I.; SABIR, S.; ULLAH, Z. Internet of things security, device authentication and access control: a review. **CoRR**, Wadern, v. 1, n. 1901.07309, 2019. Available from Internet: https://arxiv.org/abs/1901.07309. Accessed on: 10 jun. 2019.

AMARO, A.; PÓVOA, A.; MACEDO, L. **A arte de fazer questionários**. Porto: FCUP, 2005. Available from Internet: https://sites.google.com/site/sociologiaemaccao/2-metodologia-da-investigacao-sociologica/a-arte-de-fazer-questionarios.doc. Accessed on: 2 ago. 2015.

ANTONOPOULOS, A. M. **Mastering Bitcoin: unlocking digital cryptocurrencies**. Sebastopol: O'Reilly Media, Inc., 2014.

ATZORI, M. Blockchain technology and decentralized governance: Is the state still necessary? **SSRN**, New York, v. 2709713, 2015. Available from Internet: http://www.ssrn.com/abstract=2709713. Accessed on: 29 jul. 2016.

ATZORI, M. Blockchain-based architectures for the internet of things: A survey. **SSRN**, New York, v. 2846810, 2017. Available from Internet: https://ssrn.com/abstract=2846810. Accessed on: 2 abr. 2016.

BAKER, R. H. **Computer security handbook**. Pennsylvania: TAB Professional and Reference Books, 1991.

BANERJEE, M.; LEE, J.; CHOO, K.-K. R. A Blockchain future for Internet of Things Security: A position paper. **Digital Communications and Networks**, Elsevier, Edinburgh, v. 4, n. 3, p. 149–160, 2018. Available from Internet: https://doi.org/10.1016/j.dcan.2017.10.006. Accessed on: 6 dez. 2018.

BASNET, S. R.; SHAKYA, S. BSS: Blockchain security over software defined network. In: 2017 INTERNATIONAL CONFERENCE ON COMPUTING, COMMUNICATION AND

AUTOMATION (ICCCA)., 2017, Greater Noida. **Proceedings** [. . .]. New Jersey: IEEE, 2017. p. 720–725. Available from Internet: https://ieeexplore.ieee.org/abstract/document/8229910/. Accessed on: 4 jan. 2018.

BEECHAM, S. *et al.* Using an expert panel to validate a requirements process improvement model. **Journal of Systems and Software**, Elsevier, v. 76, n. 3, p. 251–275, 2005. Available from Internet: https://www.sciencedirect.com/science/article/pii/S0164121204000974. Accessed on: 2 jul. 2015.

BERMUDEZ-EDO, M. *et al.* IoT-Lite: a lightweight semantic model for the Internet of Things. In: INTL IEEE CONFERENCES ON UIC/ATC/SCALCOM/CBDCOM/IOP/SMARTWORLD, 16., 2016, Toulouse. **Proceedings** [. . .]. Toulouse: IEEE, 2016. p. 90–97. Available from Internet: https://ieeexplore.ieee.org/abstract/document/7816831/. Accessed on: 8 jan. 2017.

BERNERS-LEE, T. WWW: Past, present, and future. **Computer**, IEEE, New Jersey, v. 29, n. 10, p. 69–77, 1996. Available from Internet: https://doi.org/10.1109/2.539724. Accessed on: 6 dez. 2018.

BIOLCHINI, J. *et al.* **Systematic Review in Software Engineering**. Rio de Janeiro: COPPE/UFRJ, 2005. v. 679, n. 05, 45 p. Available from Internet: http://www.cin.ufpe.br/~in1037/leitura/systematicReviewSE-COPPE.pdf. Accessed on: 2 jan. 2015.

BIRYUKOV, A.; KHOVRATOVICH, D.; PUSTOGAROV, I. Deanonymisation of clients in Bitcoin P2P network. In: CONFERENCE ON COMPUTER AND COMMUNICATIONS SECURITY, 21., 2014, Scottsdale. **Proceedings** [. . .]. New York: ACM, 2014. p. 15–29. Available from Internet: http://doi.acm.org/10.1145/2660267.2660379. Accessed on: 27 jun. 2015.

BORGOHAIN, T.; KUMAR, U.; SANYAL, S. Survey of security and privacy issues of Internet of Things. **arXiv preprint**, Ithaca, v. 1, n. 1501.02211, 2015. Available from Internet: http://arxiv.org/abs/1501.02211. Accessed on: 22 jun. 2016.

BRAMBILLA, G.; AMORETTI, M.; ZANICHELLI, F. Using Blockchain for peer-to-peer proof-of-location. **arXiv preprint**, Ithaca, v. 1, n. 1607.00174, 2016. Available from Internet: https://arxiv.org/abs/1607.00174. Accessed on: 11 jun. 2016.

BRESCIANI, P. *et al.* Tropos: An agent-oriented software development methodology. **Autonomous Agents and Multi-Agent Systems**, Springer, New York, v. 8, n. 3, p. 203–236, 2004. Available from Internet: https://link.springer.com/article/10.1023/B:AGNT.0000018806.20944.ef. Accessed on: 4 jun. 2015.

BRODY, P.; PURESWARAN, V. Device democracy: saving the future of the Internet of Things. **IBM Corporation**, New York, 2015. Available from Internet: http://m2mworldnews.com/download/white-papers/IBM-Saving-the-future-of-IoT.pdf. Accessed on: 8 dez. 2015.

BUTERIN, V. A next-generation smart contract and decentralized application platform. **Ethereum White Paper**, Singapore, v. 3, p. 37, 2014. Available from Internet: https://cryptorating.eu/whitepapers/Ethereum/Ethereum_white_paper.pdf. Accessed on: 15 nov. 2015.

CAVALCANTE, E. *et al.* An analysis of reference architectures for the Internet of Things. In: INTERNATIONAL WORKSHOP ON EXPLORING COMPONENT-BASED TECHNIQUES FOR CONSTRUCTING REFERENCE ARCHITECTURES, 1., 2015,

Montréal. **Proceedings** [...]. New York: ACM, 2015. p. 13–16. Available from Internet: https://dl.acm.org/citation.cfm?id=2755569. Accessed on: 12 nov. 2016.

CHAE, M.; LEE, H.; LEE, K. A performance comparison of linux containers and virtual machines using Docker and KVM. **Cluster Computing**, Springer, New York, v. 22, n. 1, p. 1765–1775, 2019. Available from Internet: https://link.springer.com/article/10.1007/s10586-017-1511-2. Accessed on: 16 jun. 2019.

CHHIBBER, S.; APOSTOLAKIS, G.; OKRENT, D. A taxonomy of issues related to the use of expert judgments in probabilistic safety studies. **Reliability Engineering & System Safety**, Elsevier, New York, v. 38, n. 1-2, p. 27–45, 1992. Available from Internet: https://www.sciencedirect.com/science/article/pii/095183209290103R. Accessed on: 8 jan. 2015.

CHIDAMBER, S. R.; KEMERER, C. F. A metrics suite for object oriented design. **IEEE Transactions on software engineering**, IEEE, New Jersey, v. 20, n. 6, p. 476–493, 1994. Available from Internet: https://ieeexplore.ieee.org/abstract/document/295895/. Accessed on: 9 mar. 2016.

