QALAAI ZANIST JOURNAL
A Scientific Quarterly Refereed Journal Issued by Lebanese French University – Erbil – Kurdistan – Iraq
Vol. (2), No. (2), April 2017
Speical Issue : The 1st International Conference on Information Technology (ICoIT'17)
ISSN 2518-6566 (Online) - ISSN 2518-6558 (Print)

Internet of Things (IoT): Readme

Istabraq M. Al-Jeboury
Al-Nahrain University, College of Information Engineering, Department of Networks Engineering, Baghdad, Iraq
estabriq_94@coie-nahrain.edu.iq

Emad H. Al-Hemiary
Al-Nahrain University, College of Information Engineering, Department of Networks Engineering, Baghdad, Iraq
emad@coie-nahrain.edu.iq

ARTICLE INFO

Article History:

Received: 19 March 2017

Accepted: 1 April 2017

Published: 10 April 2017

DOI:

10.25212/lfu.qzj.2.2.35

Keywords: *Internet of Things (IoT), Internet of Everything (IoE), Machine to Machine (M2M), IoT architecture, IoT protocols, Cloud Computing, Microcontrollers*

ABSTRACT

Internet of Things (IoT) integrates billions of the heterogeneous IoT things with the Internet in which the embedded systems such as sensors and actuators linked together to improve quality of life, and becomes the future of technologies in any field of human daily life. These IoT devices cooperate with each other and generate useful information to provide better services and applications to the governments and the society. Also, there is a need to store these data on Cloud for monitoring. This paper, surveys IoT applications, new challenges and issues arise in different fields and provides IoT architecture, focuses on explanation of IoT protocols and their operations and functionalities, presents different microcontroller types used by researchers. With the huge amount of data generated from IoT devices, the integrating Cloud and IoT may helpful, Therefore, a survey on open issues faced when these two concepts integrating together is discussed. The objective of this paper is to provide a survey for everything related to IoT and direct it to all beginners in this filed or academic researchers.

1. INTRODUCTION

The IoT is a field that consists of the newest technologies make a revolution of Internet. It includes a wide range of applications such as smart cities, e-health, smart home, smart grid and where it monitors smart road, smart waste and patients in real time [1]. The term "things" consists of interconnected smart devices like sensors and actuators. These devices are communicating with each other, referred to as Machine to Machine communication (M2M) [2]. With the rapid growth of the Internet, Cisco estimates that the world will have 50 billion of IoT devices within the next 20 years. Also, these devices have sensing feature such as temperature sensor in order to measure Room temperature [3]. Thus, a large number of data needs to be stored on servers. This will lead to integration of IoT and Cloud Computing, which Cloud offers services, tools and software to end users [4].

The phrase of " Internet of Things" was created by Kevien Ashton on supply-chain management where he was cofounder and executive director of the MIT Auto-ID Center. He said regarding to IoT: "I could be wrong, but I'm fairly sure the phrase "Internet of Things" started life as the title of a presentation I made at Procter & Gamble (P&G) in 1999. Linking the new idea of RFID in P&G's supply chain to the then-red-hot topic of the Internet was more than just a good way to get executive attention. It summed up an important insight which is still often misunderstood."[5]. After years, it was introduced by David L. Brock in 2001. The "thing" is a part of people`s life to interact within the physical world that surrounds us. The phrase " Auto-ID" includes any identification technology such as bar codes and smart cards that are used for reducing errors, automating, and increasing efficiency. In 2003, Radio-Frequency Identification (RFID) application expanded into wider fields at the MIT`s Auto-ID center, which built the basis for the current IoT sight [6].

People have an addiction to purchase multiple devices, (I Pads, mobile handsets, Tablets, Laptops, etc.) therefore, the number of devices have been increased and is expected to exceed the number of people [7]. Data that are extracted from these devices have an impact on society and economy. Thus, IoT is easy to monitor operations and manage them far away from organization, track data as it moves from one place to another and even if it crosses the ocean and detect the changes in the human body like temperature and heartbeat that could save lives [8]. IoT includes essential types of networks like distributed, grid, ubiquitous, and vehicular; these have turned towards the world of IT with the five coming years [9].

The original Internet was formed its architecture before the explosion of huge and smaller devices such as sensors/actuators. With the emergence of IoT, it destroys all previous ideas regarding network architecture. Thus, create challenges for the current networking paradigm [10]. IoT develops the enormous applications which makes use of data generated from different objects that interact and cooperate with each other to provide better services to governments, organizations and humans.

The rest of this paper is organized as follows: Section 2, discusses the architecture of IoT. Section 3, presents protocols and standards involved in IoT and their challenges. Section 4, classifies the most important microcontrollers. Section 5, introduces the integration between cloud and IoT paradigm. Section 7, focuses anda derives the open issues in Cloud of Everthing.

2. IOT ARCHITECTURE

The IoT handles billions of different objects that integrate and corporate with each other to provide the scalability, configuration and modularity of IoT deployments in different fields, therefore, the layered architecture has taken into consider. It depends on various kinds of technologies supporting the IoT and describes how these technologies related to each other. The basic IoT layers consists of three model [11] are: Application, Network, and Perception Layers. new layers are added [12]: Business and Middleware Layers are recently proposed.

