# Networking Protocols and Standards for Internet of Things

**Tara Salman, Raj Jain**
Department of Computer Science and Engineering
Washington University in St. Louis
{tara.salman, jain}@wustl.edu

## Abstract

This chapter discusses different standards offered by IEEE, IETF and ITU to enable technologies matching the rapid growth in IoT. These standards include communication, routing, network and session layers of the networking stack that are being developed just to meet requirements of IoT. The discussion also includes management and security protocols.

## 1. Introduction

Internet of Things (IoT) and its protocols are among the most highly funded topics in both industry and academia. The rapid evolution of the mobile internet, mini-hardware manufacturing, micro-computing, and machine to machine (M2M) communication has enabled the IoT technologies. According to Gartner, IoT is currently on the top of their hype-cycle, which implies that a large amount of money is being invested on it by the industry. Billions of dollars are being spent on IoT enabling technologies and research while much more is expected to come in the upcoming years [Gartner14].

IoT technologies allow things, or devices that are not computers, to act smartly and make collaborative decisions that are beneficial to certain applications. They allow things to hear, see, think or act by allowing them to communicate and coordinate with others in order to make decisions that can be as critical as saving lives or buildings. They transform "things" from being passively computing and making individual decisions to actively and ubiquitously communicating and collaborating to make a single critical decision. The underlying technologies of ubiquitous computing, embedded sensors, light communication and internet protocols allow IoT to provide its significant, however, they impose lots of challenges and introduce the need for specialized standards and communication protocols.

In this chapter, we highlight IoT protocols that are operating at different layers of the networking stack, including: Medium Access Control (MAC) layer, network layer and session layer. We present standards protocols offered by Internet Engineering Task Force (IETF), Institute of Electrical and Electronics Engineers (IEEE), International Telecommunication Union (ITU) and other standard organizations. These standards were proposed over the past half-decade to meet IoT current and future needs.

The rest of the chapter is organized as follows: Section 2 describes the first layer of networking protocols, which is the data link layer and MAC protocols. Following that, Section 3 handles the network layer routing protocols while Section 4 presents network layer encapsulation protocols and Section 5 handles the session layer protocols. Section 6 briefly summarizes the management and Section 7 describes security mechanisms in key protocols. Section 8 gives some discussion points about IoT challenges. Finally, Section 9 summarizes our discussion and highlights the main points presented.

## 1.1. Related Works

There are several survey papers that handle different aspects of standardization in IoT. Examples of such surveys include a survey of IETF standards in [Sheng13], security protocols in [Granjal15], and application, or transport, layer standards in [Karagiannis15]. Other papers discuss a specific layer of standardizations such as communication protocols or routing. Most importantly, we recommend [Fuaha15] to readers interested in more details. That paper summarizes some of the most important standards that are offered by different standards organizations. It also provides a discussion of different IoT challenges including mobility, scalability. In this chapter, we aim to provide a comprehensive survey of newly arising standards including some other drafts and protocols that were not discussed in [Fuaha15]. This allows us to discuss more standards, add some of the recent standard drafts offered in IETF, and discuss state of the art protocol that are expected to go for standardization in the future.

## 1.2. IoT Ecosystem

Figure 1 shows a 7-layer model of IoT ecosystem. At the bottom layer is the market or application domain, which may be smart grid, connected home, or smart health, etc. The second layer consists of sensors that enable the application. Examples of such sensors are temperature sensors, humidity sensors, electric utility meters, or cameras. The third layer consists of interconnection layer that allows the data generated by sensors to be communicated, usually to a computing facility, data center, or a cloud. There the data is aggregated with other known data sets such as geographical data, population data, or economic data. The combined data is then analyzed using machine learning and data mining techniques. To enable such large distributed applications, we also need the latest application level collaboration and communication software, such as, software defined networking (SDN), services oriented architecture (SOA), etc. Finally, the top layer consists of services that enable the market and may include energy management, health management, education, transportation etc. In addition to these 7 layers that are built on the top of each other, there are security and management applications that are required for each of the layers and are, therefore, shown on the side.

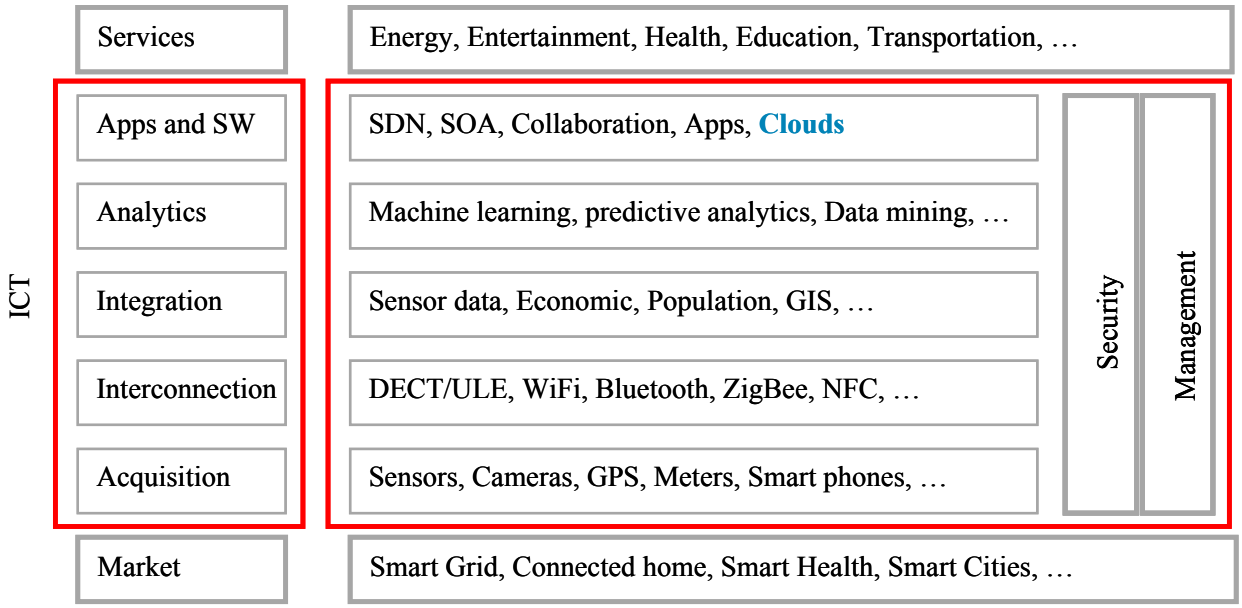


**Figure 1:** IoT ecosystem

In this chapter, we concentrate on the interconnection layer. This layer itself can be shown in a multi-layer stack as shown in Figure 2. We have shown only the datalink, network, and transport/session layers. The datalink layer connects two IoT elements which generally could be two sensors or the sensor and the gateway device that connects a set of sensors to the Internet. Often there is a need for multiple sensors to communicate and aggregate information before getting to the Internet. Specialized protocols have been designed for routing among sensors and are part of the routing layer. The session layer protocols enable messaging among various elements of the IoT communication subsystem. A number of security and management protocols have also been developed for IoT as shown in the figure.

| | | | | |
|---|---|---|---|---|
| **Session** | | **MQTT**, **SMQTT**, **CoRE**, **DDS**, **AMQP** , **XMPP**, **CoAP**, … | **Security** | **Management** |
| **Network** | **Encapsulation** | **6LowPAN**, **6TiSCH**, **6Lo**, Thread, … | TCG, Oath 2.0, SMACK, SASL, ISASecure, ace, DTLS, Dice, … | **IEEE 1905**, **IEEE 1451**, … |
| | **Routing** | **RPL**, **CORPL**, **CARP**, … | | |
| **Datalink** | | WiFi, **Bluetooth Low Energy**, **Z-Wave**, **ZigBee Smart**, **DECT/ULE**, 3G/LTE, NFC, **Weightless**, **HomePlug GP**, **802.11ah**, **802.15.4e**, **G.9959**, **WirelessHART**, **DASH7**, ANT+, **LTE-A**, **LoRaWAN**, … | | |

**Figure 2:** Protocols for IoT

Many different standards and protocols for IoT have been proposed by various standards organizations. Prominent among them are Institution of Electrical and Electronics Engineers (IEEE), Internet Engineering Task Force (IETF), and International Telecommunications Union (ITU). These protocols are listed in Figure 2. Although, we have tried to make the list as current as possible, new protocols are continuously being proposed and may appear in future. In this chapter, we concentrate on protocols shown in bold face in Figure 2. We consider these as most commonly recommended and/or designed especially for IoT.

# 2. IoT Data Link Protocols

In this section, we discuss the datalink layer protocol standards. The discussion includes physical (PHY) and MAC layer protocols which are combined by most standards.