CHRISTIDIS, K.; DEVETSIKIOTIS, M. Blockchains and smart contracts for the Internet of Things. **Ieee Access**, Ieee, New Jersey, v. 4, p. 2292–2303, 2016. Available from Internet: http://ieeexplore.ieee.org/document/7467408. Accessed on: 18 jun. 2017.

COMPTON, M. *et al.* The SSN ontology of the W3C semantic sensor network incubator group. **Web semantics: science, services and agents on the World Wide Web**, Elsevier, New York, v. 17, p. 25–32, 2012. Available from Internet: https://www.sciencedirect.com/science/article/pii/S1570826812000571. Accessed on: 9 out. 2016.

CONOSCENTI, M.; VETRO, A.; MARTIN, J. C. D. Blockchain for the Internet of Things: A systematic literature review. In: INTERNATIONAL CONFERENCE OF COMPUTER SYSTEMS AND APPLICATIONS (AICCSA), 13., 2016, Agadir. **Proceedings** [...]. New Jersey: IEEE, 2016. p. 1–6. Available from Internet: http://ieeexplore.ieee.org/document/7945805. Accessed on: 25 maio 2017.

COOKE, R. *et al.* **Experts in uncertainty: opinion and subjective probability in science**. Oxford: Oxford University Press on Demand, 1991.

COUTINHO, C. M. G. F. P. **Percursos da Investigação em Tecnologia Educativa em Portugal: uma abordagem temática e metodológica a publicações científicas (1985-2000)**. Minho: [s.n.], 2005.

DELOACH, S. A.; KUMAR, M. Multi-agent systems engineering: an overview and case study. In: AGENT-ORIENTED METHODOLOGIES. **Proceedings** [...]. Hershey: IGI Global, 2005. p. 317–340. Available from Internet: https://www.igi-global.com/chapter/agent-oriented-methodologies/5063. Accessed on: 12 set. 2015.

DENKER, G.; KAGAL, L.; FININ, T. Security in the Semantic Web using OWL. **Information Security Technical Report**, Elsevier, New York, v. 10, n. 1, p. 51–58, 2005. Available from Internet: https://www.sciencedirect.com/science/article/pii/S1363412704000032. Accessed on: 5 jan. 2015.

DIAS-NETO, A. C.; SPINOLA, R.; TRAVASSOS, G. H. Developing software technologies through experimentation: experiences from the battlefield. In: IBERO-AMERICAN CONFERENCE ON SOFTWARE ENGINEERING, 13., 2010, Cuenca. **Proceedings** [. . .]. Berlin: ResearchGate, 2010. Available from Internet: https://www.researchgate.net/profile/Arilo_Neto/publication/289812993_Developing_software_technologies_through_experimentation_Experiences_from_the_battlefield/links/569eb2e908aee4d26ad0505b.pdf. Accessed on: 15 fev. 2017.

DORRI, A. *et al.* Blockchain for IoT security and privacy: The case study of a smart home. In: IEEE INTERNATIONAL CONFERENCE ON PERVASIVE COMPUTING AND COMMUNICATIONS WORKSHOPS (PERCOM WORKSHOPS)., 2017, Kona. **Proceedings** [. . .]. New Jersey: IEEE, 2017. p. 618–623. Available from Internet: https://ieeexplore.ieee.org/abstract/document/7917634/. Accessed on: 3 mar. 2017.

DYBA, T. An instrument for measuring the key factors of success in software process improvement. **Empirical software engineering**, Springer, New York, v. 5, n. 4, p. 357–390, 2000. Available from Internet: https://link.springer.com/article/10.1023/A:1009800404137. Accessed on: 8 set. 2017.

EMAM, K. E.; MADHAVJI, N. H. An instrument for measuring the success of the requirements engineering process in information systems development. **Empirical Software Engineering**, Springer, New York, v. 1, n. 3, p. 201–240, 1996. Available from Internet: https://link.springer.com/article/10.1007/BF00127446. Accessed on: 9 ago. 2016.

EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE. **Cyber Security for Consumer Internet of Things**. New York, 2019. Available from Internet: https://www.etsi.org/deliver/etsi_ts/103600_103699/103645/01.01.01_60/ts_103645v010101p.pdf. Accessed on: 3 out. 2019.

FALLIS, A. Rootstock Platform: Bitcoin Powered Smart Contracts - White Paper. **Journal of Chemical Information and Modeling**, New York, v. 53, n. 9, p. 1689–1699, 2013.

GARCIA, V. C. **RiSE reference model for software reuse adoption in Brazilian companies**. 2010. 184 p. PhD Thesis (Ciência da Computação) — Centro de Informática, Universidade Federal de Pernambuco, Recife, 2010. Available from Internet: https://repositorio.ufpe.br/handle/123456789/1729. Accessed on: 2 ago. 2017.

GIRARDI, R.; LINDOSO, A. N. An ontology-based methodology for multiagent domain engineering. In: PORTUGUESE CONFERENCE ON ARTIFICIAL INTELLIGENCE., 2005, Covilha. **Proceedings** [. . .]. New Jersey: IEEE, 2005. p. 321–324. Available from Internet: https://ieeexplore.ieee.org/abstract/document/4145979/. Accessed on: 5 jun. 2015.

GOODMAN, E. P. The atomic age of data: Policies for the Internet of Things. **SSRN**, Elsevier, New Yourk, n. 2605201, 2015. Available from Internet: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2605201. Accessed on: 3 ago. 2016.

GREENBERG, J.; RODRIGUEZ, E. M. **Knitting the semantic web**. New York: Routledge, 2012. v. 43.

GUHA, S.; KUMAR, S. Emergence of big data research in operations management, information systems, and healthcare: Past contributions and future roadmap. **Production and Operations Management**, Wiley Online Library, New York, v. 27, n. 9, p. 1724–1735, 2018. Available from Internet: https://onlinelibrary.wiley.com/doi/abs/10.1111/poms.12833. Accessed on: 3 dez. 2018.

GUTTMAN, B.; ROBACK, E. A. **An introduction to computer security: the NIST handbook**. Collingdale: Diane Publishing Co, 1995.

GYRARD, A.; BONNET, C.; BOUDAOUD, K. An ontology-based approach for helping to secure the ETSI machine-to-machine architecture. In: IEEE INTERNATIONAL CONFERENCE ON INTERNET OF THINGS (ITHINGS), AND IEEE GREEN COMPUTING AND COMMUNICATIONS (GREENCOM) AND IEEE CYBER, PHYSICAL AND SOCIAL COMPUTING (CPSCOM), 14., 2014, Tapei. **Proceedings** [. . .]. New Jersey: IEEE, 2014. p. 109–116. Available from Internet: https://ieeexplore.ieee.org/abstract/document/7059650/. Accessed on: 3 jan. 2015.

GYRARD, A.; BONNET, C.; BOUDAOUD, K. Enrich machine-to-machine data with semantic web technologies for cross-domain applications. In: IEEE WORLD FORUM ON INTERNET OF THINGS (WF-IOT)., 2014, Seoul. **Proceedings** [. . .]. New Jersey: IEEE, 2014. p. 559–564. Available from Internet: https://ieeexplore.ieee.org/document/6803229. Accessed on: 2 jan. 2015.

