Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: April 22, 2015

Q. Fu, Ed.
China Mobile

October 19, 2014

Deployment of the Low Weight IETF protocols In Internet of Things(IOT) draft-fu-lwig-iot-usecase-00

Abstract

This draft analyze the development and deployment of the existing IETF Low weight IPv6 protocols in the IOT (Internet Of Things) industry. Taking consideration on the constrained resource nature of devices, the IETF low weight IPv6 protocols, including 6LowPan, RPC and COAP, fit perfectly in the IOT scenarios. Recent development and promotion of Zigbee IP and IPSO also extend the use of these low weight IPv6 protocols.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 22, 2015.

Copyright Notice

Table of Contents

1. Introduction

With the fast development of wireless sensor technology and IC technology, the concept of IOT (Internet Of Things) has been realized and promoted in the Information Industry. IOT intends to build a network to connect all devices, systems and services, which claims a vast need of IP addresses.Due to its inherent advantage of huge address pool, IPv6 has been chosen as the fundamental Internet protocol by IETF ever since the research of IOT.

Due to the constrained resource nature of devices in IOT, working groups in IETF mainly focus on low weight IPv6 protocols, which include 6LowPan working group, RoLL working group, and CoRE working group. Other standardization organizations, such as IPSO, Zigbee, ISA and etc., are dedicated in promoting the deployment of these protocols. With years of research and development, a number of application cases and solutions for IOT based on IPv6 have been proposed and deployed. In this draft, we will summarize the latest deployments and usecases of protocols about IPv6 in IETF.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. ZigBee IP

ZigBee is the industry alliance based on IEEE 802.15.4. It mainly focuses on standarization of network laryer and application layer in short range wireless communication.The PHY and MAC layer of ZigBee is IEEE 802.15.4.

ZigBee IP is the first open standard for an IPv6-based full wireless mesh networking solution and provides seamless Internet connections to control low-power, low-cost devices. It connects dozens of the different devices into a single control network.

ZigBee IP was designed to support the ZigBee Smart Energy version 2 standard, published in 2013, which offers a global standard for IP-based control,both wired and wireless, for energy management in Home Area Networks(HANs).Such standard is expected to be used in Smart Grid applications.

Zigbee IP has been recently updated to include 920IP, published in July, 2014,which provides specific support for ECHONET Lite and the requirements of Japanese Home Energy Management systems. 920IP was developed in response to Japan’s Ministry of Internal Affairs and Communications (MIC) designation of 920 MHz for use in HEMS and Ministry of Economy, Trade, and Industry(METI) endorsement of ECHONET Lite as a smart home standard. 920IP is the only standard referenced by the Telecommunications Technology Committee (TTC) which supports multi-hop mesh networking.

The ZigBee IP specification enriches the IEEE 802.15.4 standard by adding network and security layers and an application framework. ZigBee IP offers a scalable architecture with end-to-end IPv6 networking, laying the foundation for an Internet of Things without the need for intermediate gateways. It offers cost-effective and energy-efficient wireless mesh network based on standard Internet protocols, such as 6LoWPAN, IPv6, PANA, RPL, TCP, TLS and UDP. It also features proven, end-to-end security using TLS1.2 protocol, link layer frame security based on AES-128-CCM algorithm and support for public key infrastructure using standard X.509 v3 certificates and ECC-256 cipher suite. ZigBee IP enables low-power devices to participate natively with other IPv6-enabled Ethernet, Wi-Fi and, HomePlug devices.

ZigBee IP has been paid great attention once it was published in 2013. Several chip companies, including Exegin, Silicon Labs, TI and etc., have developed chips that support ZigBee IP.

4. IPSO

The IPSO Alliance performs interoperability tests, documents the use of new IP-based technologies, conducts marketing activities and serves as an information repository for users seeking to understand the role of IP in networks of physical objects. Its role complements the work of entities such as the Internet Engineering Task Force (IETF), Institute of Electrical and Electronics Engineers (IEEE) or

the ISA which develop and ratify technical standards in the Internet community.

The ultimate goal of IPSO is to promote the use of IP in the IOT. Considering the constraint nature of the IOT devices, several technologies, including Lightweight OS, 6LowPan, COAP, and RPL, are promoted in IPSO. Members of IPSO have developed large number of commercial devices that support IP-communication. For example, Toshiba has developed a IPv6-capable TV. Axis provides IPv6-capable camera for surveillance use.