2.1. PERCEPTION LAYER

The perception or so-called object/ sensor connectivity layer it's called that, since it represents the smart objects that are integrated with IoT sensors. Real-time information can be collected and processed by enabling the interconnection of the digital/physical world by sensors. The shrinking of hardware size was to produce a smaller form and more powerful sensors. The sensors can be different types for different purposes. The aim of this layer is collecting the information from different devices like sensors/actuators, tags (RFID and Barcode, GPS, camera), creating big data and then performing different functionalities like taking the measurements such as temperature, humidity, movement and electricity. Sensors have a number of features and characteristics, for instant: they have a memory-sized for recording a certain number of measurements. Furthermore, they have an ability to convert signals from digital to analog that can be understood by devices [13]. Most sensors need a gateway for aggregating data, which require a connectivity such as LAN (Local Area Network): Wi-Fi, Ethernet and PAN (Personal Area Network): Bluetooth and ZigBee. Also, sensors sent the data directly to servers or applications using WAN (Wide Area Network): GPRS, GSM and LTE. WSNs are part of IoT, these sensors use low-power and low-data-rate. The WSNs are better than the traditional sensors because they keep sufficient battery life and covering large areas.

2.2. NETWORK LAYER

The Network or so-called the Object Abstraction Layer. Massive volume of data is produced from the first layer, that require a high-performance transport medium (gateway) to collect the information and send to Middleware. Gateways refer to WAN, Mobile Communication Network, Wireless/ Wired Network as well as provide routing protocols and reliable delivery. Networks may include several of technologies and protocols to support IoT and M2M applications to provide new capacities such as context-aware and to support the communication requirements such as delay, bandwidth, security and privacy. Hence, Network layer of IoT model is similar to both Network and transport layers of OSI (Open System Interception) model. The aim of this layer is to transmit the data to Internet through different technologies like Wi-Fi, 3G, Bluetooth, XBee, etc. [13].

2.3. MIDDLEWARE LAYER

Middleware Layer or so-called Service management layer, it makes the processing of information possible. It provides the important functions like device modeling, management, analytics and configuration, etc. The aim of this layer is to receive and store the date form the pervious layer, processes the information, make decisions and then deliver it to the application layer. It provides information in the form of events for example good temperature for the patient`s health condition. These data may require an immediate response to emergency`s station. In addition, hiding technological detail from the user to enable the developers to develop applications more easily, quickly and without the complexity. At this layer, some middleware protocols may be included in order to make applications compatible with different objects without consideration according to hardware platforms. Cloud computing is an important platform of Middleware Layer that it may analyze the data [14].

2.4. APPLICATION LAYER

Application layer provides different services to the end user after the processing and analyzing the data from Middleware Layer. For instant, after a patient measures his blood glucose and temperature the application layer can provide these measurements to who asked for them. The application Layer presents several of fields like smart home, smart city, smart building, smart healthcare, education, industry, etc. [15].

2.5. BUSINESS LAYER

As the name suggests, its purpose is to build a business model. The data from the application Layer will be processed more in order to provide significant services to meet customer needs. In addition, Business layer can provide different functionalities such as analyzing, monitoring and evaluating data. It also enhances the service by comparing the output of each layer with the expected output, which can make money from the service provided. [15].

There are some problems with IoT architecture which still get the attention from researchers:

- *Vertical vs. Horizontal [1]:*

  In vertical, operation and control of various services of different layers (from the perception layer to the application layer) should studied.

  In horizontal, different devices and technologies are used, so that further enhancements of user-centric communication capabilities should considered (one object to another object).

- *Security:*

  Security must be included across IoT layers from object layer to application layer. Security is an important issue as the integrity of data, since the integrated physical sensors are able to access applications organizations and domains.

3. PROTOCOLS

In this section, an overview is provided of the most common protocols specified for IoT, by giving their functionalities and their challenges**.**

3.1. APPLICATION PROTOCOLS

Two of the most common protocols are motioned, that are used for constrained devices:

3.1.1 MESSAGE QUEUE TELEMETRY TRANSPORT (MQTT):

The history of MQTT was introduced by Andy Stanford-Clark and Arlen Nipper in 1999 and then MQTT v3.1.1 becomes an open standard protocol by OASIS (Advancing Open Standards for the Information Society) in 2014. It is a lightweight application protocol optimized for devices with limited constrained such as power and bandwidth. Also, it is designed for IoT applications and M2M [16].

MQTT runs over TCP/IP and based on publish/subscribe model topic-based. In this mode, it consists of two types of clients and server. Each client can be publisher or subscriber to specific topic, while server or called broker mediates publishers and subscribers and pass messages form publishers to subscriber. The subscribers receive messages related to subscribed topics [17].

MQTT is better than the HTTP protocol because it has a lower overhead, a synchronous and reliable with different levels of quality of services used for reliable networks. QoS level 0, the client receives the message at most once, an acknowledgement not used in this level, and the server does not retry sending the message at all. QoS level 1; the server delivers a message to the client at least once, here an acknowledgement used in this level. QoS level 2, this level is used when lost or duplicated of the message that is accepted by the client, this requires 4-way handshake to deliver the message at exactly once, hence this coz increase in the overhead [16].