## 2.1. IEEE 802.15.4e

IEEE 802.15.4 is the most commonly used IoT standard for MAC. It defines a frame format, headers including source and destination addresses, and how nodes can communicate with each other. The frame formats used in traditional networks are not suitable for low power multi-hop networking in IoT due to their overhead. In 2008, IEEE802.15.4e was created to extend IEEE802.15.4 and support low power communication. It uses time synchronization and channel hopping to enable high reliability, low cost and meet IoT communications requirements. Its specific MAC features can be summarized as follows [802.15.4]:

- **Slotframe Structure:** IEEE 802.15.4e frame structure is designed for scheduling and telling each node what to do. A node can sleep, send, or receive information. In the sleep mode, the node turns off its radio to save power and stores all messages that it needs to send at the next transmission opportunity. When transmitting, it sends its data and waits for an acknowledgment. When receiving, the node turns on its radio before the scheduled receiving time, receives the data, sends an acknowledgement, turn off its radio, delivers the data to the upper layers and goes back to sleep.
- **Scheduling:** The standard does not define how the scheduling is done but it needs to be built carefully such that it handles mobility scenarios. It can be centralized by a manager node which is responsible for building the schedule, informing others about the schedule and other nodes will just follow the schedule.
- **Synchronization:** Synchronization is necessary to maintain nodes' connectivity to their neighbors and to the gateways. Two approaches can be used: acknowledgment-based or frame-based synchronization. In acknowledgement-based mode, the nodes are already in communication and they send acknowledgment for reliability guarantees, thus can be used to maintain connectivity as well. In frame-based mode, nodes are not communicating and hence, they send an empty frame at pre-specified intervals (about 30 second typically).
- **Channel Hopping:** IEEE802.15.4e introduces channel hopping for time slotted access to the wireless medium. Channel hopping requires changing the frequency channel using a pre-determined random sequence. This introduces frequency diversity and reduces the effect of interference and multi-path fading. Sixteen channels are available which adds to

network capacity as two frames over the same link can be transmitted on different frequency channels at the same time.

- **Network Formation:** Network formation includes advertisement and joining components. A new device should listen for advertisement commands and upon receiving at least one such command, it can send a join request to the advertising device. In a centralized system, the join request is routed to the manger node and processed there while in distributed systems, they are processed locally. Once a device joins the network and it is fully functional, the formation is disabled and will be activated again if it receives another join request.

## 2.2. IEEE 802.11ah

IEEE 802.11ah is a light (low-energy) version of the original IEEE 802.11 wireless medium access standard. It has been designed with less overhead to meet IoT requirements. IEEE 802.11 standards (also known as Wi-Fi) are the most commonly used wireless standards. They have been widely used and adopted for all digital devices including laptops, mobiles, tablets, and digital TVs. However, the original WiFi standards are not suitable for IoT applications due to their frame overhead and power consumption. Hence, IEEE 802.11 working group initiated 802.11ah task group to develop a standard that supports low overhead, power friendly communication suitable for sensors and motes [Park15]. The basic 802.11ah MAC layer features include:

- **Synchronization Frame:** A station is not allowed to transmit unless it has valid medium information that allows it to capture the medium and stop packet exchange by others. It can know such information if it receives the duration field packet correctly. If it does not receive it correctly, then it should wait for a duration called *Probe Delay*. Probe Delay can be configured by the access points in 802.11ah and announced by transmitting a synchronization frame at the beginning of the time slot.
- **Efficient Bidirectional Packet Exchange:** This feature allows the sensor device to save more power by allowing both uplink and downlink communication between the access point and the sensor and allowing it to go to sleep as soon as it finishes the communication.
- **Short MAC Frame:** The normal IEEE 802.11 frame is about 30 bytes, which is too large for IoT applications. IEEE 802.11ah mitigates this problem by defining a short MAC frame with about 12 bytes.
- **Null Data Packet:** In IEEE 802.11 the control frames, such as Acknowledgment (ACK) frames, are about 14 bytes and have no data, which adds a lot of overhead. IEEE 802.11ah mitigates this problem by replacing the ACK frame with a preamble, a tiny signal.
- **Increased Sleep Time:** 802.11ah is designed for low-power sensors and, hence, it allows a long sleep period of time and waking up infrequently to exchange data only.

## 2.3. WirelessHART

WirelessHART is a datalink protocol that operates on the top of IEEE 802.15.4 PHY and adopts Time Division Multiple Access (TDMA) in its MAC. It is a secure and reliable MAC protocol

that uses advanced encryption to encrypt the messages and calculate the integrity in order to offer reliability. The architecture, as shown in Figure 3 consists of a network manager, a security manager, a gateway to connect the wireless network to the wired networks, wireless devices as field devices, access points, routers and adapters. The standard offers end-to-end, per-hop or peer-to-peer security mechanisms. End to end security mechanisms enforce security from sources to destinations while per-hop mechanisms secure it to next hop only [Kim08, Raza10].

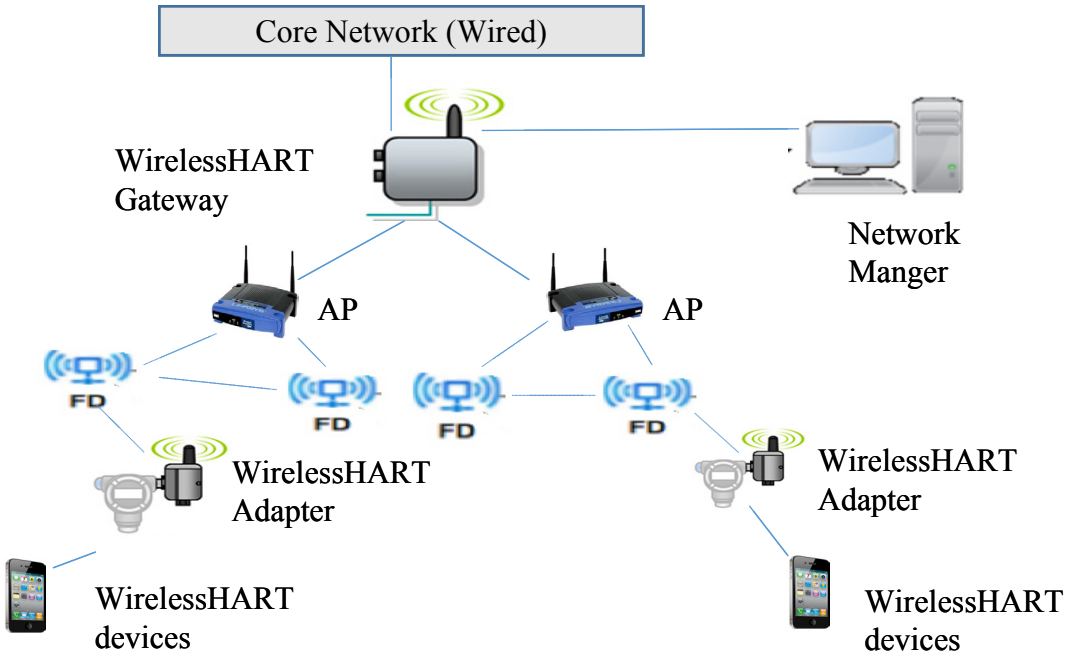


**Figure 3:** WirelessHART Architecture

## 2.4. Z-Wave

Z-Wave is a low-power MAC protocol designed for home automation and has been used for IoT communication, especially for smart home and small commercial domains. It covers about 30-meter point-to-point communication and is suitable for small messages in IoT applications, like light control, energy control, wearable healthcare control and others. It uses CSMA/CA for collision detection and ACK messages for reliable transmission. It follows a master/slave architecture in which the master control the slaves, send them commands, and handling scheduling of the whole network [Z-Wave].

## 2.5. Bluetooth Low Energy

Bluetooth low energy or Bluetooth smart is a short range communication protocol with PHY and MAC layer widely used for in-vehicle networking. Its low energy can reach ten times less than the classic Bluetooth while its latency can reach 15 times. Its access control uses a contention-less MAC with low latency and fast transmission. It follows master/slave architecture and offers two types of frames: adverting and data frames. The Advertising frame is used for discovery and is sent by slaves on one or more of dedicated advertisement channels. Master nodes sense advertisement channels to find slaves and connect them. After connection, the master tells the

slave it's waking cycle and scheduling sequence. Nodes are usually awake only when they are communicating and they go to sleep otherwise to save their power [Decuir10, Gomez12].

## 2.6. ZigBee Smart Energy

ZigBee smart energy is designed for a large range of IoT applications including smart homes, remote controls and healthcare systems. It supports a wide range of network topologies including star, peer-to-peer, or cluster-tree. A coordinator controls the network and is the central node in a star topology, the root in a tree or cluster topology and may be located anywhere in peer-to-peer. ZigBee standard defines two stack profiles: ZigBee and ZigBee Pro. These stack profiles support full mesh networking and work with different applications allowing implementations with low memory and processing power. ZigBee Pro offers more features including security using symmetric-key exchange, scalability using stochastic address assignment, and better performance using efficient many-to-one routing mechanisms [ZigBee08].