5. Usecases for Operators

The broad IOT market brings opportunities for operators world wide. Interesting deployments of IOT or M2M (Machine to Machine) cases based on IPv6 have emerged over the years.For example, the French Telecom deploys IPv6 based M2M network in Smart Metering, Intelegent Health Monitering, and Smart City. Considering the constraint nature of the devices, low weight IPv6 protocols, including 6LowPan, RPL, and COAP are utilized.

6. Conclusion

This draft introduces the recent development and deployments of low weight IPv6 protocols studied in IETF. Taking careful consideration on the constrained resource nature of the devices in IOT, these low weight IPv6 protocols,including 6LowPan, RPL, and COAP, are proved to be quite a success in the IOT scenarios. Generation and improvements of industrial standards, such as Zigbee IP, also accelerate the deployment of the low weight IPv6 protocols. We can predice that IOT might be a "killer scenario" of the extensive deployment of IPv6 world wide in the near future.

7. Informative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

Author’s Address

Qiao Fu (editor)
China Mobile
Xuanwumenxi Ave. No.32
Beijing
China

Email: fuqiao1@outlook.com

Internet Engineering Task Force
Internet-Draft
Intended status: Informational
Expires: April 24, 2018

C. Gomez
Universitat Politecnica de Catalunya
M. Kovatsch
ETH Zurich
H. Tian
China Academy of Telecommunication Research
Z. Cao, Ed.
Huawei Technologies
October 21, 2017

Energy-Efficient Features of Internet of Things Protocols
draft-ietf-lwig-energy-efficient-08

Abstract

This document describes the challenges for energy-efficient protocol operation on constrained devices and the current practices used to overcome those challenges. It summarizes the main link-layer techniques used for energy-efficient networking, and it highlights the impact of such techniques on the upper layer protocols so that they can together achieve an energy efficient behavior. The document also provides an overview of energy-efficient mechanisms available at each layer of the IETF protocol suite specified for constrained node networks.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 24, 2018.

Table of Contents

1. Introduction

Network systems for physical world monitoring contain many battery-powered or energy-harvesting devices. For example, in an environmental monitoring system, or a temperature and humidity

monitoring system, there may not be always-on and sustained power supplies for the potentially large number of constrained devices. In such deployment scenarios, it is necessary to optimize the energy consumption of the constrained devices. In this document we describe techniques that are in common use at Layer 2 and at Layer 3, and we indicate the need for higher-layer awareness of lower-layer features.

Many research efforts have studied this "energy efficiency" problem. Most of this research has focused on how to optimize the system's power consumption in certain deployment scenarios, or how an existing network function such as routing or security could be more energy-efficient. Only few efforts have focused on energy-efficient designs for IETF protocols and standardized network stacks for such constrained devices [I-D.kovatsch-lwig-class1-coap].

The IETF has developed a suite of Internet protocols suitable for such constrained devices, including IPv6 over Low-Power Wireless Personal Area Networks (6LoWPAN) [RFC6282],[RFC6775],[RFC4944], the IPv6 Routing Protocol for Low-Power and Lossy Networks (RPL) [RFC6550], and the Constrained Application Protocol (CoAP) [RFC7252]. This document tries to summarize the design considerations for making the IETF constrained protocol suite as energy-efficient as possible. While this document does not provide detailed and systematic solutions to the energy efficiency problem, it summarizes the design efforts and analyzes the design space of this problem. In particular, it provides an overview of the techniques used by the lower layers to save energy and how these may impact on the upper layers. Cross-layer interaction is therefore considered in this document from this specific point of view. Providing further design recommendations that go beyond the layered protocol architecture is out of the scope of this document.

After reviewing the energy-efficient designs of each layer, we summarize the document by presenting some overall conclusions. Though the lower layer communication optimization is the key part of energy efficient design, the protocol design at the upper layers is also important to make the device energy-efficient.

1.1. Terminology

Terms used in this document are defined in [RFC7228] [I-D.bormann-lwig-7228bis].