3.1.2 CONSTRAINED APPLICATION PROCOLCOL (CoAP):

CoAP is web transfer protocol based on the Representational State Transfer [REST] architecture for Resource-Oriented such as 6LowPAN networks and nodes. In 2010, CoAP was created by CoRE (The Constrained RESTful Environments) working group. The aim of this protocol was to optimize application for IoT and M2M by reducing message overhead [18]. CoAP uses the same functionality as the HTTP protocol (supports request/response model), but CoAP adds some additional features such as low processing power and energy consumption; so that the protocol can be suitable for constrained environments such as sensors/actuators. Thereby, it will offer a new feature like multicast, asynchronous/synchronous message exchange, low-bandwidth and reliable transmission (the bandwidth and data can be exchanged via the client-server model). CoAP runs overTCP so that it will support unicast only, while running over UDP, it will support both unicast and multicast [19]. All above features make the use of CoAP more favorable than using traditional HTTP [20].

CoAP supports publish/subscribe model by using messaging queue (MQ-CoAP). Here, MQ-CoAP is using URI (Universal Resource Identifier) to access resource on the destination instead of using topics like MQTT. each specific resource has a unique URI. CoAP has two layers: request/response layer and transaction layer. In the Request/Response layer, it is responsible for influencing the resource by using these commands (GET, PUT, POST and DELETE) to provide resource oriented between client and server. Publishers send the data to subscribers which request the data through URI [21]. While in the transaction layer, it is a reliable because it has multiple different levels of QoS. This layer is used to exchange a single message between client and server. There are four types of messages that require QoS. The CON (Confirmable), is required to return an acknowledgment to the client that could be sent as synchronous or asynchronous. The NON (Non- Confirmable), does not require to return an acknowledgment. The ACK (Acknowledgment), is used to confirm the reception of a CON message type. The RST (Rest), is used if the message cannot be processed.

HTTP-CoAP mapping offers transformation between Client HTTP and CoAP server by allowing a HTTP client to access resource at the CoAP server using a proxy, but if the process of mapping is reversed, a reverse proxy will be used [20].

CoAP is designed with no security feature, therefore the IETF proposed two protocols to secure CoAP: DTLS and IPsec. Recently, IPSec stops securing CoAP. CoAP with DTLS (Datagram Transport Layer Security) is similar to HTTPs. DTLS functionality is similar to TLS (Transport Layer Security) functionality. However, the difference between TLS and DTLS is that DTLS runs on top of UDP. DTLS provides authentication, data integrity, confidentiality and automatic key management. But the problem with DLTS is that it's unsuitable for constrained devices because it does not support multicast, which is the important role in CoAP [22]. Some researchers have proposed some solutions to that problem [23].

There are many challenges that CoAP that may face, some of them are mentioned:

- *Congestion control:* CoAP runs on the top of UDP protocol, UDP does not provide reliable or congestion control mechanism, therefore, CoAP must deal with the congestion control mechanism by itself. It has a basic mechanism and it is achieved by "Confirmable" and "NON Confirmable" messages. However, this mechanism could not be handled with constrained resources. Some researchers have proposed some solution to that problem [24].

- *Size of MTU:* constrained environments have very small size of MTU. The frame size of IEEE 802.15.4 is 127 bytes as a maximum, while the size of MTU of IPv6 is 1,280 bytes as a minimum, but the IP network supports MTU 1500 byte or higher. 6LowPAN provides an adoption layer between the link layer and the network layer to provide fragmentation method and header compression to recover the above problem. Some researchers have proposed some solution to that problem [25].

TABLE 1: MQTT vs. CoAP

	MQTT	**CoAP**
Messaging transformation	Multipoint-to-Multipoint commination	Point-to-Point communication
	Topics	URI
Transport	TCP-Based	UDP-Based
Architecture	Publish/Subscribe model	Request/Response model
Layers	Single layered	Two sublayer: Request/Response and Transaction
Security	SSL/TLS	SSL/DTLS
Reliability	3 QoS levels	4 QoS levels
Performance analysis	Lower delays	Lower overhead and lower packet loss

3.2. INFRASTRUCRE PROTOCOLS

3.1.3 ROUTING PROTOCOL FOR LOW POWER AND LOSSY NETWORKS (RPL)

RPL is a distance vector routing protocol for LLNs (Low power and Lossy Networks). LLNs consist of thousands of embedded devices which are constrained like limited power, memory and processing resource. Routers are interconnected by lossy links; however, these links have a problem on a low data rate. In addition, LLNs support three types of traffic: point-to-point (transfer traffic between nodes), point-to-multipoint (transfer traffic from root to other nodes) and multipoint-to-point (transfer traffic from nodes to root) [26]. For the above reasons, LLNs must have specific routing requirements. The previous routing protocols were like OSPF, IS-IS, AVOD and OSLR but all these protocols did not meet LLN routing requirements. IETF forms a new Working Group to meet LLN solutions. The Routing Over Low power and Lossy (ROLL) proposed an RPL routing protocol for various application scenarios like healthcare, industrial, urban, smart home, smart cities and for WSNs and IoT. RPL was standardized in 2011 and defined as RFC6650. RPL is committed to IPv6-based to prevent fragmentation and intends to support a different kinds of link layer. RPL supports various technologies such as Wi Fi and IEEE 802.15.4 [27]. RPL routing has two attributes that are used for path calculation: the first attribute is constraints, the node and link clipped from the path that did not commit the constraint; the second attribute is metrics that determine the least-cost path [28].

Objective Function uses metrics and constraints to determine the best path [29]. RPL mechanism is based on Directed Acyclic Graphs (DAGs) which consists of Destination Oriented Directed Acyclic Graphs (DODAGs). DODAGs have one root that is called LBR (LowPAN Border Router) or it can have a multiple root in the network. In addition, they have a preferred parent node in which each router can selects its parent. Furthermore, they have a several sink routers that are called a leaf node. The DODAG root advertises the routing metrics and filters the link to look as if it fulfills the properties [30].