GYRARD, A.; DATTA, S. K.; BONNET, C. A survey and analysis of ontology-based software tools for semantic interoperability in IoT and WoT landscapes. In: WORLD FORUM ON INTERNET OF THINGS (WF-IOT), 4., 2018, Singapore. **Proceedings** [. . .]. New Jersey: IEEE, 2018. p. 86–91. Available from Internet: https://ieeexplore.ieee.org/abstract/document/8355091/. Accessed on: 13 abr. 2019.

GYRARD, A. *et al.* Standardizing generic cross-domain applications in Internet of Things. In: IEEE GLOBECOM WORKSHOPS (GC WKSHPS), 14., 2014, Austin. **Proceedings** [. . .]. New Jersey: IEEE, 2014. p. 589–594. Available from Internet: https://ieeexplore.ieee.org/abstract/document/7063496/. Accessed on: 3 fev. 2015.

HAKIM, C. **Research design: Strategies and choices in the design of social research**. London: Allen and Unwin, 1987.

HAMMI, M. T. *et al.* Bubbles of Trust: A decentralized blockchain-based authentication system for IoT. **Computers & Security**, Elsevier, New York, v. 78, p. 126–142, 2018. Available from Internet: https://www.sciencedirect.com/science/article/pii/S0167404818300890. Accessed on: 27 dez. 2018.

HANG, L.; KIM, D.-H. Design and Implementation of an Integrated IoT Blockchain Platform for Sensing Data Integrity. **Sensors**, Multidisciplinary Digital Publishing Institute, Bethesda, v. 19, n. 10, p. 2228, 2019. Available from Internet: https://www.mdpi.com/1424-8220/19/10/2228. Accessed on: 5 jun. 2019.

HARDJONO, T.; SMITH, N. Cloud-based commissioning of constrained devices using permissioned blockchains. In: ACM INTERNATIONAL WORKSHOP ON IOT PRIVACY, TRUST, AND SECURITY, 2., 2016, Xi'an. **Proceedings** [. . .]. New Jersey: ACM, 2016. p. 29–36. Available from Internet: https://dl.acm.org/citation.cfm?id=2899012. Accessed on: 5 jan. 2017.

HART, L. *et al.* Usage scenarios and goals for ontology definition metamodel. In: INTERNATIONAL CONFERENCE ON WEB INFORMATION SYSTEMS ENGINEERING., 2004, Brisbane. **Proceedings** [. . .]. New York: Springer, 2004. p. 596–607. Available from Internet: https://link.springer.com/chapter/10.1007/978-3-540-30480-7_62. Accessed on: 8 nov. 2015.

HASHEMI, S. H. *et al.* World of empowered IoT users. In: 2016 IEEE FIRST INTERNATIONAL CONFERENCE ON INTERNET-OF-THINGS DESIGN AND IMPLEMENTATION (IOTDI)., 2016, Berlin. **Proceedings** [. . .]. New Jersey: IEEE, 2016. p. 13–24. Available from Internet: https://ieeexplore.ieee.org/abstract/document/7471347/. Accessed on: 2 out. 2017.

HEFLIN, J.; STUCKENSCHMIDT, H. Web-scale semantic information processing. **Journal of Web Semantics-Science Services and Agents on the World Wide Web**, Elsevier, New York, v. 10, p. 1, 2012. Available from Internet: https://pdfs.semanticscholar.org/36d4/e92766d87e7e9fdc7d3b0fdce5f98f0ec9f2.pdf. Accessed on: 2 ago. 2015.

HERZOG, A.; SHAHMEHRI, N.; DUMA, C. An ontology of information security. **International Journal of Information Security and Privacy (IJISP)**, IGI Global, Hershey, v. 1, n. 4, p. 1–23, 2007. Available from Internet: https://www.igi-global.com/article/ontology-information-security/2468. Accessed on: 12 out. 2015.

HORA, S.; IMAN, R. Expert opinion in risk analysis: the NUREG-1150 methodology. **Nuclear Science and Engineering**, Taylor & Francis, Abingdon, v. 102, n. 4, p. 323–331, 1989. Available from Internet: https://www.tandfonline.com/doi/abs/10.13182/NSE89-A23645. Accessed on: 13 out. 2015.

HUCKLE, S. *et al.* Internet of Things, blockchain and shared economy applications. **Procedia computer science**, Elsevier, New York, v. 98, p. 461–466, 2016. Available from Internet: https://www.sciencedirect.com/science/article/pii/S1877050916322190. Accessed on: 2 jan. 2017.

HUNG, M. Leading the IoT, Gartner insights on how to lead in a connected world. **Gartner Research**, Stamford, p. 1–29, 2017.

ITU Telecommunication Standardization Sector. Recommendation ITU-T Y.2060: Overview of the Internet of things. **Series Y: Global information infrastructure, internet protocol aspects and next-generation networks - Frameworks and functional architecture models**, Geneva, p. 22, 2012. Available from Internet: https://www.itu.int/rec/T-REC-Y.2060-201206-I/en. Accessed on: 2 jan. 2015.

JAIN, R. **The art of computer systems performance analysis: techniques for experimental design, measurement, simulation, and modeling**. Hoboken: John Wiley & Sons, 1990.

JAVAID, U.; AMAN, M. N.; SIKDAR, B. BlockPro: Blockchain based Data Provenance and Integrity for Secure IoT Environments. In: WORKSHOP ON BLOCKCHAIN-ENABLED NETWORKED SENSOR SYSTEMS, 1., 2018, Shenzhen. **Proceedings** [. . .]. New York: ACM, 2018. p. 13–18. Available from Internet: https://dl.acm.org/citation.cfm?id=3282281. Accessed on: 2 dez. 2018.

JR, F. J. F.; FOWLER, F. J. **Improving survey questions: Design and evaluation**. Tousand Oaks: SAGE Publications, Inc, 1995.

KEENEY, R. L.; MCDANIELS, T. L. Value-focused thinking about strategic decisions at BC Hydro. **Interfaces**, INFORMS PubsOnLine, Catonsville, v. 22, n. 6, p. 94–109, 1992. Available from Internet: https://pubsonline.informs.org/doi/abs/10.1287/inte.22.6.94. Accessed on: 8 out. 2015.

KHAN, M. A.; SALAH, K. IoT security: Review, blockchain solutions, and open challenges. **Future Generation Computer Systems**, Elsevier, New York, v. 82, p. 395–411, 2018. Available from Internet: https://www.sciencedirect.com/science/article/pii/S0167739X17315765. Accessed on: 2 jan. 2019.

KIM, A.; LUO, J.; KANG, M. Security ontology for annotating resources. In: OTM CONFEDERATED INTERNATIONAL CONFERENCES "ON THE MOVE TO MEANINGFUL INTERNET SYSTEMS"., 2005, Agia Napa. **Proceedings** [. . .]. New York: Springer, 2005. p. 1483–1499. Available from Internet: https://link.springer.com/chapter/10.1007/11575801_34. Accessed on: 2 fev. 2015.