2. Overview

The IETF has developed protocols to enable end-to-end IP communication between constrained nodes and fully capable nodes. This work has expedited the evolution of the traditional Internet

protocol stack to a light-weight Internet protocol stack. As shown in Figure 1 below, the IETF has developed CoAP as the application layer and 6LoWPAN as the adaption layer to run IPv6 over IEEE 802.15.4 and Bluetooth Low-Energy, with the support of routing by RPL and efficient neighbor discovery by 6LoWPAN-ND. 6LoWPAN is currently being adapted by the 6lo working group to support IPv6 over various other technologies, such as ITU-T G.9959 [G9959], DECT ULE [TS102], MS/TP-BACnet [MSTP], and Near Field Communication (NFC) [NFC].

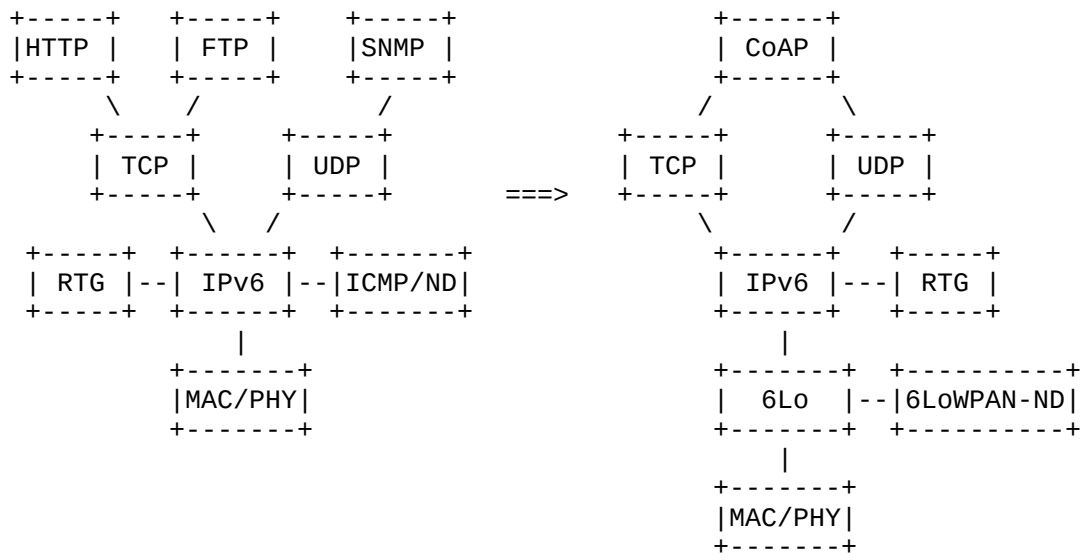

Figure 1: Traditional and Light-weight Internet Protocol Stack

There are numerous published studies reporting comprehensive measurements of wireless communication platforms [Powertrace]. As an example, below we list the energy consumption profile of the most common operations involved in communication on a prevalent sensor node platform. The measurement was based on the Tmote Sky with ContikiMAC [ContikiMAC] as the radio duty cycling algorithm. From this and many other measurement reports (e.g.[AN079]), we can see that the energy consumption of optimized transmission and reception are in the same order. For IEEE 802.15.4 and Ultra WideBand (UWB) links, transmitting may actually be even cheaper than receiving. It also shows that broadcast and non-synchronized communication transmissions are energy costly because they need to acquire the medium for a long time.

Activity	Energy (uJ)
Broadcast reception	178
Unicast reception	222
Broadcast transmission	1790
Non-synchronized unicast transmission	1090
Synchronized unicast transmission	120
Unicast TX to awake receiver	96
Listening (for 1000 ms)	63000

Figure 2: Power consumption of common operations involved in communication on the Tmote Sky with ContikiMAC

At the Physical layer, one approach that may allow reducing energy consumption of a device that uses a wireless interface is based on reducing the device transmit power level as long as the intended next hop(s) are still within range of the device. In some cases, if node A has to transmit a message to node B, a solution to reduce node A transmit power is to leverage an intermediate device, e.g. node C as a message forwarder. Let d be the distance between node A and node B. Assuming free-space propagation, where path loss is proportional to d^2, if node C is placed right in the middle of the path between A and B (that is, at a distance d/2 from both node A and node B), the minimum transmit power to be used by node A (and by node C) is reduced by a factor of 4. However, this solution requires additional devices, it requires a routing solution, and it also increases transmission delay between A and B.