RPL supports new messages of Internet Control Message Protocol (ICMPv6) for the IPv6 specification to exchange related information [31]. These messages are DIO (DODAG Information Object), DIS (DODAG Information Solicitation) and DAO (DODAG Information Object). RPL uses Trickle algorithms to exchange messages dynamically between nodes. [50]. The DIO message is used to advertise the table of neighboring nodes of the graph, determine distance from each node to the root based on certain rules (objective function, DAG characteristics, advertised path cost and local policy), and to make a decision whatever the node is to join the network or not. The DAO message is used to advertise the upward traffic and downward traffic. The DIO message is used by a node to collect DIO messages from neighboring nodes.

3.3. IPv6

IPv6 was created by the IETF and defined as RFC-791. It is considered as an interactive of IPV4. It is similar to IPv4, but adds some additional features such as the length of IP address is longer than IPv4 and it is extended from 32bit into 64bit. IPv6 becomes the future of IoT.

The following aspects of IPv6 make it suitable to be used by IoT:

- The IP is a requirement to any connection that is used to transfer data between devices over the internet.
- It offers scalability by providing 2^128 unique address space which covers all old and future devices.
- It supports the Network Address Translation (NAT) which enables multiple users to share the same public IP address that solves the limitation on IPv4 problem.
- It provides solutions to support mobility of end-to-end devices and routing mobility of the network.
- It provides the stateless mechanism, in another word the nodes can define their address and hence reduce the cost and configuration effort.

If the number of devices is increased, then a security issue is considered, such as network attack. For example; in the case of smart house, there will be a security risk on information, therefore IPv6 provides security such as IPSec and VPN (Virtual Private Network) where the last is supported by all compatible devices**.**

ISP (Internet Service Provider) may suffer from a problem, when the network is not supporting IPv6. Some researchers have proposed some solutions to that problem [32].

3.4. IEEE802.15.4

The IEEE802.15.4 is an IETF standardized protocol for constrained devices such as sensors/actuators, designed for power consumption, low data rate, low cost, low complexity and a few meters of radio frequency; specified for low-rate wireless personal area networks (LR-WPANs), IoT and M2M. It designates a physical layer and Medium Access Control (MAC) sublayer [33]. IEEE802.15.4 supports two types of devices: a full-function device (FFD) and a reduced-function device (RFD). The FFD device serves as a personal area network (PAN) coordinator or just a coordinator. On the other hand, the RFD cannot serve as PAN coordinator or just a coordinator. The RFD is designed for simple node like a switch or a passive infrared sensor with limited resources and memory. The network has at least one FFD and multiple RFD which can communicate with each other. IEEE802.15.4 consists of two types of topologies and operates one of them: the star topology or the peer to peer topology (or called mesh). The star topology contains of one central FFD and some RFD devices. The FFDs control, initiate, terminate and manage the other RFD devices. The star topology offers services in the fields such as smart home, healthcare and computers. While the mesh topology contains one FFD coordinator and multiple RFD devices which can communicate with each other through intermediate nodes. The mesh topology is more complex than star topology. It supports applications like WSN and industrial. Cluster tree topology is extended from mesh; the most devices are FFD. The other devices from outside the network cannot associate with cluster tree due to RFD devices at the end of the cluster [34].

There are other protocols developed to support IEEE802.15.4 such as ZigBee, WiFi and WirelessHART [35].

4. MICROCONTROLLER

In this section, we consider the microcontroller as one of the most important and fundamental elements of IoT, by giving the main types used by researchers, where the selection of microcontrollers depends on many factors to choose the suitable one for a project for example, processing speed, energy computation and the supported programming languages, also we mention some important characteristics for micro-controllers, with comparison between some important versions which related to micro-controllers and to make the selection and learning easier. Also, some simulators programs which are available are mentioned with books.

4.1. Arduino

Arduino is an open-source microcontroller used by teachers and students to prototype a powerful, low cost and flexible electronic project. Arduino consists of hardware and software that is called IDE (Integrated Development Environments). It takes input from different sensors like light, temperature, degree of flex, pressure, proximity, acceleration, carbon monoxide, radioactivity, humidity and barometric pressure, and then sends them to the Arduino board to convert it to digital output. Arduino`s software used to control the board to take different actions. It supports variety of products like: IoT application, wearable, entry level and enhanced features. In addition, it is compatibles with shields that are used to provide additional capabilities such as connecting to the Internet (Wi-Fi, Bluetooth, 6LowPAN) [36].

TABLE 2: Arduino features

Board	Features	Communication	I/O
Arduino PRIMO	ARM, 7-15V, 64MHZ	ETH, BLE, Wi-Fi, CAN	6 ADC
Arduino STAR - OTTO	ARM, 3.3, 180MHZ	ETH, BLE, Wi-Fi, 2CAN	2 ADC
Arduino UNO WIFI	AVR, ESP8266, 5V, 16-80MHZ	ETH, BLE, Wi-Fi, CAN	ADC
Arduino INDUSTRIAL 101	ARM, 5V, 16MHZ	ETH, Wi-Fi, CAN	-