## 2.7. DASH7

DASH7 is a wireless communication protocol for active RFID that operates in globally available Industrial Scientific Medical (ISM) band and is suitable for IoT requirements. It is mainly designed for scalable, long range outdoor coverage with higher data rate compared to traditional ZigBee. It is a low-cost solution that supports encryption and IPv6 addressing. It supports a master/slave architecture and is designed for burst, lightweight, asynchronous and transitive traffic. Its MAC layer features can be summarized as follows [Cetinkaya15]:

- **Filtering:** Incoming frames are filtered using three processes; cyclic redundancy check (CRC) validation, a 4-bit subnet mask, and link quality assessment. Only the frames that pass all three checks are processed further.
- **Addressing:** DASH7 uses two types of addresses: the unique identifier which is the EUI-64 ID and dynamic network identifier which is 16-bit address specified by the network administrator.
- **Frame format:** The MAC frame has a variable length of maximum 255 bytes including addressing, subnets, estimated power of the transmission and some other optional fields.

## 2.8. HomePlug

HomePlug GreenPHY (HomePlugGP) is another MAC protocol developed by HomePlug Powerline Alliance that is used in home automation applications. HomePlug suite covers both PHY and MAC layers and has three versions: HomePlug-AV, HomePlug-AV2, and HomePlugGP. HomePlug-AV is the basic power line communication protocol which uses TDMA and CSMA/CA as MAC layer protocol, supports adaptive bit loading which allows it to change its rate depending on the noise level and uses Orthogonal Frequency Division Multiplexing (OFDM) and four modulation techniques.

HomePlugGP is designed for IoT generally and specifically for home automation and smart grid applications. It is basically designed to reduce the cost and power consumption of HomePlug-AV while keeping its interoperability, reliability and coverage. Hence, it uses OFDM, as in

HomePlug, but with one modulation only. In addition, HomePlugGP uses Robust OFDM coding to support low rate and high reliability transmission. HomePlug-AV uses only CSMA as a MAC layer technique while HomePlugGP uses both CSMA and TDMA. Moreover, HomePlugGP has a power-save mode that allows nodes to sleep much more than Home Plug by synchronizing their sleep time and waking up only when necessary [HomePlug].

## 2.9. G.9959

G.9959 is a MAC layer protocol from ITU, designed for low bandwidth and cost, half-duplex reliable wireless communication. It is designed for real-time applications where time is really critical, reliability is important, and low power consumption is required. The MAC layer characteristics include: unique network identifiers that allow 232 nodes to join one network, collision avoidance mechanisms, backoff time in case of collision, automatic retransmission to guarantee reliability, dedicated wakeup pattern that allows nodes to sleep when they are out of communication and hence saves their power. G9959 MAC layer features include unique channel access, frame validation, acknowledgments, and retransmission [RFC7428, G9959].

## 2.10. LTE-A

Long-Term Evolution Advanced (LTE-A) is a set of standards designed to fit M2M communication and IoT applications in cellular networks. LTE-A is a scalable, lower-cost protocol compared to other cellular protocols. LTE-A uses OFDMA (Orthogonal Frequency Division Multiple Access) as a MAC layer access technology, which divides the frequency into multiple bands and each one can be used separately. The architecture of LTE-A consists of a core network (CN), a radio access network (RAN), and the mobile nodes. The CN is responsible for controlling mobile devices and to keep track of their IPs. RAN is responsible for establishing the control and data planes and handling the wireless connectivity and radio-access control. RAN and CN communicate using S1 link, as shown in Figure 4 where RAN consists of the eNB's to which other mobile nodes are connected wirelessly [Hasan13].

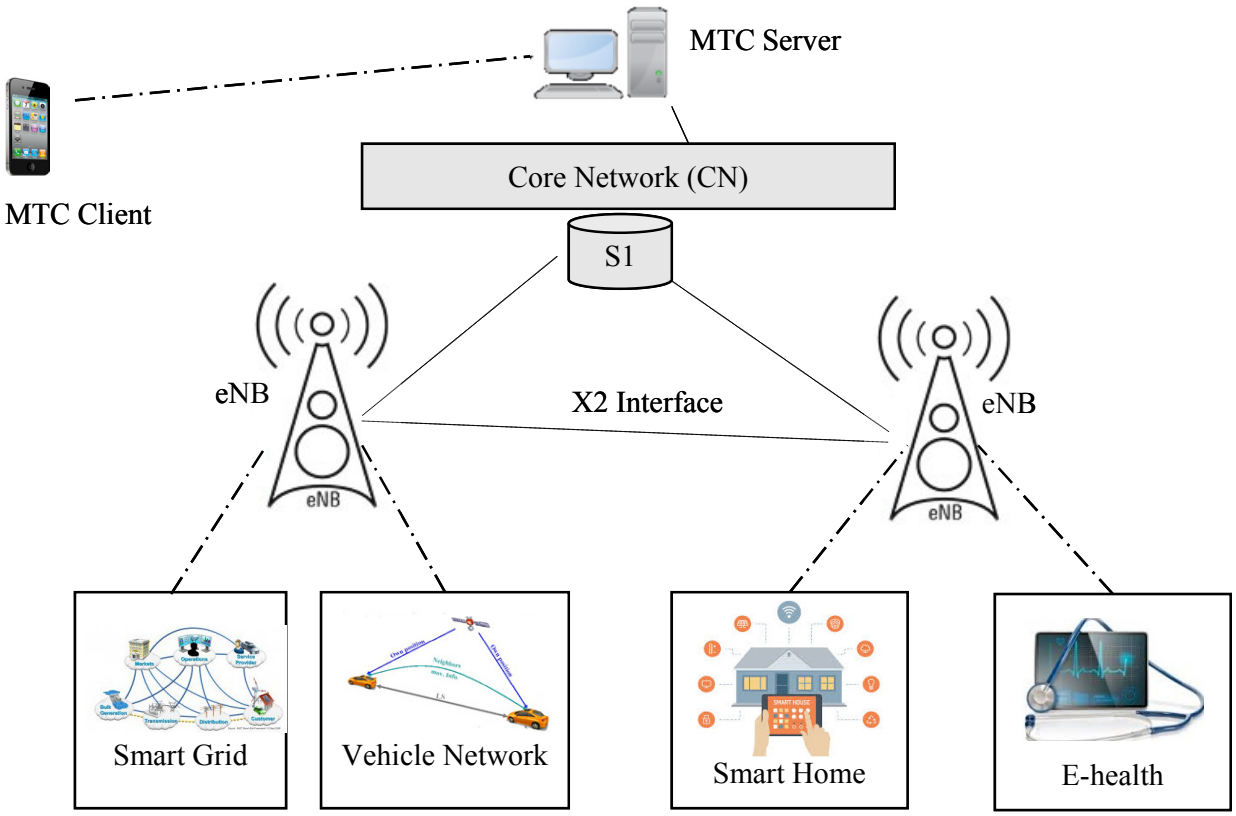


**Figure 4:** LTE-A Architecture

## 2.11. LoRaWAN

LoRaWAN is a newly arising wireless technology designed for low-power WAN networks with low cost, mobility, security, and bi- directional communication for IoT applications. It is a low-power consumption optimized protocol designed for scalable wireless networks with millions of devices. It supports redundant operation, location free, low cost, low power and energy harvesting technologies to support the future needs of IoT while enabling mobility and ease of use features [Lorawan15].

## 2.12. Weightless

Weightless is another wireless WAN technology for IoT applications designed by the Weightless Special Interest Group (SIG) - a non-profit global organization. It has two sets of standards: Weightless-N and Weightless-W. Weightless-N was first developed to support low cost, low power M2M communication using time division multiple access with frequency hopping to minimize the interference. It uses ultra-narrow bands in the sub-1GHz ISM frequency band. On the other hand, Weightless-W provides the same features but uses television band frequencies [Poole14].

### 2.13. DECT/ULE

DECT (Digital Enhanced Cordless Telecommunications) is a universal European standard for cordless phones. In their latest extension DECT/ULE (Ultra Low Energy), they have specified a low-power and low-cost air interface technology that can be used for IoT applications. Due to its dedicated channel assignment, DECT does not suffer from congestion and interference. DECT/ULE supports FDMA, TDMA and time division multiplexing which were not supported in the original DECT protocol [Sush2015].

### 2.14. Summary

In this section, different datalink protocols were discussed in brief to present their main differences and usage in IoT. Generally, the most widely used standards in IoT are Bluetooth and ZigBee. IEEE 802.11ah, on the other hand, is the easiest to be used due to the existing and widely separated infrastructure of IEEE 802.11 which is the most used infrastructure in other wireless applications. However, some providers would seek for more reliable and secured technology and hence would use HomePlug for LAN connectivity. Newly arising LoRaWAN seems to be promising for such applications as well.