3. Medium Access Control and Radio Duty Cycling

In networks, communication and power consumption are interdependent. The communication device is typically the most power-consuming component, but merely refraining from transmissions is not enough to achieve a low power consumption: the radio may consume as much power in listen mode as when actively transmitting. This illustrates the key problem known as idle listening, whereby the radio of a device may be in receive mode (ready to receive any message), even if no message is being transmitted to that device. Idle listening can consume a huge amount of energy unnecessarily. To reduce power consumption, the radio must be switched completely off -- duty-cycled

-- as much as possible. By applying duty-cycling, the lifetime of a device operating on a common button battery may be on the order of years, whereas otherwise the battery may be exhausted in a few days or even hours. Duty-cycling is a technique generally employed by devices that use the P1 strategy [RFC7228], which need to be able to communicate on a relatively frequent basis. Note that a more aggressive approach to save energy relies on the P0, Normally-off strategy, whereby devices sleep for very long periods and communicate infrequently, even though they spend energy in network reattachment procedures.

From the perspective of Medium Access Control (MAC) and Radio Duty Cycling (RDC), all upper layer protocols, such as routing, RESTful communication, adaptation, and management flows, are applications. Since the duty cycling algorithm is the key to energy-efficiency of the wireless medium, it synchronizes transmission and/or reception requests from the higher layers.

MAC and RDC are not in the scope of the IETF, yet lower layer designers and chipset manufacturers take great care to save energy. By knowing the behaviors of these lower layers, IETF engineers can design protocols that work well with them. The IETF protocols to be discussed in the following sections are the customers of the lower layers.

3.1. Radio Duty Cycling techniques

This subsection describes three main three RDC techniques. Note that more than one of these techniques may be available or can even be combined in a specific radio technology:

a) Channel sampling. In this solution, the radio interface of a device periodically monitors the channel for very short time intervals (i.e. with a low duty cycle) with the aim of detecting incoming transmissions. In order to make sure that a receiver can correctly receive a transmitted data unit, the sender may prepend a preamble of a duration at least the sampling period to the data unit to be sent. Another option for the sender is to repeatedly transmit the data unit, instead of sending a preamble before the data unit. Once a transmission is detected by a receiver, the receiver may stay awake until the complete reception of the data unit. Examples of radio technologies that use preamble sampling include ContikiMAC, the Coordinated Sampled Listening (CSL) mode of IEEE 802.15.4e, and the Frequently Listening (FL) mode of ITU-T G.9959 [G9959].

b) Scheduled transmissions. This approach allows a device to know the particular time at which it should be awake (during some time interval) in order to receive data. Otherwise, the device may remain

in sleep mode. The decision on the times at which communication is attempted relies on some form of negotation between the involved devices. Such negotiation may be performed per transmission or per session/connection. Bluetooth Low Energy (Bluetooth LE) is an example of a radio technology based on this mechanism.

c) Listen after send. This technique allows a node to remain in sleep mode by default, wake up and poll a sender (which must be ready to receive a poll message) for pending transmissions. After sending the poll message, the node remains in receive mode, ready for a potential incoming transmission. After a certain time interval, the node may go back to sleep. For example, the Receiver Initiated Transmission (RIT) mode of 802.15.4e, and the transmission of data between a coordinator and a device in IEEE 802.15.4-2003 use this technique.

3.2. Latency and buffering

The latency of a data unit transmission to a duty-cycled device is equal to or greater than the latency of transmitting to an always-on device. Therefore, duty-cycling leads to a trade-off between energy consumption and latency. Note that in addition to a latency increase, RDC may introduce latency variance, since the latency increase is a random variable (which is uniformly distributed if duty-cycling follows a periodical behavior).

On the other hand, due to the latency increase of duty-cycling, a sender waiting for a transmission opportunity may need to store subsequent outgoing packets in a buffer, increasing memory requirements and potentially incurring queuing waiting time that contributes to the packet’s overall delay and increases the probability of buffer overflow, leading to losses.

3.3. Throughput

Although throughput is not typically a key concern in constrained node network applications, it is indeed important in some services in such networks, such as over-the-air software updates or when off-line sensors accumulate measurements that have to be quickly transferred when there is an opportunity for connectivity.

Since RDC introduces inactive intervals in energy-constrained devices, it reduces the throughput that can be achieved when communicating with such devices. There exists a trade-off between the achievable throughput and energy consumption.