KITCHENHAM, B.; CHARTERS, S. Guidelines for performing Systematic Literature Reviews in Software Engineering. CiteSeer, Princeton, 2007. Available from Internet: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.117.471. Accessed on: 2 jan. 2015.

KOTIS, K.; KATASONOV, A. An ontology for the automated deployment of applications in heterogeneous IoT environments. **Semantic Web Journal (SWJ)**, IOS Press, Clifton, 2012. Available from Internet: http://www.academia.edu/download/37043198/swj247_0.pdf. Accessed on: 2 abr. 2015.

KRUIJFF, J. d.; WEIGAND, H. Towards a Blockchain Ontology. **Research Report Tillburg University**, Tilburg University, Tilburgo, 2017. Available from Internet: https://www.list.lu/fileadmin/files/Event/sites/tudor/files/Training_Center/OTHERS/VMBO2017_paper_5.pdf. Accessed on: 2 jan. 2018.

KVALE, S. **InterViews: An Introduction to Qualitative Research Interviewing**. Tousand Oaks: SAGE Publications, Inc, 1996.

LAURENCE, T. **Blockchain for Dummies**. 2. ed. Hoboken: John Wiley & Sons, Inc, 2019.

LAURIDSEN, M. *et al.* Coverage and capacity analysis of LTE-M and NB-IoT in a rural area. In: VEHICULAR TECHNOLOGY CONFERENCE (VTC-FALL), 84., 2016, Montreal. **Proceedings** [. . .]. New Jersey: IEEE, 2016. p. 1–5. Available from Internet: https://ieeexplore.ieee.org/abstract/document/7880946/. Accessed on: 2 jan. 2017.

LI, M.; SMIDTS, C. S. A ranking of software engineering measures based on expert opinion. **IEEE Transactions on Software engineering**, IEEE, Princeton, v. 29, n. 9, p. 811–824, 2003. Available from Internet: https://ieeexplore.ieee.org/abstract/document/1232286/. Accessed on: 2 jul. 2015.

LI, S.; XU, L. D.; ZHAO, S. The Internet of Things: a survey. **Information Systems Frontiers**, Springer, New York, v. 17, n. 2, p. 243–259, 2015. Available from Internet: https://link.springer.com/article/10.1007/s10796-014-9492-7. Accessed on: 25 fev. 2016.

LIDDELL, H. G. *et al.* **Greek–English Lexicon: with a revised supplement**. Oxford: Oxford: Clarendon Press, 1996. Available from Internet: http://www.areopage.net. Accessed on: 9 jan. 2015.

LIU, B. *et al.* Blockchain based data integrity service framework for IoT data. In: INTERNATIONAL CONFERENCE ON WEB SERVICES (ICWS)., 2017, Honolulu. **Proceedings** [. . .]. New Jersey: IEEE, 2017. p. 468–475. Available from Internet: https://ieeexplore.ieee.org/abstract/document/8029796/. Accessed on: 6 dez. 2017.

LOGVINOV, O. *et al.* **Standard for an architectural framework for the Internet of Things (IoT) IEEE p2413**. New Jersey: IEEE, 2016. Available from Internet: http://meptec.org/Resources/8%20-%20Logvinov.pdf. Accessed on: 30 set. 2017.

LUU, L. *et al.* Making smart contracts smarter. In: ACM SIGSAC CONFERENCE ON COMPUTER AND COMMUNICATIONS SECURITY., 2016, Vienna. **Proceedings** [...]. New York: ACM, 2016. p. 254–269. Available from Internet: https://dl.acm.org/citation.cfm?id=2978309. Accessed on: 8 dez. 2016.

LUU, L. *et al.* Demystifying incentives in the consensus computer. In: ACM SIGSAC CONFERENCE ON COMPUTER AND COMMUNICATIONS SECURITY, 22., 2015, Denver. **Proceedings** [...]. New York: ACM, 2015. p. 706–719. Available from Internet: https://dl.acm.org/citation.cfm?id=2813659. Accessed on: 2 ago. 2016.

MAKSIMOVIC, M.; VUJOVIC, V.; OMANOVIC-MIKLICANIN, E. A low cost Internet of Things solution for traceability and monitoring food safety during transportation. In: INTERNATIONAL CONFERENCE ON INFORMATION AND COMMUNICATION TECHNOLOGIES IN AGRICULTURE, FOOD AND ENVIRONMENT (HAICTA), 7., 2016, Kavala. **Proceedings** [...]. Aachen: CEUR, 2015. p. 583–593. Available from Internet: http://ceur-ws.org/Vol-1498/HAICTA_2015_paper66.pdf. Accessed on: 13 nov. 2016.

MARTIMIANO, L. A. F. **Sobre a estruturação de informação em sistemas de segurança computacional: o uso de ontologias**. 2006. 184 p. PhD Thesis (Ciência da Computação e Matemática Computacional) — Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Paulo, 2006. Available from Internet: http://www.teses.usp.br/teses/disponiveis/55/55134/tde-02102006-091853/en.php. Accessed on: 28 abr. 2015.

MAY, I. **Systems and software engineering–architecture description**. New Jersey, 2011.

MELINE, T. Selecting studies for systematic review: Inclusion and exclusion criteria. **Contemporary issues in communication science and disorders**, ASHA, Rockville, v. 33, n. 21-27, 2006. Available from Internet: https://www.asha.org/uploadedfiles/asha/publications/cicsd/2006sselectingstudiesforsystematicreview.pdf. Accessed on: 5 mar. 2015.

MENDEZ MENA, D. M.; YANG, B. Blockchain-based Whitelisting for consumer IoT Devices and Home Networks. In: ANNUAL SIG CONFERENCE ON INFORMATION TECHNOLOGY EDUCATION, 19., 2018, Lauderdale. **Proceedings** [...]. New York: ACM, 2018. p. 7–12. Available from Internet: https://dl.acm.org/citation.cfm?id=3241853. Accessed on: 27 dez. 2018.

MENDONÇA, S. F. T. O.; SILVA JÚNIOR, J. F.; ALENCAR, F. M. R. The Blockchain-based Internet of Things development: Initiatives and challenges. **The Twelfth International Conference on Software Engineering Advances**, ICSEA, Athens, n. c, p. 39, 2017. Available from Internet: https://www.researchgate.net/profile/Luigi_Lavazza/publication/320402853_ICSEA_2017_The_Twelfth_International_Conference_on_Software_Engineering_Advances/links/59e2311e458515393d57ecca/ICSEA-2017-The-Twelfth-International-Conference-on-Software-Engineering-Advances.pdf#page=40. Accessed on: 9 jan. 2018.

MENDONÇA, S. F. T. O. *et al.* A Local Water Control Architecture based on Internet of Things to Water Supply Crisis. In: SOUTH SYMPOSIUM ON MICROELECTRONICS, 31., 2016, Porto Alegre. **Proceedings** [...]. Porto Alegre: SBC, 2016. p. 113–116. Available from Internet: https://inf.ufrgs.br/gme/emicro. Accessed on: 2 dez. 2016.

MEULEN, R. van der. **Gartner Says 6.4 Billion Connected "Things" Will Be in Use in 2016, Up 30 Percent From 2015**. Stanford: Gartner, 2015. Available from Internet: http://www.gartner.com/newsroom/id/3165317. Accessed on: 22 ago. 2016.