# 3. Network Layer Routing Protocols

In this section, we discuss some standard and non-standard protocols that are used for routing in IoT applications. It should be noted that we have partitioned the network layer in two sublayers: routing layer which handles the transfer the packets from source to destination, and an encapsulation layer that forms the packets. Encapsulation mechanisms will be discussed in the next section.

### 3.1. RPL

Routing Protocol for Low-Power and Lossy Networks (RPL) is distance-vector protocol that can support a variety of datalink protocols, including the ones discussed in the previous section. It builds a Destination Oriented Directed Acyclic Graph (DODAG) that has only one route from each leaf node to the root in which all the traffic from the node will be routed to. At first, each node sends a DODAG Information Object (DIO) advertising itself as the root. This message is propagated in the network and the whole DODAG is gradually built. When communicating, the node sends a Destination Advertisement Object (DAO) to its parents, the DAO is propagated to the root and the root decides where to send it depending on the destination. When a new node wants to join the network, it sends a DODAG Information Solicitation (DIS) request to join the network and the root will reply back with a DAO Acknowledgment (DAO-ACK) confirming the join. RPL nodes can be stateless, which is most common, or stateful. A stateless node keeps tracks of its parents only. Only root has the complete knowledge of the entire DODAG. Hence, all communications go through the root in every case. A stateful node keeps track of its children and parents and hence when communicating inside a sub-tree of the DODAG, it does not have to go through the root [RFC6550].

### 3.2. CORPL

An extension of RPL is CORPL, or cognitive RPL, which is designed for cognitive networks and uses DODAG topology generation but with two new modifications to RPL. CORPL utilizes opportunistic forwarding to forward the packet by choosing multiple forwarders (forwarder set) and coordinates between the nodes to choose the best next hop to forward the packet to. DODAG is built in the same way as RPL. Each node maintains a forwarding set instead of its parent only and updates its neighbor with its changes using DIO messages. Based on the updated information, each node dynamically updates its neighbor priorities in order to construct the forwarder set [Aijaz15].

### 3.3. CARP

Channel-Aware Routing Protocol (CARP) is a distributed routing protocol designed for underwater communication. It can be used for IoT due to its lightweight packets. It considers link quality, which is computed based on historical successful data transmission gathered from neighboring sensors, to select the forwarding nodes. There are two scenarios: network initialization and data forwarding. In network initialization, a HELLO packet is broadcasted from the sink to all other nodes in the networks. In data forwarding, the packet is routed from sensor to sink in a hop-by-hop fashion. Each next hop is determined independently. The main problem with CARP is that it does not support reusability of previously collected data. In other words, if the application requires sensor data only when it changes significantly, then CARP data forwarding is not beneficial to that specific application. An enhancement of CARP was done in E-CARP by allowing the sink node to save previously received sensory data. When new data is needed, E-CARP sends a *Ping* packet which is replied with the data from the sensors nodes. Thus, E-CARP reduces the communication overhead drastically [Shou15].

### 3.4. Summary

Three routing protocols in IoT were discussed in this section. RPL is the most commonly used one. It is a distance vector protocol designed by IETF in 2012. CORPL is a non-standard extension of RPL that is designed for cognitive networks and utilizes the opportunistic forwarding to forward packets at each hop. On the other hand, CARP is the only distributed hop based routing protocol that is designed for IoT sensor network applications. CARP is used for underwater communication mostly. Since it is not standardized and just proposed in literature, it is not yet used in other IoT applications.

# 4. Network Layer Encapsulation Protocols

One problem in IoT applications is that IPv6 addresses are too long and cannot fit in most IoT datalink frames which are relatively much smaller. Hence, IETF is developing a set of standards to encapsulate IPv6 datagrams in different datalink layer frames for use in IoT applications. In this section, we review these mechanisms briefly.

## 4.1. 6LoWPAN

IPv6 over Low power Wireless Personal Area Network (6LoWPAN) is the first and most commonly used standard in this category. It efficiently encapsulates IPv6 long headers in IEEE802.15.4 small packets, which cannot exceed 128 bytes. The specification supports different length addresses, low bandwidth, different topologies including star or mesh, power consumption, low cost, scalable networks, mobility, unreliability and long sleep time. The standard provides header compression to reduce transmission overhead, fragmentation to meet the 128-byte maximum frame length in IEEE802.15.4, and support of multi-hop delivery. Frames in 6LoWPAN use four types of headers: No 6loWPAN header (00), Dispatch header (01), Mesh header (10) and Fragmentation header (11). In No 6loWPAN header case, any frame that does not follow 6loWPAN specifications is discarded. Dispatch header is used for multicasting and IPv6 header compressions. Mesh headers are used for broadcasting; while Fragmentation headers are used to break long IPv6 header to fit into fragments of maximum 128-byte length.

## 4.2. 6TiSCH

6TiSCH working group in IETF is developing standards to allow IPv6 to pass through Time-Slotted Channel Hopping (TSCH) mode of IEEE 802.15.4e datalinks. It defines a Channel Distribution usage matrix consisting of available frequencies in columns and time-slots available for network scheduling operations in rows. This matrix is portioned into chucks where each chunk contains time and frequencies and is globally known to all nodes in the network. The nodes within the same interference domain negotiate their scheduling so that each node gets to transmit in a chunk within its interference domain. Scheduling becomes an optimization problem where time slots are assigned to a group of neighboring nodes sharing the same application. The standard does not specify how the scheduling can be done and leaves that to be an application specific problem in order to allow for maximum flexibility for different IoT applications. The scheduling can be centralized or distributed depending on application or the topology used in the MAC layer [Dujovne14].

## 4.3. 6Lo

IPv6 over Networks of Resource-constrained Nodes (6Lo) working group in IETF is developing a set of standards on transmission of IPv6 frames on various datalinks. Although, 6LowPAN and 6TiSCH, which cover IEEE 802.15.4 and IEEE 802.15.4e, were developed by different working groups, it became clear that there are many more datalinks to be covered and so 6Lo working group was formed. At the time of this writing most of the 6Lo specifications have not been finalized and are in various stages of drafts. For example, IPV6 over Bluetooth Low Energy Mesh Networks, IPv6 over IEEE 485 Master-Slave/Token Passing (MS/TP) networks, IPV6 over DECT/ULE, IPV6 over NFC, IPv6 over IEEE 802.11ah, and IPv6 over Wireless Networks for Industrial Automation Process Automation (WIA-PA) drafts are being developed to specify how to transmit IPv6 datagrams over their respective datalinks [6Lo]. Two of these 6Lo specifications “IPv6 over G.9959” and “IPv6 over Bluetooth Low Energy” have been approved as RFC and are described next.

### 4.4. IPv6 over G.9959

RFC 7428 defines the frame format for transmitting IPv6 packet on ITU-T G.9959 networks. G.9959 defines a unique 32-bit home network identifier that is assigned by the controller and 8-bit host identifier that is allocated for each node. An IPv6 link local address must be constructed by the link layer derived 8-bit host identifier so that it can be compressed in G.9959 frame. Furthermore, the same header compression as in 6lowPAN is used here to fit an IPv6 packet into G.9959 frames. RFC 7428 also provides a level of security by a shared network key that is used for encryption. However, applications with a higher level of security requirements need to handle their end-to-end encryption and authentication using their own higher layer security mechanisms [6Lo].

### 4.5 IPv6 over Bluetooth Low Energy

Bluetooth Low Energy is also known as Bluetooth Smart and was introduced in Bluetooth V4.0 and enhanced in V4.1. RFC 7668 [RFC7668], which specifies IPv6 over Bluetooth LE, reuses most of the 6LowPAN compression techniques. However, since the Logical Link Control and Adaptation Protocol (L2CAP) sublayer in Bluetooth already provides segmentation and reassembly of larger payloads in to 27 byte L2CAP packets, fragmentation features from 6LowPAN standards are not used. Another significant difference is that Bluetooth Low Energy does not currently support formation of multi-hop networks at the link layer. Instead, a central node acts as a router between lower-powered peripheral nodes.

### 4.6. Summary

In this section, encapsulation protocols for IPv6 in IoT MAC frame were discussed. First, two standards for IPv6 over 802.15.4 and 802.15.4e were discussed. Such protocols are important as 802.15.4e is the most widely use encapsulation framework designed for IoT. Following that, 6Lo specifications are briefly and broadly discussed just to present their existence in IETF standards. These drafts handle passing IPv6 over different channel access mechanism using 6LoWPAN standards. Then, two of 6Lo Specifications which became IETF RFCs are discussed in more details. The importance of presenting these standards is to highlight the challenge of interoperability between different MAC standards which is still challenging due to the diversity of protocols.