3.4. Radio interface tuning

The parameters controlling the radio duty cycle have to be carefully tuned to achieve the intended application and/or network requirements. On the other hand, upper layers should take into account the expected latency and/or throughput behavior due to RDC. The next subsection provides details on key parameters controlling RDC mechanisms, and thus fundamental trade-offs, for various examples of relevant low-power radio technologies.

3.5. Packet bundling

Another technique that may be useful to increase communication energy efficiency is packet bundling. This technique, which is available in several radio interfaces (e.g. LTE and some 802.11 variants), allows to aggregate several small packets into a single large packet. Header and communication overhead is therefore reduced.

3.6. Power save services available in example low-power radios

This subsection presents power save services and techniques used in a few relevant examples of wireless low-power radios: IEEE 802.11, Bluetooth LE and IEEE 802.15.4. For a more detailed overview of each technology, the reader may refer to the literature or to the corresponding specifications.

3.6.1. Power Save Services Provided by IEEE 802.11

IEEE 802.11 defines the Power Save Mode (PSM) whereby a station may indicate to an Access Point (AP) that it will enter a sleep mode state. While the station is sleeping, the AP buffers any frames that should be sent to the sleeping station. The station wakes up every Listen Interval (which can be a multiple of the Beacon Interval) in order to receive beacons. The AP signals in the beacon whether there is data pending for the station or not. If there are not frames to be sent to the station, the latter may get back to sleep mode. Otherwise, the station may send a message requesting the transmission of the buffered data and stay awake in receive mode.

IEEE 802.11v [IEEE80211v] further defines mechanisms and services for power save of stations/nodes that include flexible multicast service (FMS), proxy ARP advertisement, extended sleep modes, and traffic filtering. Upper layer protocols knowledge of such capabilities provided by the lower layer enables better interworking.

These services include:

Proxy ARP: The Proxy ARP capability enables an Access Point (AP) to indicate that the non-AP station (STA) will not receive ARP frames. The Proxy ARP capability enables the non-AP STA to remain in power-save for longer periods of time.

Basic Service Set (BSS) Max Idle Period management enables an AP to indicate a time period during which the AP does not disassociate a STA due to non-receipt of frames from the STA. This supports improved STA power saving and AP resource management.

FMS: A service in which a non-access point (non-AP) STA can request a multicast delivery interval longer than the delivery traffic indication message (DTIM) interval for the purposes of lengthening the period of time a STA may be in a power save state.

Traffic Filtering Service (TFS): A service provided by an access point (AP) to a non-AP STA that can reduce the number of frames sent to the STA by dropping individually addressed frames that do not match traffic filters specified by the STA.

Using the above services provided by the lower layer, the constrained nodes can achieve either client initiated power save (via TFS) or network assisted power save (Proxy-ARP, BSS Max Idel Period and FMS).

Upper layer protocols should synchronize with the parameters such as FMS interval and BSS MAX Idle Period, so that the wireless transmissions are not triggered periodically.

3.6.2. Power Save Services Provided by Bluetooth LE

Bluetooth LE is a wireless low-power communications technology that is the hallmark component of the Bluetooth 4.0, 4.1, and 4.2 specifications [Bluetooth42]. BT-LE has been designed for the goal of ultra-low-power consumption. IPv6 can be run IPv6 over Bluetooth LE networks by using a 6LoWPAN variant adapted to BT-LE [RFC7668].

Bluetooth LE networks comprise a master and one or more slaves which are connected to the master. The Bluetooth LE master is assumed to be a relatively powerful device, whereas a slave is typically a constrained device (e.g. a class 1 device).

Medium access in Bluetooth LE is based on a Time Division Multiple Access (TDMA) scheme which is coordinated by the master. This device determines the start of connection events, in which communication between the master and a slave takes place. At the beginning of a connection event, the master sends a poll message, which may encapsulate data, to the slave. The latter must send a response, which may also contain data. The master and the slave may continue

exchanging data until the end of the connection event. The next opportunity for communication between the master and the slave will be in the next connection event scheduled for the slave.

The time between consecutive connection events is defined by the connInterval parameter, which may range between 7.5 ms and 4 s. The slave may remain in sleep mode since the end of its last connection event until the beginning of its next connection event. Therefore, Bluetooth LE is duty-cycled by design. Furthermore, after having replied to the master, a slave is not required to listen to the master (and thus may keep the radio in sleep mode) for connSlaveLatency consecutive connection events. connSlaveLatency is an integer parameter between 0 and 499 which should not cause link inactivity for more than connSupervisionTimeout time. The connSupervisionTimeout parameter is in the range between 100 ms and 32 s.