4.2. Raspberry Pi

Raspberry Pi is a small-sized computer and low-cost device intended for the education field. It is used by people to improve programming skills. The Raspberry Pi advances the Arduino microcontroller. The physical hardware is an open source and run on Linux, Windows and Risc OS. The board consists of important things: processor, RAM and ports. There are different types of models. The first is the Raspberry Pi A, this model is low price, has only one USB port, uses less power computation and only 256 MB of RAM. the Raspberry Pi A is upgraded to the Raspberry Pi A+, suitable for robotics and embedded system. The second is the Raspberry Pi B, this model has two ports one for the Ethernet and the other one for 512 MB of RAM. The Raspberry Pi B is also upgraded to the Raspberry Pi B+, this model has a double number of USB ports and improved power consumption. The third is the Raspberry Pi 2, is the replacement of B+, the important features including 1GB of RAM and 4 USB ports. The Raspberry Pi 3 is the last model, supports important features like it has built such as Wi-Fi IEEE 802.11n and Bluetooth [37]. There are several book`s guides for the Raspberry Pi in [38]. Also, Raspberry Pi simulators in [39].

4.3. TelosB

TelosB sensor is an open-source platform used in WSNs. The platform includes: USB programming and data collecting capability, microcontroller, transceiver, 10KB RAM, 250Kbps and IEEE802.15.4 radio with antenna built-in and low power consumption (which offers a long-life battery and fast wake ups from deep sleep). It uses an open-source, simple and energy efficient TinyOs image, which was developed by US Berkeley. It is integrated with five sensors: Temperature, Sound, Solar radiation, Light and Humidity sensors [40].

4.4. NodeMCU-ESP8266

The Node MCU is an open-source microcontroller that is used in IoT applications and was designed by Espressif system. It is based on the ESP8266 Wi-Fi SoC (System on Chip) and ESP-12 module. Node MCU was created on 13 OCT 2014, Arduino developed their IDE (Integrated Development Environments) software to be compatible with Node MCU. It uses two programming languages: The Lua scripting language and C/C++ Arduino language.

TABLE 3: NodeMCU types

Generation	Version	Features
1st	0.9V1	ESP-12, 4MBflash, 47mm*31mm
2nd	1.0V2	ESP-12E, 4MBflash
3nd	1.0V3	The as V2 but it has two reserve pins for USB power out and the other for an additional GND.

4.5. Onino-OMEGA 2

Omega 2 is a tiny microcontroller, cloud integrated and designed for building IoT projects. It is a Linux computer server with Wi-Fi built-in and designed for connectivity and on-board flash memory. It is a small size and has power efficiency as Arduino and flexible as Raspberry Pi. It has new expandable docks like Expansion Dock, Mini Dock, Power Dock, Relay Expansion Dock, Bluetooth, Cellular, GPS and Arduino Dock R2 (make use of Arduino shield). It is a simple and an affordable to start with it. It offers users an application store to get solutions for IoT. It can be programmed with many languages like C, C++, Node.js, Python and PHP. It has nine sensors: humidity, temperature, gyroscope, light, LED driver, potentiometer, pressure, gas and buzzer [41].

TABLE 4: Onino types

	Omega	Omega2	Omega2 Plus
Memory	64MB	64MB	128MB
CPU	400MHZ	580MHZ	580MHZ
Storage	16MB	16MB	32MB
Price	19$	5$	9$
USB	2.0	2.0	2.0
Wi-Fi	b/g/n	b/g/n	b/g/n
GIPO	18	15	15
PWM	-	2	2
I2C	-	1	1
SPI	-	1	1
I2S	-	1	1

4.6. Intel Galileo

The Intel Galileo is the first microcontroller based on the Intel architecture. It is designed to be compatible with Arduino software and hardware. It uses software Arduino development IDE, which makes easier getting started. The hardware is operating at 3.3v or 5v for Arduino shield. It is the best choice for students and professional developers who seek for simple, efficient and low-cost board due to its features like: a full sized mini-PCI Express slot, Micro-SD slot, 100MB Ethernet port, USB host port, USB TTL UART header and 8Mbyte NOR flash. Galileo supports many programming languages such as Python, Videio4linux, Secure Shell (SSL), Advanced Linux Sound, OpenC, Node.js and Advanced Linux Sound Architecture (ALSA). Galileo uses the default Linux image [42].

5. INTERGRATING CLOUD AND IOT

The new revolution within the field of devices and objects of the real world that these will integrate into the virtual world. In 2011, the number of connected devices was about 12.5 Billion and they were exceeded the number of people on the world [43]. Since the number of devices has increased, there will be Big Data. Storing that Big Data temporally will not be the solution for IoT. There must be another solution with low-cost and light-weight to store that big data as permanently in the cloud [44]. Integrating the cloud and internet of things, it becomes the new wave of computing. We call this new paradigm Cloud of Everything (CoE). The two terms cloud and IoT have grown exponentially. Furthermore, their architectures are different from each other, IoT uses Cloud to improve its processing, computing, energy and power. Many researchers are interested in this integration and proposed a new solution of architecture [15].

6. COE ISSUES

In this section, some problems and difficulties are explained which still appear when integrating between IoT and cloud.

6.1. PROTOCOLS

The protocols represent a major role in complete fulfillment of the sensing requirements. They consider as a data tunnel between IoT devices and the outside world [45]. The sensing devices use IoT protocols like IEEE802.15.4 or IEEE802.11 to transmit data to the person concerned. However, some gateways do not support some of the protocols that the sensor being used. People at the most time choose a cheap and easy sensor to work with. If a sensor is added to the network, it cannot be guaranteed that sensor will be compatible with the gateway. It must consider whether the gateway supports the standardized protocols [15].