# 5. Session Layer Protocols

This section reviews standards and protocols for message passing in IoT session layer proposed by different standardization organizations. Most of the IP applications, including IoT applications use TCP or UDP for transport. However, there are several message distribution functions that are common among many IoT applications; it is desirable that these functions be implemented in an interoperable standard ways by different applications. These are the so called “Session Layer” protocols described in this section.

## 5.1. MQTT

Message Queue Telemetry Transport (MQTT) was introduced by IBM in 1999 and standardized by OASIS in 2013 [Locke10, Karagiannis15]. It is designed to provide embedded connectivity between applications and middleware's on one side and networks and communications on the other side. It follows a publish/subscribe architecture, as shown in Figure 5, where the system consists of three main components: publishers, subscribers, and a broker. From IoT point of view, publishers are basically the lightweight sensors that connect to the broker to send their data and go back to sleep whenever possible. Subscribers are applications that are interested in a certain topic, or sensory data, so they connect to brokers to be informed whenever new data are received. The brokers classify sensory data in topics and send them to subscribers interested in the topics.

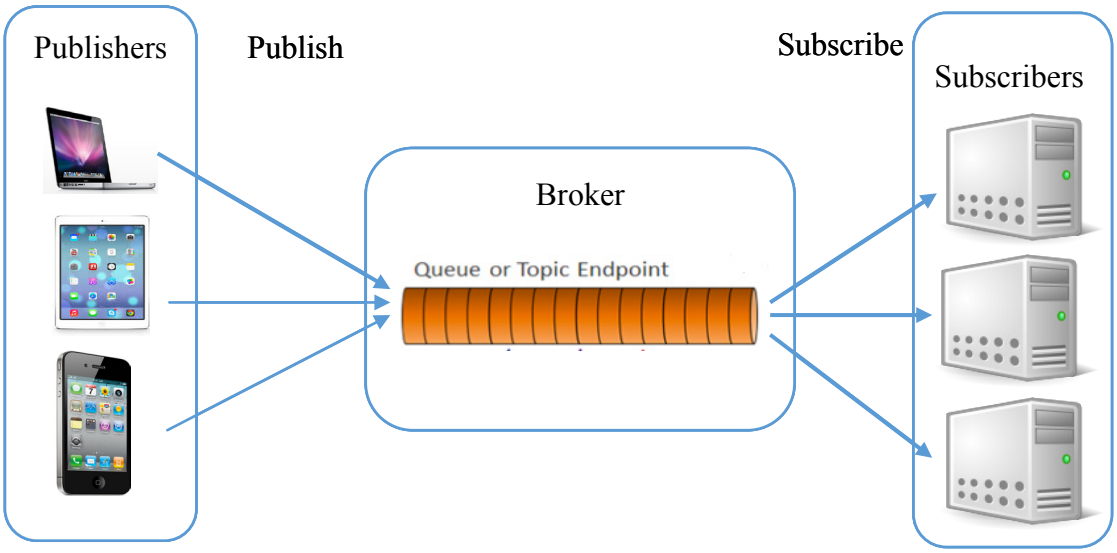


**Figure 5:** MQTT Architecture

## 5.2. SMQTT

An extension of MQTT is Secure MQTT (SMQTT) which uses encryption based on lightweight attribute based encryption. The main advantage of using such encryption is the broadcast encryption feature, in which one message is encrypted and delivered to multiple other nodes, which is quite common in IoT applications. In general, the algorithm consists of four main stages: setup, encryption, publish and decryption. In the setup phase, the subscribers and publishers register themselves to the broker and get a master secret key according to their developer's choice of key generation algorithm. Then, when the data is published, it is encrypted, published by the broker which sends it to the subscribers and finally decrypted at the subscribers which have the same master secret key. The key generation and encryption algorithms are not standardized. SMQTT is proposed only to enhance MQTT security feature [Singh15].

## 5.3. AMQP

The Advanced Message Queuing Protocol (AMQP) is another session layer protocol that was designed for financial industry. It runs over TCP and provides a publish/subscribe architecture which is similar to that of MQTT. The difference is that the broker is divided into two main components: exchange and queues, as shown in Figure 6. The exchange is responsible for receiving publisher messages and distributing them to queues based on pre-defined roles and conditions. Queues basically represent the topics and subscribed by subscribers which will get the sensory data whenever they are available in the queue [AMQP12].

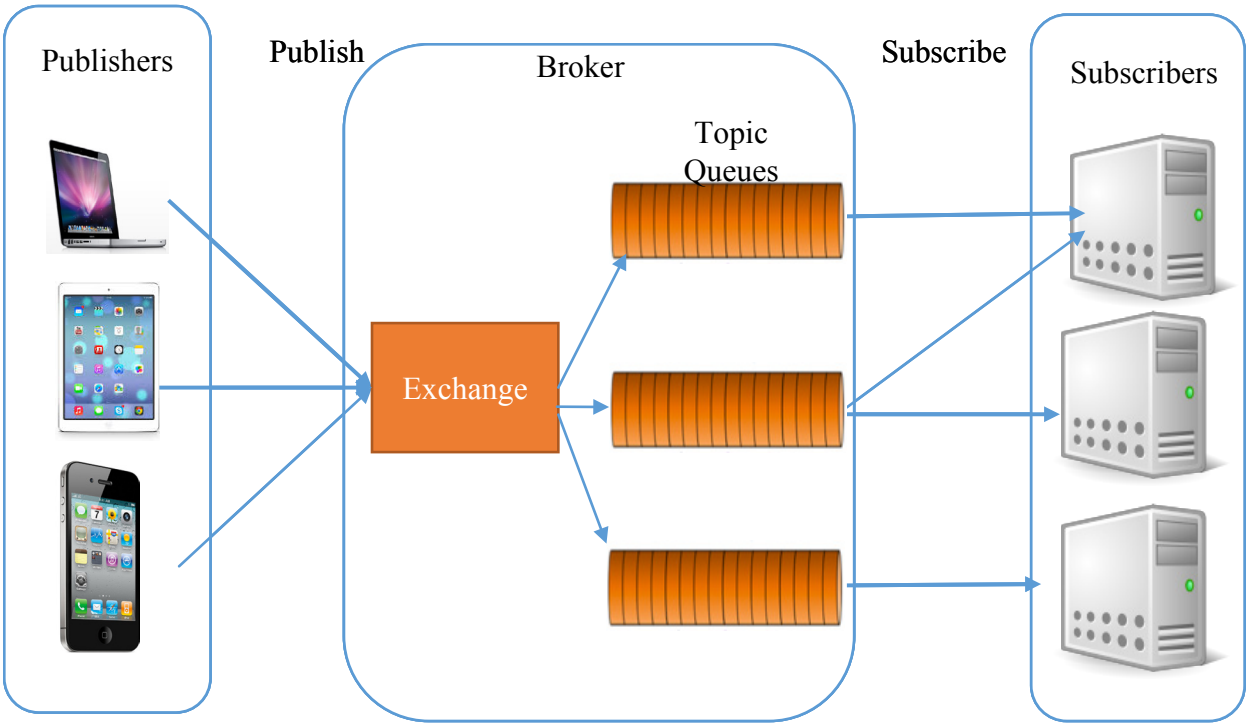


**Figure 6:** AMQP Architecture

## 5.4. CoAP

The Constrained Application Protocol (CoAP) is another session layer protocol designed by IETF Constrained RESTful Environment (Core) working group to provide lightweight RESTful (HTTP) interface. Representational State Transfer (REST) is the standard interface between HTTP client and servers. However, for lightweight applications such as IoT, REST could result in significant overhead and power consumption. CoAP is designed to enable low-power sensors to use RESTful services while meeting their power constrains. It is built over UDP, instead of TCP commonly used in HTTP and has a light mechanism to provide reliability. CoAP architecture is divided into two main sublayers: messaging and request/response. The messaging sublayer is responsible for reliability and duplication of messages while the request/response sublayer is responsible for communication. As shown in Figure 7, CoAP has four messaging modes: confirmable, non-confirmable, piggyback and separate. Confirmable and non-confirmable modes represent the reliable and unreliable transmissions, respectively while the

other modes are used for request/response. Piggyback is used for client/server direct communication where the server sends its response directly after receiving the message, i.e., within the acknowledgment message. On the other hand, the separate mode is used when the server response comes in a message separate from the acknowledgment, and may take some time to be sent by the server. As in HTTP, CoAP utilizes GET, PUT, PUSH, DELETE messages requests to retrieve, create, update, and delete, respectively [RFC7252, Karagiannis15].