MIESSLER, D. Securing the Internet of Things: Mapping attack surface areas using the OWASP IoT top 10. In: RSA CONFERENCE., 2015, San Francisco. **Proceedings** [...]. California: OWASP, 2015. Available from Internet: https://www.owasp.org/images/5/51/RSAC2015-OWASP-IoT-Miessler.pdf. Accessed on: 17 ago. 2016.

MINERVA, R.; BIRU, A.; ROTONDI, D. Towards a definition of the Internet of Things (IoT). **IEEE Internet Initiative**, IEEE, New Jersey, v. 1, p. 1–86, 2015. Available from Internet: https://iot.ieee.org/images/files/pdf/IEEE_IoT_Towards_Definition_Internet_of_Things_Revision1_27MAY15.pdf. Accessed on: 8 nov. 2016.

MINOLI, D.; OCCHIOGROSSO, B. Blockchain mechanisms for IoT security. **Internet of Things**, Elsevier, New York, v. 1, p. 1–13, 2018. Available from Internet: https://www.sciencedirect.com/science/article/pii/S2542660518300167. Accessed on: 22 nov. 2018.

MOIN, S. *et al.* Securing IoTs in distributed blockchain: Analysis, requirements and open issues. **Future Generation Computer Systems**, Elsevier, New York, v. 100, p. 325–343, 2019. Available from Internet: https://www.sciencedirect.com/science/article/pii/S0167739X18330851. Accessed on: 8 jun. 2019.

MORGAN, D. L. **Focus groups as qualitative research**. 2. ed. Tousand Oaks: Sage Publications, Inc, 1997. v. 16.

MOZZAQUATRO, B. A.; JARDIM-GONCALVES, R.; AGOSTINHO, C. Towards a reference ontology for security in the Internet of Things. In: IEEE INTERNATIONAL WORKSHOP ON MEASUREMENTS & NETWORKING (M&N)., 2015, Coimbra. **Proceedings** [...]. New Jersey: IEEE, 2015. p. 1–6. Available from Internet: https://ieeexplore.ieee.org/abstract/document/7322984/. Accessed on: 12 out. 2016.

MOZZAQUATRO, B. A. *et al.* An ontology-based security framework for decision-making in industrial systems. In: INTERNATIONAL CONFERENCE ON MODEL-DRIVEN ENGINEERING AND SOFTWARE DEVELOPMENT (MODELSWARD), 4., 2016, Rome. **Proceedings** [...]. New Jersey: IEEE, 2016. p. 779–788. Available from Internet: https://ieeexplore.ieee.org/abstract/document/7954433/. Accessed on: 22 out. 2017.

NGUYEN, T. D.; PHAM, H.-A.; THAI, M. T. Leveraging Blockchain to Enhance Data Privacy in IoT-Based Applications. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL SOCIAL NETWORKS., 2018, Shanghai. **Proceedings** [...]. New York: Springer, 2018. p. 211–221. Available from Internet: https://link.springer.com/chapter/10.1007/978-3-030-04648-4_18. Accessed on: 19 dez. 2018.

NORTA, A. Creation of smart-contracting collaborations for decentralized autonomous organizations. In: INTERNATIONAL CONFERENCE ON BUSINESS INFORMATICS RESEARCH., 2015, Stockholm. **Proceedings** [...]. New York: Springer, 2015. p. 3–17. Available from Internet: https://link.springer.com/chapter/10.1007/978-3-319-21915-8_1. Accessed on: 9 ago. 2016.

NOY, N. F.; MCGUINNESS, D. L. **Ontology development 101: A guide to creating your first ontology**. Stanford: Stanford University Press, 2001. Available from Internet: http://www.corais.org/sites/default/files/ontology_development_101_aguide_to_creating_your_first_ontology.pdf. Accessed on: 27 nov. 2015.

PADGHAM, L.; WINIKOFF, M. Prometheus: A practical agent-oriented methodology. In: AGENT-ORIENTED METHODOLOGIES. **Collection** [. . .]. Hershey: IGI Global, 2005. p. 107–135. Available from Internet: https://www.igi-global.com/chapter/agent-oriented-methodologies/5057. Accessed on: 8 mar. 2015.

PANIKKAR, S. *et al.* ADEPT: An IoT Practitioner Perspective. **Draft Copy for Advance Review, IBM**, IBM, New York, 2015. Available from Internet: https://archive.org/details/pdfy-esMcC00dKmdo53-_. Accessed on: 25 set. 2017.

PEREZ, D.; LIVSHITS, B. Smart contract vulnerabilities: does anyone care? arXiv preprint, Ithaca, 2019. Available from Internet: https://arxiv.org/abs/1902.06710. Accessed on: 8 jun. 2019.

PILKINGTON, M. 11 Blockchain technology: principles and applications. **Research handbook on digital transformations**, Edward Elgar Publishing Ltd, Cheltenham, v. 225, 2016. Available from Internet: http://papers.ssrn.com/abstract=2662660. Accessed on: 25 set. 2017.

POPEK, G. J.; KLINE, C. S. Encryption and secure computer networks. **ACM Computing Surveys (CSUR)**, CiteSeer, New York, v. 11, n. 4, p. 331–356, 1979. Available from Internet: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.127.5172&rep=rep1&type=pdf. Accessed on: 26 jan. 2015.

PRODANOV, C. C.; FREITAS, E. C. de. **Metodologia do trabalho científico: métodos e técnicas da pesquisa e do trabalho acadêmico**. 2. ed. Novo Hamburgo: Editora Feevale, 2013.

QU, S. Q.; DUMAY, J. The qualitative research interview. **Qualitative research in accounting & management**, Emerald Group Publishing Limited, Bradford, v. 8, n. 3, p. 238–264, 2011. Available from Internet: https://www.emeraldinsight.com/doi/abs/10.1108/11766091111162070. Accessed on: 2 jan. 2015.

REARDON, J. Robust key management for secure data deletion. In: SECURE DATA DELETION. **Collection** [. . .]. New York: Springer, 2016. p. 143–174. Available from Internet: https://link.springer.com/chapter/10.1007/978-3-319-28778-2_11. Accessed on: 13 out. 2017.

ROSQVIST, T.; KOSKELA, M.; HARJU, H. Software quality evaluation based on expert judgement. **Software Quality Journal**, Springer, New York, v. 11, n. 1, p. 39–55, 2003. Available from Internet: https://link.springer.com/article/10.1023/A:1023741528816. Accessed on: 8 set. 2015.

RUTA, M. *et al.* Semantic Blockchain to improve scalability in the Internet of Things. **Open Journal of Internet Of Things (OJIOT)**, Research Online Publishing, Lübeck, v. 3, n. 1, p. 46–61, 2017. Available from Internet: https://www.ronpub.com/ojiot/ojiot_2017v3i1n05_ruta.html. Accessed on: 19 jan. 2018.

SERRANO, M.; GYRARD, A. A Review of Tools for IoT Semantics and Data Streaming Analytics. **Build. Blocks IoT Anal**, River Publishers, Gistrup, v. 6, p. 139–163, 2016. Available from Internet: https://www.riverpublishers.com/downloadchapter.php?file=RP_9788793519046C6.pdf. Accessed on: 8 nov. 2017.