Using MAC (Medium Access Control) protocols make the system work efficiently. MAC protocols (collision free) have been proposed for several areas like: DMA, TDMA, CSMA and FDMA. However, these schemes are impossible with billions of available IoT devices. This becomes an issue, which requires further study [46].

6.2. ENERGY EFFICIENCY

In the near future, WSNs will be a part of IoT. When sensor nodes join the network and sense data this will lead to the appearance of power consumption problems. A WSN has four basic elements: a sensing unit, a processing unit, a transmitting/receiving unit, and a power unit. The challenge is to design energy-efficiency sensors instead of the need of battery replacement in a traditional way. For example, design new video sensing, encoding and decoding to consume low-power. In general, the encoding is more complex than decoding because the encoders have to compress the video efficiently by analyzing the redundancy in the video. This process is acceptable because the encoding process is done only once while the decoding process takes multiple times [47].

The inappropriate way is to have a sensor with batteries, especially, if there are billions of sensors all over the world, it`s difficult to change the huge number of batteries. The most suitable way is to have sensors with permanent power supply, as well as taking advantage of the environment that generate electricity like vibrations, light, and airflow. Furthermore, sciences publicized a new chip called Nan-generator that used body movement to generate electricity. Moreover, enable sleep-mode [15].

6.3. IDENTITY MANAGEMENT

There are many "things" communicate over the Internet; these are becoming part of IoT in order to understand, collaborate and to do their job of sensing. There are two types of "things", the first type, dies over a short time, which doesn’t need an identity. The other type requires identification, for example, mobile nodes in smart city with motivation. The identifying object in IoT is one of the main problems in the resource constraint like energy, lifetime, point-to-point delay, memory, routing overhead latency and bandwidth [47].

Identity management (IDM) is defined as the management of things involves identifying devices in the network to make things distinguishable and assigning identifiers. It provides authentication, authorization and polices and controls access to resources. Normally, things have only one identity, but sometimes they have multi identifiers. Various components of the IDM system include; directory services, access management and Password administration [48].

6.4. IPv6

In spite of the IPv6 gets widely used to identify devices, but some existing companies, technologies and customers still use the IPv4. The problem comes with development of IPv6 to be compatible with IPv4 devices [49]. A solution is to enable seamless communication by using a transition technology that is called dual stack routers. Nodes may have dual stack support both IPv4 and IPv6 packets. In addition, use tunneling techniques in cloud virtualization: 6to4, Teredo, and ISATAP.

6.5. QUALITY OF SERVICE

By default, heterogeneous devices support multi traffic types and multi services; a single network can handle with all services and traffic without QoS. Communication networks offer to the end user three types of multimedia: high quality and smooth video, no drop calls and real time voice, and data. As IoT expanded, there are sensors provide different services at the same time; this leads to a bottleneck, hence, QoS becomes an issue. Achieving QoS are guaranteed by maximizing the bandwidth, control delay (the limited time that a packet takes to reach to the destination), jitter (the variation in the end-to-end delay of received packet) and packet loss (the number of packet that failures to reach the destination) to meet customer`s need [50].

6.6. DATA LOCATION

Since cloud and IoT support data mobility, location of data becomes an important issue for several reasons; for user`s perspective, it does not matter to know the location of their data. For example, people share their photos on social networks and they do not concern where their data are stored. However, some systems have a sensitive data that preferred to know the location; and in some cases, they specify their own location. The service provider side should also take care of systems security. In addition, they may take contract with another cloud, so that when the data moved from one place to another, it will not become a problem. The cloud services provider may depend on several factors including the cost of data center (researchers proposed a system reducing the cost [51]), the temperature of servers and dynamic routing may help to optimize resource and reduce costs [52]. Another consideration, the time that takes to access the data should be minimalized; the data should be stored in closest location to the user [15].

7. CONCLUSION

IoT is a new concept and becoming the future of Internet, which a huge number of data generated from things and shared across Internet around the world. A billion of devices, applications and new technologies will connect with each other to improve the quality of life more than anything did before. This paper shows an overview of IoT and about its history and application domains. Some remarks are concluded related to this topic; IoT architecture has been presented to understand the role of each IoT layers to know which the IoT components and technologies are suitable with each layer. IoT protocols functionality existed nowadays have been explained their functionality with open research challenges. Finally, for providing new services arising from this integration to users, Integration cloud with IoT has been emerged. This integration termed as Cloud of Everything, including some issues which have been explained.

In further, we are looking into surveying the next-generation technologies and networks. The 5G networks have been expected to handle trillions of objects by 2020 more. However, it will be a part of IoT; connecting all mobile nodes and fixed devices with each other. New challenges will arise from this integration that should be studied study.

8. REFERENCES

[1] A. H. H. Ngu, M. Gutierrez, V. Metsis, S. Nepal, and M. Z. Sheng, “IoT Middleware: A Survey on Issues and Enabling technologies,” IEEE Internet of Things Journal, pp. 1–1, 2016.

[2] L. Durkop, B. Czybik, and J. Jasperneite, “Performance evaluation of M2M protocols over cellular networks in a lab environment,” 2015 18th International Conference on Intelligence in Next Generation Networks, 2015.

[3] S. N. Srirama, "Web services and cloud enabling mobile Internet of Things" CSI Transactions on Information and Communication Technology, April 2017.