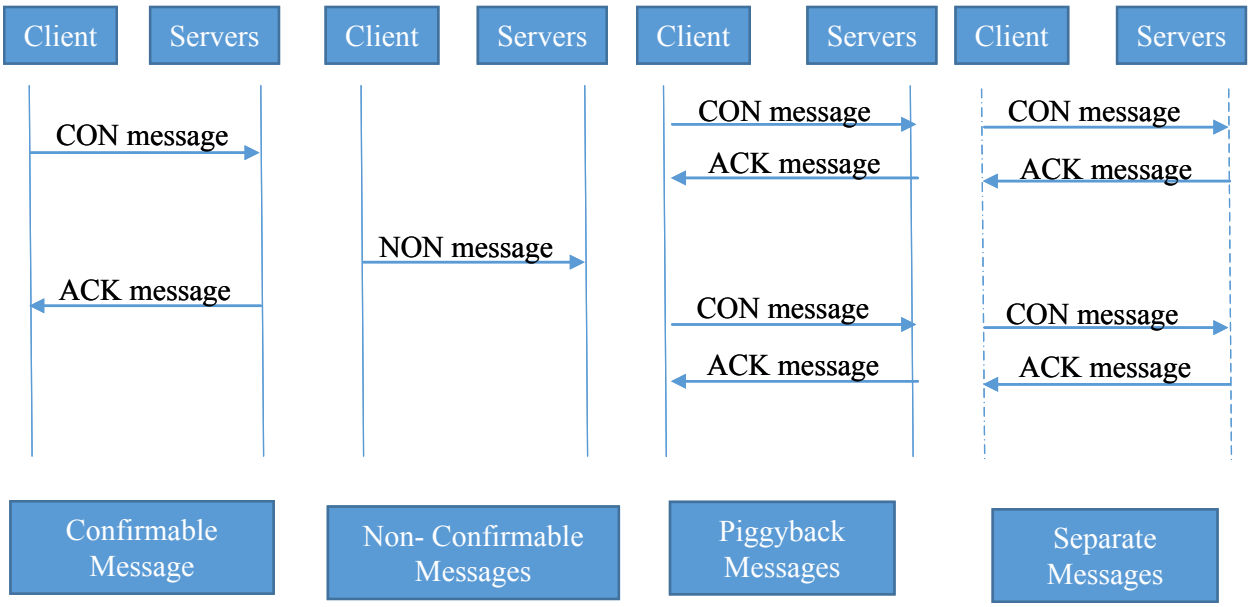


**Figure 7: CoAP Messages**

## 5.5. XMPP

Extensible Messaging and Presence Protocol (XMPP) is a messaging protocol that was designed originally for chatting and message exchange applications. It was standardized by IETF more than a decade ago. Hence, it is well known and has proven to be highly efficient over the internet. Recently, it has been reused for IoT applications as well as a protocol for SDN. This reusing of the same standard is due to its use of XML which makes it easily extensible. XMPP supports both publish/subscribe and request/ response architecture and it is up to the application developer to choose which architecture to use. It is designed for near real-time applications and, thus, efficiently supports low-latency small messages. It does not provide any quality of service guarantees and, hence, is not practical for M2M communications. Moreover, XML messages create additional overhead due to lots of headers and tag formats which increase the power consumption that is critical for IoT application. Hence, XMPP is rarely used in IoT but has gained some interest for enhancing its architecture in order to support IoT applications [RFC6120, Karagiannis15].

## 5.6. DDS

Data Distribution Service (DDS) is another publish/subscribe protocol that is designed by the Object Management Group (OMG) for M2M communications [DDS]. The basic benefit of this

protocol is the excellent quality of service levels and reliability guarantees as it relies on a broker-less architecture, which suits IoT and M2M communication. It offers 23 quality-of-service levels which allow it to offer a variety of quality criteria including: security, urgency, priority, durability, reliability, etc. It defines two sublayers: data-centric publish-subscribe and data-local reconstruction sublayers. The first takes the responsibility of message delivery to the subscribers while the second is optional and allows a simple integration of DDS in the application layer. Publisher layer is responsible for sensory data distribution. Data writer interacts with the publishers to agree about the data and changes to be sent to the subscribers. Subscribers are the receivers of sensory data to be delivered to the IoT application. Data readers basically read the published data and deliver it to the subscribers and the topics are basically the data that are being published. In others words, data writers and data reader take the responsibilities of the broker in the broker-based architectures.

### 5.7. Summary

IoT has many standardized session layer protocols which were briefly highlighted in this section. These session layer protocols are application dependent and the choice between them are very application specific. It should be noted that MQTT is the most widely used in IoT due to its low overhead and power consumption It's an organizational and applications specific to choose between these standards. For example, if an application has already been built with XML and can, therefore, accept a bit of overhead in its headers, XMPP might be the best option to choose among session layer protocols. On the other hand, if the application is really overhead and power sensitive, then choosing MQTT would be the best option, however, that comes with the additional broker implementation. If the application requires REST functionality as it will be HTTP based, then CoAP would be the best option if not the only one. Table 1 summarizes comparison points between these different session layer protocols.

**Table 1: A Comparison of IoT Session Layer Standards**

| Protocols | UDP/TCP | Architecture | Security and QoS | Header Size (bytes) | Max Length(bytes) |
|---|---|---|---|---|---|
| **MQTT** | TCP | Pub/Sub | Both | 2 | 5 |
| **AMQP** | TCP | Pub/Sub | Both | 8 | - |
| **CoAP** | UDP | Req/Res | Both | 4 | 20 (typical) |
| **XMPP** | TCP | Both | Security | - | - |
| **DDS** | TCP/UDP | Pub/Sub | QoS | - | - |

# 6. IoT Management Protocols

This section discusses two main management standards for IoT that provide heterogeneous communication – communication between different datalinks. Management protocols play an important role in IoT due to the diversity of protocols and standards at different layers of networking. The need for heterogeneous and easy communication between different protocols at the same or different layers is critical for IoT applications. Existing standards mainly facilitate communication between protocols at the same layer; however, it is still a challenge to facilitate communication at different layers in IoT.

### 6.1. Interconnection of Heterogeneous Datalinks

As IoT environments rely on many different MAC protocols, interoperability among all these technologies in a challenge that needs to be handled. IEEE 1905.1 standards offer such interoperability by providing an abstraction layer that is built in top on all these heterogeneous MAC protocols [1905]. This abstraction hides the diversity of the different protocols without requiring any change to the design of each MAC, as illustrated in Section 2. The basic idea behind this protocol is the abstraction layer which is used to exchange messages, called Control Message Data Units (CMDUs) among all standards compatible devices. As shown in Figure 8, All IEEE 1905.1 compliant devices understand a common "Abstraction Layer Management Entity (ALME)" protocol which offers different services including: neighbor discovery, topology exchange, topology change notification, measured traffic statistics exchange, flow forwarding rules, and security associations.

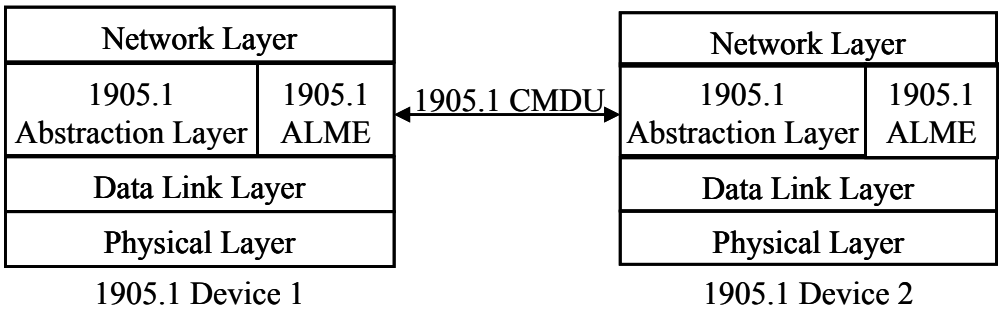


**Figure 8**: IEEE 1905.1 Protocol Structure

### 6.2. Smart Transducer Interface

IEEE 1451 is a set of standards developed to allow management of different analog transducers and sensors. The basic idea of this standard is the use of plug and play identification using standardized Transducer electronic data sheets (TEDSs). Each transducer contains a TEDS which includes all the information needed by the measurement system including device ID, characteristics and interface beside the data coming from the sensors. Data sheets are stored embedded memory within the transducer or the sensor and have a defined encoding mechanism to understand a broad number of sensor types and applications. The memory usage is minimized by utilizing the small XML based messages which are understood by different manufactures and different applications [Malar14].

## 7. Security in IoT Protocols

Security is another aspect of IoT applications which is critical and can be found in all almost all layers of the IoT protocols. Threats exist at all layers including datalink, network, session, and application layers. In this section, we briefly discuss the security mechanisms built in the IoT protocols that we have discussed in this survey.

### 7.1. MAC 802.15.4

MAC 802.15.4 offers different security modes by utilizing the “Security Enabled Bit” in the Frame Control field in the header. Security requirements include confidentiality, authentication, integrity, access control mechanisms and secured Time-Synchronized Communications.