SFAR, A. R. *et al.* A roadmap for security challenges in the Internet of Things. **Digital Communications and Networks**, Elsevier, New York, v. 4, n. 2, p. 118–137, 2018. Available from Internet: https://www.sciencedirect.com/science/article/pii/S2352864817300214. Accessed on: 2 jan. 2019.

SHARMA, P. K. *et al.* DistBlockNet: A distributed Blockchains-based secure sdn architecture for IoT networks. **IEEE Communications Magazine**, IEEE, New Jersey, v. 55, n. 9, p. 78–85, 2017. Available from Internet: https://ieeexplore.ieee.org/abstract/document/8030491/. Accessed on: 8 set. 2018.

SIGNORINI, M. **Towards an Internet of Trust: issues and solutions for identification and authentication in the Internet of Things**. 2015. 188 p. PhD Thesis (Information and Communication Technologies) — Departament de Tecnologies de la Informació i les Comunicacions, Universitat Pompeu Fabra, Barcelona, 2016. Available from Internet: http://repositori.upf.edu/handle/10230/25746. Accessed on: 25 set. 2017.

SILVA JÚNIOR, J. F. *et al.* Perdas em sistemas de distribuição de água: estudo baseado em Revisão Sistemática. In: CONGRESSO ABES/FENASAN., 2017, São Paulo. **Anais** [...]. São Paulo: ABES, 2017. p. 1–17. Available from Internet: https://www.tratamentodeagua.com.br/wp-content/uploads/2018/10/XI-081.pdf. Accessed on: 16 out. 2017.

STALLINGS, W. **Cryptography and network security: principles and practice**. 2. ed. Upper Saddle River: Prentice Hall, 2012.

STEICHEN, M.; HOMMES, S.; STATE, R. ChainGuard—A firewall for Blockchain applications using SDN with OpenFlow. In: PRINCIPLES, SYSTEMS AND APPLICATIONS OF IP TELECOMMUNICATIONS (IPTCOMM)., 2017, Chicago. **Proceedings** [...]. New Jersey: IEEE, 2017. p. 1–8. Available from Internet: https://ieeexplore.ieee.org/abstract/document/8169748/. Accessed on: 19 dez. 2017.

SULTAN, K.; RUHI, U.; LAKHANI, R. Conceptualizing Blockchains: characteristics & applications. **arXiv preprint**, Ithaca, 2018. Available from Internet: https://arxiv.org/abs/1806.03693. Accessed on: 29 dez. 2018.

TAPSCOTT, D.; TAPSCOTT, A. **Blockchain Revolution: How the Technology Behind Bitcoin is Changing Money, Business, and the World**. 2. ed. New York: Penguin Random House LLC, 2018.

TONES-D, D. *et al.* **Ontology-based services: usage scenarios and test Ontologies**. Seattle: Semantic School, Allen Institute for Artificial Intelligence, 2007.

TRAN, Q.-N. N.; LOW, G. MOBMAS: A methodology for ontology-based multi-agent systems development. **Information and Software Technology**, Elsevier, New York, v. 50, n. 7-8, p. 697–722, 2008. Available from Internet: https://www.sciencedirect.com/science/article/pii/S0950584907000766. Accessed on: 23 set. 2017.

TVERSKY, A.; KAHNEMAN, D. Judgment under uncertainty: Heuristics and biases. **science**, American Association for the Advancement of Science, Washington, D.C., v. 185, n. 4157, p. 1124–1131, 1974. Available from Internet: https://science.sciencemag.org/content/185/4157/1124.short. Accessed on: 3 set. 2016.

UCKELMANN, D.; HARRISON, M.; MICHAHELLES, F. An architectural approach towards the future Internet of Things. In: ARCHITECTING THE INTERNET OF THINGS. **Collection** [. . .]. New York: Springer, 2011. p. 1–24. Available from Internet: https://link.springer.com/chapter/10.1007/978-3-642-19157-2_1. Accessed on: 2 fev. 2015.

UGARTE, H. A more pragmatic Web 3.0: Linked Blockchain Data. ResearchGate, Berlin, 2017. Available from Internet: https://www.researchgate.net/publication/315619465_A_more_pragmatic_Web_30_Linked_Blockchain_Data. Accessed on: 2 dez. 2017.

VIGNA, P.; CASEY, M. J. The Age of Cryptocurrency: How Bitcoin and Digital Money Are Challenging the Global Economic Order Hardcover. St. Martin's Press, New York, 2015. Available from Internet: https://pdfs.semanticscholar.org/b741/1bd26451ce64d914304b283f14be5475ff2b.pdf. Accessed on: 4 set. 2016.

WALKER, J. K. Thomas P. Keenan Memorial Lecture-The Demise of the Nation-State, the Dawn of New Paradigm Warfare, and a Future for the Profession of Arms. **AFL Rev.**, HeinOnline, Buffalo, v. 51, p. 323, 2001. Available from Internet: https://heinonline.org/hol-cgi-bin/get_pdf.cgi?handle=hein.journals/airfor51&section=13. Accessed on: 2 jun. 2019.

WANG, W. *et al.* A survey on consensus mechanisms and mining management in Blockchain networks. arXiv preprint, Ithaca, p. 1–33, 2018. Available from Internet: https://www.researchgate.net/profile/Zehui_Xiong2/publication/325034085_A_Survey_on_Consensus_Mechanisms_and_Mining_Management_in_Blockchain_Networks/links/5b0530040f7e9b24a2af6f5f/A-Survey-on-Consensus-Mechanisms-and-Mining-Management-in-Blockchain-Networks.pdf. Accessed on: 2 dez. 2018.

WANJAWA, B. W.; MUCHEMI, L. Automatic Semantic Network Generation from Unstructured Documents–The Options. In: INTERNATIONAL CONFERENCE ON SOFT COMPUTING & MACHINE INTELLIGENCE (ISCMI), 5., 2018, Nairobi. **Proceedings** [. . .]. New Jersey: IEEE, 2018. p. 72–78. Available from Internet: https://ieeexplore.ieee.org/abstract/document/8703225/. Accessed on: 2 jan. 2019.

WEBER, R. H. Internet of Things–New security and privacy challenges. **Computer law & security review**, Elsevier, New York, v. 26, n. 1, p. 23–30, 2010. Available from Internet: https://www.sciencedirect.com/science/article/pii/S0267364909001939. Accessed on: 4 nov. 2016.

WILKINSON, S. Focus groups: A feminist method. **Psychology of women quarterly**, Wiley Online Library, Hoboken, v. 23, n. 2, p. 221–244, 1999. Available from Internet: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1471-6402.1999.tb00355.x. Accessed on: 3 nov. 2015.

WILLNER, A. *et al.* The open-multinet upper ontology towards the semantic-based management of federated infrastructures. **EAI Endorsed Transactions on Scalable Information Systems**, European Alliance for Innovation (EAI), Ghent, v. 2, n. 7, 2015. Available from Internet: http://search.proquest.com/openview/3c9f918bd0155a204c48138240304b98/1?pq-origsite=gscholar&cbl=4477225. Accessed on: 3 set. 2016.