[4] D. Kelaidonis, P. Vlacheas, V. Stavroulaki, S. Georgoulas, K. Moessner, Y. Hashi, K. Hashimoto, Y. Miyake, K. Yamada, and P. Demestichas, “Cloud Internet of Things Framework for Enabling Services in Smart Cities,” Designing, Developing, and Facilitating Smart Cities, pp. 163–191, Jun. 2016.

[5] O. Vermesan and P. Friess, Eds., Internet of things: Converging technologies for smart environments and integrated ecosystems. Aalborg: River Publishers., 2013.

[6] K. Rose, S. Eldridge, and L. Chapin, "The internet of things: An overview", The Internet Society (ISOC), pp. 1–50, Oct. 2015.

[7] J. Höller, V. Tsiatsis, C. Mulligan, S. Karnouskos, S. Avesand, and D. Boyle, From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence. Amsterdam, The Netherlands: Elsevier, 2014.

[8] J. Manyika, M. Chui, P. Bisson, J. Woetzel, R. Dobbs, J. Bughin and D. Aharon. The Internet of Things: mapping the value beyond the hype. June, 2015.

[9] F. Hu, Security and privacy in Internet of things (IoTs): Models, Algorithms, and Implementations. United Kingdom: CRC Press, 2016.

[10] F. DaCosta, Rethinking the Internet of things a scalable approach to connecting everything. New York, NY: ApressOpen, 2013.

[11] M. Yun and B. Yuxin, “Research on the architecture and key technology of Internet of Things (IoT) applied on smart grid,” 2010 International Conference on Advances in Energy Engineering, 2010.

[12] M. A. and S. T., "Internet of Things: Architecture, Security Issues and Countermeasures", International Journal of Computer Applications, vol. 125, no. 14, pp. 1-4, Sep. 2015.

[13] R. Khan, S. U. Khan, R. Zaheer, and S. Khan, “Future Internet: The Internet of Things Architecture, Possible Applications and Key Challenges,” 2012 10th International Conference on Frontiers of Information Technology, 2012.

[14] M. Aazam, I. Khan, A. A. Alsaffar, and E.-N. Huh, “Cloud of Things: Integrating Internet of Things and cloud computing and the issues involved,” Proceedings of 2014 11th International Bhurban Conference on Applied Sciences & Technology (IBCAST) Islamabad, Pakistan, 14th - 18th January, 2014, 2014.

[15] "FAQ - Frequently Asked Questions | MQTT", Mqtt.org. [Online]. Available: http://mqtt.org/faq. [Accessed: 9- March- 2017].

[16] S. Lee, H. Kim, D.-K. Hong, and H. Ju, “Correlation analysis of MQTT loss and delay according to QoS level,” The International Conference on Information Networking 2013 (ICOIN), 2013.

[17] Z. Shelby, K. Hartke, C. Bormann, and B. Frank, “draft-ietf-core-coap-18 - The Constrained Application Protocol (CoAP),” Internet Eng. Task Force (IETF), Fremont, CA, USA, 2013. [Online]. Available: https://tools.ietf.org/html/draft-ietf-core-coap-18. [Accessed: 9- March- 2017].

[18] W. Colitti, K. Steenhaut, and N. D. Caro, "Integrating wireless sensor networks with the web," Extending the Internet to Low power and Lossy Networks (IP+ SN 2011) ,2011.

[19] M. R. Palattella, N. Accettura, X. Vilajosana, T. Watteyne, L. A. Grieco, G. Boggia, and M. Dohler, “Standardized Protocol Stack for the Internet of (Important) Things,” IEEE Communications Surveys & Tutorials, vol. 15, no. 3, pp. 1389–1406, 2013.

[20] M. Koster, J. Jimenez, and rfcmarkup version 1, "Publish-subscribe broker for the constrained application protocol (CoAP)", Tools.ietf.org. [Online]. Available:

https://tools.ietf.org/html/draft-koster-core-coap-pubsub-05. [Accessed: 9- March- 2017].

[21] S. Raza, H. Shafagh, K. Hewage, R. Hummen and T. Voigt, "Lithe: Lightweight Secure CoAP for the Internet of Things", IEEE Sensors J., vol. 13, no. 10, pp. 3711-3720, Oct. 2013.

[22] H. J. Ban, J. Choi, and N. Kang, “Fine-Grained Support of Security Services for Resource Constrained Internet of Things,” International Journal of Distributed Sensor Networks, vol. 2016, pp. 1–8, 2016.

[23] A. Betzler, C. Gomez, I. Demirkol and J. Paradells, "CoAP congestion control for the internet of things", IEEE Communications Magazine, vol. 54, no. 7, pp. 154-160, Jul. 2016.

[24] A. Betzler, C. Gomez, I. Demirkol, and J. Paradells, “CoAP congestion control for the internet of things,” IEEE Communications Magazine, vol. 54, no. 7, pp. 154–160, 2016.

[25] J. Hui, D. Culler, and V. Manral, “An IPv6 Routing Header for Source Routes with the Routing Protocol for Low-Power and Lossy Networks (RPL),” 2012.

[26] "Routing Protocol for LLN (RPL) Configuration Guide, Cisco IOS Release 15M&T", Cisco. [Online]. Available: http://www.cisco.com/c/en/us/td/docs/ios-xml/ios/rpl/configuration/15-mt/rpl-15-mt-book.html. [Accessed9- March- 2017].