### 7.2. 6LoWPAN

6LoWPAN by itself does not offer any mechanisms for security. However, relevant documents include discussion of security threats, requirement and approach to consider in IoT network layer. For example, RFC 4944 discusses the possibility of duplicate EUI-64 interface addresses which are supposed to be unique [RFC4944]. RFC 6282 discusses the security issues that are raised due to the problems introduced in RFC 4944 [RFC6282]. RFC 6568 addresses possible mechanisms to adopt security within constrained wireless sensor devices [RFC6568]. In addition, a few recent drafts in [6Lo] discuss mechanisms to achieve security in 6loWPAN. See also [Pongle15, Wallgren13].

### 7.3. RPL

RPL offers different level of security by utilizing a “Security” field after the 4-byte ICMPv6 message header. Information in this field indicates the level of security and the cryptography algorithm used to encrypt the message. RPL offers support for data authenticity, semantic security, protection against replay attacks, confidentiality and key management. Levels of security in RPL include Unsecured, Preinstalled, and Authenticated. RPL attacks include Selective Forwarding, Sinkhole, Sybil, Hello Flooding, Wormhole, Black hole and Denial of Service attacks.

### 7.4. Application Layer

Applications can provide additional level of security using TLS or SSL as a transport layer protocol. In addition, end to end authentication and encryption algorithms can be used to handle different levels of security as required. For further discussion on security, see [Granjal15]

It should be noted that a number of new security approaches are also being developed that are suitable for resource constrained IoT devices. Some of these protocols are listed in Figure 2.

## 8. IoT Challenges

Developing a successful IoT application is still not an easy task due to multiple challenges. These challenges include: mobility, reliability, scalability, management, availability, interoperability, and security and privacy. In the following, we briefly describe each of these challenges.

## 8.1. Mobility

IoT devices need to move freely and change their IP address and networks based on their location. Thus, the routing protocol, such as RPL has to reconstruct the DODAG each time a node goes off the network or joins the network which adds a lot of overhead. In addition, mobility might result in a change of service provider which can add another layer of complexity due to service interruption and changing gateway.

## 8.2. Reliability

System should be perfectly working and delivering all of its specifications correctly. It is a very critical requirement in applications that requires emergency responses. In IoT applications, the system should be highly reliable and fast in collecting data, communicating them and making decisions and eventually wrong decisions can lead to disastrous scenarios.

## 8.3. Scalability

Scalability is another challenge of IoT applications where millions and trillions of devices could be connected on the same network. Managing their distribution is not an easy task. In addition, IoT applications should be tolerant of new services and devices constantly joining the network and, therefore, must be designed to enable extensible services and operations.

## 8.4. Management

Managing all These devices and keeping track of the failures, configurations, and performance of such large number of devices is definitely a challenge in IoT. Providers should manage Fault, Configuration, Accounting, Performance and Security (FCAPS) of their interconnected devices and account for each aspect.

## 8.5. Availability

Availability of IoT includes software and hardware levels being provided at anytime and anywhere for service subscribers. Software availability means that the service is provided to anyone who is authorized to have it. Hardware availability means that the existing devices are easy to access and are compatible with IoT functionality and protocols. In addition, these protocols should be compact to be able to be embedded within the IoT constrained devices.

## 8.6. Interoperability

Interoperability means that heterogeneous devices and protocols need to be able to inter-work with each other. This is challenging due to the large number of different platforms used in IoT systems. Interoperability should be handled by both the application developers and the device manufacturers in order to deliver the services regardless of the platform or hardware specification used by the customer.

# 9. Summary

In this chapter, we have provided a comprehensive survey of protocols for IoT. Many such protocols have been developed by IETF, IEEE, ITU, and other organizations and many more in development. Due to their large number, the discussion of each protocol is brief and references for further information have been provided. The aim of this chapter is to give an insight to developers and service providers of different layers of protocols in IoT and how to choose between them.

We categorized the standards based on their layer of operation to: datalink layer, network routing standards, network encapsulation layer, session layer, and management standards. At each layer, we presented most of the finalized standards and some drafts. In addition, we briefly reviewed IoT management protocols and current state of security issues related to these protocols. We also provided a brief comparison between different IoT protocols and how to choose between them. Finally, we discussed some challenges that still exist in IoT systems and researchers are trying to solve them.

# List of Acronyms

| | |
|---|---|
| 6Lo | IPv6 over Networks of Resource Constrained Nodes |
| 6LoWPAN | IPv6 over Low Power Wireless Personal Area Networks |
| 6TiSCH | IPv6 over Time Slotted Channel Hopping Mode of IEEE 802.15.4e |
| ACK | Acknowledgement |
| ALME | Abstraction Layer Management Entity |
| AMQP | The Advanced Message Queuing Protocol |
| AV | Audio-Visual |
| CA | Collision Avoidance |
| CARP | Channel-Aware Routing Protocol |
| CMDUs | Control Message Data Units |
| CoAP | Constrained Application Protocol |
| CoRE | Constrained RESTful Environment |
| CORPL | Cognitive RPL |
| CRC | Cyclic redundancy check |
| CSMA | Carrier Sense Multiple Access |
| CSMA/CA | Carrier Sense Multiple Access with Collision Avoidance |
| DAO | Destination Advertisement Object |
| DAO-ACK | DAO Acknowledgment |
| DASH7 | Named after last two characters in ISO 18000-7 |
| DDS | Data Distribution Service |
| DECT | Digital Enhanced Cordless Telephone |
| DECT/ULE | Digital Enhanced Cordless Telephone with Ultra Low Energy |
| DIO | DODAG Information Object |
| DIS | DODAG Information Solicitation |
| DODAG | Destination Oriented Directed Acyclic Graph |
| eNB | E-UTRAN Node B (4G Base station) |

| | |
|---|---|
| EUI-64 | Extended Unique Identifier 64-bit |
| FCAPS | Fault, Configuration, Accounting, Performance and Security |
| FDMA | Frequency division multiple access |
| GHz | Giga Hertz |
| HART | Highway Addressable Remote Transducer Protocol |
| HomePlug-AV | HomePlug Audio-Visual |
| HomePlugGP | HomePlug GreenPHY |
| IBM | International Business Machine Corporation |
| ICMPv6 | Internet Control Message Protocol Version 6 |
| ID | Identifier |
| IEEE | Institution of Electrical and Electronic Engineers |
| IETF | Internet Engineering Task Force |
| IoT | Internet of Things |
| IP | Internet Protocol |
| IPv6 | Internet Protocol version 6 |
| ISM | Industrial, Scientific and Medical frequency band |
| ITU-T | International Telecommunications Union - Telecommunications |
| ITU | International Telecommunications Union |
| L2CAP | Logical Link Control and Adaptation Protocol |
| LoRaWAN | Long Range Wide Area Network |
| LTE-A | Long-Term Evolution Advanced |
| LTE | Long-Term Evolution |
| M2M | Machine to Machine |
| MAC | Media Access Control |
| MQTT | Message Queue Telemetry Transport |
| MS/TP | Master-Slave/Token Passing |
| NFC | Near Field Communication |
| OASIS | Advancing Open Standards in the Information Society |
| OFDM | Orthogonal Frequency Division Multiplexing |
| OMG | Object Management Group |
| PA | Process Automation |
| PHY | Physical Layer |
| QoS | Quality of Service |
| RAN | Radio Access Network |
| REST | Representational State Transfer |
| RESTful | Representational State Transfer based |
| RFC | Request for Comments |
| RFID | Radio-frequency identification |
| RPL | Routing Protocol for Low-Power and Lossy Networks |
| SDN | Software Defined Networking |
| SIG | Special Interest Group |
| SMQTT | Secure MQTT |
| SOA | Services Oriented Architecture |
| SSL | Secure Socket Layer |
| TCP | Transmission Control Protocol |
| TDMA | Time Division Multiple Access |

| | |
|---|---|
| TEDS | Transducer Electronic Data sheets |
| TLS | Transport Level Security |
| TSCH | Time-Slotted Channel Hopping |
| UDP | User Datagram Protocol |
| ULE | Ultra-Low Energy |
| WIA-PA | Wireless Networks for Industrial Automation Process Automation |
| WiFi | Wireless Fidelity |
| WirelessHART | Wireless Highway Addressable Remote Transducer Protocol |
| WPAN | Wireless Personal Area Network |
| XML | Extensible Markup Language |
| XMPP | Extensible Messaging and Presence Protocol |

# References


[1905] IEEE 1905.1-2013, “IEEE Standard for a Convergent Digital Home Network for Heterogeneous Technologies," 93 pp., April 12 2013, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6502164

[6Lo] IETF, “IPv6 over Networks of Resource-constrained Nodes (6lo),” https://datatracker.ietf.org/wg/6lo/documents/