WÜST, K.; GERVAIS, A. Do you need a Blockchain? In: CRYPTO VALLEY CONFERENCE ON BLOCKCHAIN TECHNOLOGY (CVCBT)., 2018, Zug. **Proceedings** [. . .]. New Jersey: IEEE, 2018. p. 45–54. Available from Internet: https://ieeexplore.ieee.org/abstract/document/8525392/. Accessed on: 3 dez. 2018.

XU, L. D.; HE, W.; LI, S. Internet of Things in industries: a survey. **IEEE Transactions on industrial informatics**, IEEE, New Jersey, v. 10, n. 4, p. 2233–2243, 2014. Available from Internet: https://ieeexplore.ieee.org/abstract/document/6714496/. Accessed on: 2 out. 2015.

XU, Y. *et al.* A Blockchain-based Non-Repudiation Network Computing Service Scheme for Industrial IoT. **IEEE Transactions on Industrial Informatics**, IEEE, New Jersey, 2019. Available from Internet: https://ieeexplore.ieee.org/abstract/document/8633331/. Accessed on: 8 jun. 2019.

YUE, X. *et al.* Healthcare data gateways: found healthcare intelligence on Blockchain with novel privacy risk control. **Journal of medical systems**, Springer, New York, v. 40, n. 10, p. 218, 2016. Available from Internet: https://link.springer.com/article/10.1007/s10916-016-0574-6. Accessed on: 2 nov. 2017.

ZAMBONELLI, F.; JENNINGS, N. R.; WOOLDRIDGE, M. Multi-agent systems as computational organizations: the Gaia methodology. In: AGENT-ORIENTED METHODOLOGIES. **Collection** [. . .]. Hershey: IGI Global, 2005. p. 136–171. Available from Internet: https://www.igi-global.com/chapter/agent-oriented-methodologies/5058. Accessed on: 9 ago. 2016.

ZHANG, Y.; WEN, J. An IoT electric business model based on the protocol of Bitcoin. In: INTERNATIONAL CONFERENCE ON INTELLIGENCE IN NEXT GENERATION NETWORKS, 18., 2015, Paris. **Proceedings** [. . .]. New Jersey: IEEE, 2015. p. 184–191. Available from Internet: https://ieeexplore.ieee.org/abstract/document/7073830/. Accessed on: 9 nov. 2016.

ZHANG, Y.; WEN, J. The IoT electric business model: Using blockchain technology for the Internet of Things. **Peer-to-Peer Networking and Applications**, Springer, New York, v. 10, n. 4, p. 983–994, 2017. Available from Internet: https://link.springer.com/article/10.1007/s12083-016-0456-1. Accessed on: 25 set. 2018.

ZHENG, G.; ZHANG, Z. Intelligent wireless electric power management and control system based on ZigBee technology. In: INTERNATIONAL CONFERENCE ON TRANSPORTATION, MECHANICAL, AND ELECTRICAL ENGINEERING (TMEE)., 2011, Changchun. **Proceedings** [. . .]. New Jersey: IEEE, 2011. p. 1120–1124. Available from Internet: https://ieeexplore.ieee.org/abstract/document/6199401/. Accessed on: 2 mar. 2015.

ZHOU, L. *et al.* BeeKeeper 2.0: confidential Blockchain-enabled IoT system with fully Homomorphic Computation. **Sensors**, MDPI, Basel, v. 18, n. 11, p. 3785, 2018. Available from Internet: https://www.mdpi.com/1424-8220/18/11/3785. Accessed on: 8 dez. 2018.

ZYSKIND, G.; NATHAN, O. *et al.* Decentralizing privacy: using Blockchain to protect personal data. In: IEEE SECURITY AND PRIVACY WORKSHOPS., 2015, San Jose. **Proceedings** [. . .]. New Jersey: IEEE, 2015. p. 180–184. Available from Internet: https://ieeexplore.ieee.org/abstract/document/7163223/. Accessed on: 2 nov. 2016.

ZYSKIND, G.; NATHAN, O.; PENTLAND, A. Enigma: decentralized computation platform with guaranteed privacy. **arXiv preprint**, Ithaca, 2015. Available from Internet: https://arxiv.org/abs/1506.03471. Accessed on: 2 fev. 2016.

APPENDIX A – SURVEY QUESTIONNAIRE

GENERAL VIEW OF THE RESEARCH

My name is Sérgio Francisco T. de O. Mendonça, and I am a Doctorate candidate in Electrical Engineering in the Post-Graduate Program in Electrical Engineering in Department of Electrical Engineering, School of Engineering of the Federal University of Pernambuco, under the guidance of the Professor Ph.D. Fernanda Maria Ribeiro de Alencar.

First of all, I would like to thank you for volunteering to answer this research. Your feedback is extremely valuable to the conclusion of this work. The following questionnaire finishes one more evaluation step of this Doctorate Work, which has as basis the elaboration of a knowledge model for designing in Blockchain-based Internet of Things solutions.

Blockchain has recently attracted the interest of stakeholders across a wide a span of several industries, from finance and healthcare to utilities, real estate, and the government sector. This research aims to identify the objectives, entities, and processes that involve the development of security and privacy on IoT development, to ensure an architecture resilience to attacks, data authentication, access control, and client privacy need to be established. Since business processes are concerned, a high degree of reliability. Identifying how blockchain can help those goals.

About 12 minutes are necessary for reading and answering this questionnaire completely. We kindly ask you to COMPLETELY answer it. On the contrary, we will need to discard it, once the incomplete questionnaires will not be considered valid to our research. Your personal data will be kept in confidence, and your contributions will be used only for academic purposes. In the end, we will present our knowledge model for designing in Blockchain-based Internet of Things called BIoT (Blockchain-based Internet of Things Model).

Regards,

Sérgio Francisco T. de O. Mendonça

ABOUT YOURSELF AND YOUR EXPERIENCE

1. What is your name?
2. What is your e-mail?
3. How old are you?
4. What is your gender?
    a) Female
    b) Male
    c) I would prefer not to answer that question.
5. What City do you live in?
6. What State do you live in?
7. What Country do you live in?
8. Which is your schooling?
    a) Undergraduate/Bachelor Degree
    b) Graduate (*lato sensu*)
    c) Master Degree
    d) Doctorate Degree
    e) Post-Doctorate Degree
9. What is your area of expertise?
    a) Computing
    b) Electical/Electronics Engineering
    c) Management
    d) Other
10. What is your present position?
    a) Project Manager
    b) Technical Leader
    c) Software Developer
    d) Software Tester
    e) Requirements Analyst
    f) Researcher
    g) Embedded Systems Expert
    h) Internet of Things Expert
    i) Blockchain Expert
    j) Other:
11. What type of expertise do you have in Devices Desinging?
    a) Industry
    b) Academy
    c) Both
12. How long is your Device Designing experience?