[27] P. Thubert, "RPL Objective Function Zero", 2012. [Online]. Available: https://tools.ietf.org/html/draft-ietf-roll-of0-19. [Accessed: 9- March- 2017].

[28] JP. Vasseur, M. Kim, K. Pister, N. Dejean, and D. Barthel, “draft-ietf-roll-routing-metrics-19 - Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks,” Internet Engineering Task Force (IETF) Request for Comments, No. RFC 6551, 2011. [Online]. Available: https://tools.ietf.org/html/draft-ietf-roll-routing-metrics-19. [Accessed: 9- March- 2017].

[29] A. Conta and S. Deering, “Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification,” 2006.

[30] P. Levis, Clausen, T. Heide, J. Hui, O. Gnawali, J. Ko and rfcmarkup version 1, "The trickle algorithm", 2011. [Online]. Available: https://tools.ietf.org/html/rfc6206#section-1. [Accessed: 9- March- 2017].

[31] P. Kinney, IEEE 802.15 WPAN™ Task Group 4 (TG4). [Online]. Available: http://www.ieee802.org/15/pub/TG4.html. [Accessed: 9- March- 2017].

[32] S. Ziegler, P. Kirstein, L. Ladid, Antonio S. and Antonio Jara, "The Case for IPv6 as an Enabler of the Internet of Things - IEEE Internet of Things", Iot.ieee.org, 2015. [Online]. Available: http://iot.ieee.org/newsletter/july-2015/the-case-for-ipv6-as-an-enabler-of-the-internet-of-things.html. [Accessed: 9- March- 2017].

[33] IEEE 802 Working Group, IEEE Standard for Local and Metropolitan Area Networks—Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs), IEEE Std, pp. 1–314, Sep, 2011.

[34] J. Song, S. Han, A. Mok, D. Chen, M. Lucas, M. Nixon, and W. Pratt, “WirelessHART: Applying Wireless Technology in Real-Time Industrial Process Control,” 2008 IEEE Real-Time and Embedded Technology and Applications Symposium, 2008.

[35] Arduino, "Arduino - Introduction", Arduino.cc. [Online]. Available: https://www.arduino.cc/en/Guide/Introduction. [Accessed: 9- March- 2017].

[36] "Introduction to Processing | Raspberry Pi Learning Resources", Raspberrypi.org. [Online]. Available: https://www.raspberrypi.org/learning/introduction-to-processing/. [Accessed: 9- March- 2017].

[37] E. Upton and G. Halfacree, Raspberry pi user guide. Chichester, United Kingdom: Wiley-Blackwell (an imprint of John Wiley & Sons Ltd), 2014.

[38] A. Lofquist, "EmulationStation", Emulationstation.org. [Online]. Available: http://emulationstation.org/gettingstarted.html. [Accessed: 9- March- 2017].

[39] "Telosb Sensors | Telosb Motes |Telosb", Telosb Sensors | Telosb Motes |Telosb, 2014. [Online]. Available: https://telosbsensors.wordpress.com/. [Accessed: 9- March- 2017].

[40] O. Corporation, "Onion – invention platform for IoT". [Online]. Available: https://onion.io/. [Accessed: 9- March- 2017].

[41] I. Corporation, "Intel® Galileo gen 2 development Board—Empower your prototype", Intel. [Online]. Available: http://www.intel.com/content/www/us/en/do-it-yourself/galileo-maker-quark-board.html. [Accessed: 9- March- 2017].

[42] P. N. Howard, "How big is the Internet of Things and how big will it get?" The Brookings Institution, Jun, 2015.

[43] D. Evans, "The Internet of Things – How the Next Evolution of the Internet is Changing Everything," White Paper-Cisco Internet Business Solutions Group, April 2011.

[44] K. Chen, "Challenges and opportunities of internet of things." 17th Asia and South Pacific Design Automation Conference. IEEE, 2012.

[45] P. N. Mahalle and P. N. Railkar, Identity Management for Internet of Things. Vol. 39. River Publishers, 2015.

[46] R. Shaikh and M. Sasikumar, "Identity Management in Cloud Computing," International Journal of Computer Applications, vol. 63, no. 11, pp. 17–19, 2013.

[47] J. Gubbi, R. Buyya, S. Marusic and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions", Future Generation Computer Systems, vol. 29, no. 7, pp. 1645-1660, Sep. 2013.

[48] Y. Cui, P. Wu, M. Xu, J. Wu, Y. Lee, A. Durand and C. Metz, "4over6: network layer virtualization for IPv4-IPv6 coexistence", IEEE Network, vol. 26, no. 5, pp. 44-48, Sep. 2012.

[49] M. Aazam and E.-N. Huh, "Impact of ipv4-ipv6 coexistence in cloud virtualization environment," annals of telecommunications - annales des télécommunications, vol. 69, no. 9-10, pp. 485–496, Aug. 2013.

[50] "Quality of Service (QoS)", Cisco. [Online]. Available: http://www.cisco.com/c/en/us/products/ios-nx-os-software/quality-of-service-qos/index.html. [Accessed: 9- March- 2017].

[51] P. T. Jaeger, J. Lin and J. Grimes, "Cloud Computing and Information Policy: Computing in a Policy Cloud?", Journal of Information Technology & Politics, vol. 5, no. 3, pp. 269-283, Oct. 2008.

[52] Z. Mahmood, "Data Location and Security Issues in Cloud Computing," 2011 International Conference on Emerging Intelligent Data and Web Technologies, 2011.