[802.15.4] IEEE 802.15.4-2011, “IEEE Standard for Local and metropolitan area networks--Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)," 314 pp., Sept. 5 2011, http://standards.ieee.org/getieee802/download/802.15.4-2011.pdf

[Aijaz15] A. Aijaz and A. Aghvami, "Cognitive machine-to-machine communications for internet-of-things: A protocol stack perspective," IEEE Internet of Things Journal, vol. 2, no. 2, pp. 103-112, April 2015, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=7006643

[AMQP12] OASIS, "OASIS Advanced Message Queuing Protocol (AMQP) Version 1.0," 2012, http://docs.oasis-open.org/amqp/core/v1.0/os/amqp-core-complete-v1.0-os.pdf

[Cetinkaya15] O. Cetinkaya and O. Akan, "A dash7-based power metering system," in 12th Annual IEEE Consumer Communications and Networking Conference (CCNC), Jan 2015, pp. 406-411, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?reload=true&arnumber=7158010

[DDS] Object Management Group, "Data Distribution Service V1.4," April 2015, http://www.omg.org/spec/DDS/1.4

[Decuir10] J. Decuir, "Bluetooth 4.0: Low Energy", Presentation slides, 2010, http://chapters.comsoc.org/vancouver/BTLER3.pdf

[Dujovne14] D. Dujovne, T. Watteyne, X. Vilajosana, and P. Thubert, "6TiSCH: Deterministic IP-enabled industrial internet (of things)," IEEE Communications Magazine, vol. 52,

no. 12, pp. 36-41, December 2014, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6979984

[Fuaha15] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols and applications," IEEE Communications Surveys Tutorials, vol. PP, no. 99, 2015, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?tp=&arnumber=7123563

[G9959] ITU-T, "Short range narrow-band digital radio communication transceivers - PHY and MAC layer specifications," 2012, http://freepdfs.net/itu-t-rec-g9959-022012-short-range-narrow-band-digital/72f15e8ad3d84d4e80069533db810234/

[Gartner14] Gartner, "Gartner's 2014 hype cycle for emerging technologies maps the journey to digital business,", August 2014, http://www.gartner.com/newsroom/id/2819918

[Gomez12] C. Gomez, J. Oller, and J. Paradells, "Overview and evaluation of Bluetooth low energy: An emerging low-power wireless technology," Sensors, vol. 12, no. 9, pp. 11734-11753, 2012, http://www.mdpi.com/1424-8220/12/9/11734

[Granjal15] J. Granjal, E. Monteiro, and J. Sa Silva, "Security for the internet of things: A survey of existing protocols and open research issues," IEEE Communications Surveys Tutorials, vol. 17, no. 3, pp. 1294-1312, 2015, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7005393

[Hasan13] M. Hasan, E. Hossain, D. Niyato, "Random access for machine-to-machine communication in LTE-advanced networks: issues and approaches," in IEEE Communications Magazine, vol. 51, no. 6, pp. 86-93, June 2013, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?reload=true&arnumber=6525600

[HomePlug] HomePlug Alliance, "Homeplug GreenPHY v1.1," 2012, http://www.homeplug.org/tech-resources/resources/

[Karagiannis15] V. Karagiannis, P. Chatzimisios, F. Vazquez-Gallego, and J. Alonso-Zarate, "A survey on application layer protocols for the internet of things," Transaction on IoT and Cloud Computing, vol. 3, no. 1, pp. 11-17, 2015, https://jesusalonsozarate.files.wordpress.com/2015/01/2015-transaction-on-iot-and-cloud- computing.pdf

[Kim08] A. Kim, F. Hekland, S. Petersen, and P. Doyle, "When HART goes wireless: Understanding and implementing the WirelessHART standard," in IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2008), Sept 2008, pp. 899-907, https://library.e.abb.com/public/eb20fe80a391ca8485257bc600667573/When%20HART%20Goes%20Wireless%20Understanding%20and%20Implementing%20the%20WirelessHART%20Standard.pdf

[Locke10] D. Locke, "MQ telemetry transport (MQTT) v3. 1 protocol specification," IBM Developer Works Technical Library, 2010, http://www.ibm.com/developerworks/webservices/library/ws-mqtt/index.html

[Lorawan15] LoRa Alliance, "LoRaWAN specification," 2015, https://www.lora-alliance.org/portals/0/specs/LoRaWAN%20Specification%201R0.pdf

[Malar14] K. Malar and N. Kamaraj, "Development of smart transducers with IEEE 1451.4 standard for industrial automation," in 2014 International Conference on Advanced Communication Control and Computing Technologies (ICACCCT), May 2014, pp. 111-114, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?&arnumber=7019280

[Park15] M. Park, "IEEE 802.11ah: sub-1-GHz license-exempt operation for the internet of things," in IEEE Communications Magazine, vol.53, no.9, pp.145-151, September 2015, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7263359

[Pongle15] P. Pongle and G. Chavan, "A survey: Attacks RPL and 6LowPAN in IoT," in International Conference on Pervasive Computing (ICPC 2015), Jan 2015, pp. 1-6, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7087034

[Poole14] I. Poole, "Weightless wireless - M2M white space communications - tutorial," 2014, http://www.radio-electronics.com/info/wireless/weightless-m2m-white-space-wireless-communications/basics-overview.php

[Raza10] S. Raza, T. Voigt, "Interconnecting WirelessHART and legacy HART networks," in 6th IEEE International Conference on Distributed Computing in Sensor Systems Workshops (DCOSSW), 2010, pp.1-8, 21-23 June 2010, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5593285

[RFC4944] G. Montenegro, et al, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks," IETF RFC 4944, Sep 2007, https://tools.ietf.org/html/rfc4944

[RFC6120] P. Saint-Andre, "Extensible messaging and presence protocol (XMPP): Core," IETF RFC 6120, 2011, https://tools.ietf.org/html/rfc6120

[RFC6282] J. Hui and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-based Networks," IETF RFC 6262, Sep 2011, https://tools.ietf.org/html/rfc6282

[RFC6550] T. Winter, et al, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks," IETF RFC 6550, Mar. 2012, http://www.ietf.org/rfc/rfc6550.txt

[RFC6568] E. Kim, D. Kaspar, and J. Vasseur, "Design and Application Spaces for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)," IETF RFC 6568, Apr. 2012, http://www.ietf.org/rfc/rfc6568.txt

[RFC7252] Z. Shelby, K. Hartke, and C. Bormann, "The Constrained Application Protocol (CoAP)," IETF RFC 7252, Jun. 2014, http://www.ietf.org/rfc/rfc7252.txt

[RFC7428] A. Brandt and J. Buron, "Transmission of IPv6 Packets over ITU-T G.9959 Networks," IETF RFC 7428, Feb. 2015, http://www.ietf.org/rfc/rfc7428.txt

[RFC7668] J. Nieminen, et al, "IPv6 over Bluetooth Low Energy," IETF RFC 7668, October 2015, http://www.ietf.org/rfc/rfc7668.txt

[Sheng13] Z. Sheng, S. Yang, Y. Yu, A. Vasilakos, J. McCann, and K. Leung, "A survey on the IETF protocol suite for the internet of things: standards, challenges, and opportunities," IEEE Wireless Communications, vol. 20, no. 6, pp. 91- 98, December 2013, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6704479

[Shou15] Z. Zhou, B. Yao, R. Xing, L. Shu, and S. Bu, "E-CARP: An energy efficient routing protocol for UWSNs in the internet of underwater things," IEEE Sensors Journal, vol. PP, no. 99, 2015, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7113774

[Singh15] M. Singh, M. Rajan, V. Shivraj, and P. Balamuralidhar, "Secure MQTT for Internet of Things (IoT)," in Fifth International Conference on Communication Systems and Network Technologies (CSNT 2015), April 2015, pp. 746-751, http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=7280018

[Sush2015] S. Bush, "DECT/ULE connects homes for IoT," Electronics weekly, Sepetember 2015, http://www.electronicsweekly.com/news/design/communications/dect-ule-connects-homes-iot-2015-09/

[Wallgren13] L. Wallgren, S. Raza, and T. Voigt, "Routing attacks and countermeasures in the RPL-based internet of things," International Journal of Distributed Sensor Networks, vol. 2013, 2013, http://www.hindawi.com/journals/ijdsn/2013/794326/

[Z-Wave] Z-Wave, "Z-Wave Protocol Overview," v. 4, May 2007, https://wiki.ase.tut.fi/courseWiki/images/9/94/SDS10243_2_Z_Wave_Protocol_Overview.pdf

[ZigBee08] ZigBee Standards Organization, "ZigBee Specification," Document 053474r17, Jan 2008, 604 pp., http://home.deib.polimi.it/cesana/teaching/IoT/papers/ZigBee/ZigBeeSpec.pdf