a) 1 - 3 years
b) 4 - 10 years
c) more than 10 years

13. What type of experience do you have in Safety Requirements for Devices?
    a) Industry
    b) Academy
    c) Both
14. What type of expertise do you have in Internet of Things?
    a) Industry
    b) Academy
    c) Both
15. What is the name of the company you currently work for?
16. How big is the company you work for?
    a) Micro (up to 9 employees)
    b) Small (from 10 to 49 employees)
    c) Medium (from 50 to 99 employees)
    d) Large (more than 100 employees)
    e) I am a Student
    f) Other

WHEN WE TALK ABOUT SECURITY CONCEPTS FOR THE INTERNET OF THINGS

17. How is confidentiality treated in your IoT solution?
18. How is integrity treated in your IoT solution?
19. How is the availability of your IoT solution handled?
20. How is authenticity treated in your IoT solution?
21. How is accountability addressed in your IoT solution?
22. Is there any process or workflow that the company adopted to design devices based on Internet of Things conceptions?
23. What is the degree of data integrity (resilience to attacks) on IoT development?
    a) 1
    b) 2
    c) 3
    d) 4
    e) 5
24. How to ensure data integrity (resilence to attacks) on IoT development?
25. What is the degree of address and object information (data authentication) were authenticated on IoT development?
    a) 1

b) 2
c) 3
d) 4
e) 5

26. How to ensure that the address and object information (data authentication) were authenticated on IoT development?
27. What is the degree of adaptability (access control) on IoT development?
    a) 1
    b) 2
    c) 3
    d) 4
    e) 5
28. How to ensure adaptability (access control) on IoT development?
29. What is the degree of anonymity (client privacy) on IoT development?
    a) 1
    b) 2
    c) 3
    d) 4
    e) 5
30. How to ensure anonymity (client privacy) on IoT development?

WHAT ARE THE USE CASES OF THE BLOCKCHAIN BEYOND CRYPTOCURRENCIES?

31. What are the use cases of the blockchain beyond cryptocurrencies?
32. Are there any use cases applicable to the IoT?
33. What are the implementation differences with respect to the Bitcoin blockchain?
34. Which data are stored in the blockchain?
35. Which mining techniques are used?
36. Would you adopt a blockchain-based solution for Internet of Things development?
    a) Yes
    b) No
    c) Maybe
37. If you wish, justify the answer of the previous question.
38. When integrating blockchain, if end nodes have to interact with the blockchain, cryptographic functionality could be provided in IoT devices or in Middleware?
39. In your opinion, what benefit you identify to adopt a blockchain-based solution for Internet of Things development brings to the project or to the business?
40. When high performance is required, blockchain alone may not be a correct solution, but can a hybrid approach be applied to optimize it?

APPENDIX B – STUDY QUALITY ASSESSMENT

Table 15 – Study Quality Assessment Form

Quality Criteria	**0**	**0.5**	**1**
QC1: Are the Inclusion and Exclusion criteria rightly described and suitable?			
QC2: Did the literature research potentially include all relevant investigations?			
QC3: Did the included studies were evaluated in quality and validity aspects?			
QC4: Has the study base been adequately described?			

Source – The Author.

APPENDIX C – BLOCKCHAIN-BASED IOT CODES

The HTTP verbs run in a blockchain network structure, as we can see in the following API codes:

Listing C.1 – Block structure on BIoT API.

```
const sha256 = require('crypto-js/sha256')
class Block {
    constructor(index = 0, previousHash = null,
                    data = 'GenesisBlock') {
    this.index = index
    this.previousHash = previousHash
    this.data = data
    this.timestamp = new Date()
    this.hash = this.generateHash()
    }
    generateHash() {
                return sha256(
                      this.index +
                      this.previousHash +
                      JSON.stringify(this.data) +
                      this.timestamp).toString()
    }
}
module.exports = Block
```

Listing C.2 – Blockchain (local) on BIoT API.

```
const Block = require('./block')

class Blockchain {
    constructor() {
        this.blocks = [new Block()]
        this.index = 1
    }

    getLastBlock() {
        return this.blocks[this.blocks.length - 1]
    }

    addBlock(data) {
        const index = this.index
        const previousHash = this.getLastBlock().hash

        const block = new Block(index, previousHash, data)

        this.index++
        this.blocks.push(block)
    }
}

module.exports = Blockchain
```

Listing C.3 – Hash validation on BIoT API.

```
# example of iterating a nonce in a hashing algorithm's input
import hashlib

text = "I am Satoshi Nakamoto"

# iterate nonce from 0 to 19
for nonce in xrange(20):
    # add the nonce to the end of the text
    input = text + str(nonce)
    # calculate the SHA-256 hash of the input (text+nonce)
    hash = hashlib.sha256(input).hexdigest()
    # show the input and hash result
    print input, '=>', hash
```

Listing C.4 – Blockchain integrity on BIoT API.

```
isValid() {
    for (let i = 1; i < this.blocks.length; i++) {
        const currentBlock = this.blocks[i]
        const previousBlock = this.blocks[i - 1]

        if (currentBlock.hash !== currentBlock.generateHash())
        ↪ {
            return false;
        }

        if (currentBlock.index !== previousBlock.index + 1) {
            return false;
        }

        if (currentBlock.previousHash !== previousBlock.hash) {
            return false;
        }
    }
    return true;
}
```

Listing C.5 – Analog to Digital Acquisition on BIoT API.

```
import streams  # import the streams module
import adc      # import the adc driver

# create a stream linked to the default serial port
streams.serial()

while True:

# Basic usage of ADC for acquiring the analog signal from a pin
    value = adc.read(A0)
    print("One sample:",value)

# The complete definition of adc.read()
# is adc.read(pin, samples=1)
# For an advanced usage of adc.read refer
# to the official Zerynth documentation

#acquire 10 samples with default sampling period
    value2 = adc.read(A0,10)
    print("10 samples:\n",value2)

# acquire 3 samples from the first 4 analog
# pins of the board with default sampling period
    value3= adc.read([A0,A1,A2,A3],3)
    print("3 samples from A0, A1, A2 and A3:\n",value3)

    print()
    sleep(300)
```

Listing C.6 – Crypto Hash on BIoT API.

```
import streams

# import all supported hash functions
from crypto.hash import md5 as md5
from crypto.hash import sha1 as sha1
from crypto.hash import sha2 as sha2
from crypto.hash import sha3 as sha3
from crypto.hash import keccak as keccak
# also import HMAC
from crypto.hash import hmac as hmac

# open stdout
streams.serial()

message = "Zerynth"

while True:
    try:
        ss = md5.MD5()
        ss.update(message)
        print("MD5: ",ss.hexdigest())

        ss = sha1.SHA1()
        ss.update(message)
        print("SHA1:",ss.hexdigest())

        ss = sha2.SHA2(sha2.SHA512)
        ss.update(message)
        print("SHA2:",ss.hexdigest())

        ss = sha3.SHA3()
        ss.update(message)
        print("SHA3:",ss.hexdigest())

        ss = keccak.Keccak()
        ss.update(message)
        print("KECCAK:",ss.hexdigest())

        # generate a hmac with key="Python"
        # and sha1 hash
        hh = hmac.HMAC("Python",sha1.SHA1())
        hh.update(message)
        print("HMAC:",hh.hexdigest())
    except Exception as e:
        print(e)
    sleep(2000)
```