Upper layer protocols should take into account the medium access and duty-cycling behavior of Bluetooth LE. In particular, connInterval, connSlaveLatency and connSupervisionTimeout determine the time between two consecutive connection events for a given slave. The upper layer packet generation pattern and rate should be consistent with the settings of the aforementioned parameters (and vice versa). For example, assume connInterval=4 seconds, connSlaveLatency=7 and connSupervisionTimeout=32 seconds. With these settings, communication opportunities between a master and a slave will occur during a given interval every 32 seconds. Duration of the interval will depend on several factors, including number of connected slaves, amount of data to be transmitted, etc. In the worst case, only one data unit can be sent from master to slave and vice versa every 32 seconds.

3.6.3. Power Save Services in IEEE 802.15.4

IEEE 802.15.4 is a family of standard radio interfaces for low-rate, low-power wireless networking [fifteendotfour]. Since the publication of its first version in 2003, IEEE 802.15.4 has become the de-facto choice for a wide range of constrained node network application domains and has been a primary target technology of various IETF working groups such as 6LoWPAN [RFC6282], [RFC6775], [RFC4944] and 6TiSCH [I-D.ietf-6tisch-architecture]. IEEE 802.15.4 specifies a variety of related PHY and MAC layer functionalites.

IEEE 802.15.4 defines three roles called device, coordinator and Personal Area Network (PAN) coordinator. The device role is adequate for nodes that do not implement the complete IEEE 802.15.4 functionality, and is mainly targeted for constrained nodes with a limited energy source. The coordinator role includes synchronization

capabilities and is suitable for nodes that do not suffer severe constraints (e.g. a mains-powered node). The PAN coordinator is a special type of coordinator that acts as a principal controller in an IEEE 802.15.4 network.

IEEE 802.15.4 defines two main types of networks depending on their configuration: beacon-enabled and nonbeacon-enabled networks. In the first network type, coordinators periodically transmit beacons. The time between beacons is divided in three main parts: the Contention Access Period (CAP), the Contention Free Period (CFP) and an inactive period. In the first period, nodes use slotted Carrier Sense Multiple Access / Collision Avoidance (CSMA/CA) for data communication. In the second one, a TDMA scheme controls medium access. During the idle period, communication does not take place, thus the inactive period is a good opportunity for nodes to turn the radio off and save energy. The coordinator announces in each beacon the list of nodes for which data will be sent in the subsequent period. Therefore, devices may remain in sleep mode by default and wake up periodically to listen to the beacons sent by their coordinator. If a device wants to transmit data, or learns from a beacon that it is an intended destination, then it will exchange messages with the coordinator (and thus consume energy). An underlying assumption is that when a message is sent to a coordinator, the radio of the coordinator will be ready to receive the message.

The beacon interval and the duration of the beacon interval active portion (i.e. the CAP and the CFP), and thus the duty cycle, can be configured. The parameters that control these times are called macBeaconOrder and macSuperframeOrder, respectively. As an example, when IEEE 802.15.4 operates in the 2.4 GHz PHY, both times can be (independently) set to values in the range between 15.36 ms and 251.6 seconds.

In the beaconless mode, nodes use unslotted CSMA/CA for data transmission. The device may be in sleep mode by default and may activate its radio to either i) request to the coordinator whether there is pending data for the device, or ii) to transmit data to the coordinator. The wake-up pattern of the device, if any, is out of the scope of IEEE 802.15.4.

Communication between the two ends of an IEEE 802.15.4 link may also take place in a peer-to-peer configuration, whereby both link ends assume the same role. In this case, data transmission can happen at any moment. Nodes must have their radio in receive mode, and be ready to listen to the medium by default (which for battery-enabled nodes may lead to a quick battery depletion), or apply

synchronization techniques. The latter are out of the scope of IEEE 802.15.4.

The main MAC layer IEEE 802.15.4 amendment to date is IEEE 802.15.4e. This amendment includes various new MAC layer modes, some of which include mechanisms for low energy consumption. Among these, the Time-Slotted Channel Hopping (TSCH) is an outstanding mode which offers robust features for industrial environments, among others. In order to provide the functionality needed to enable IPv6 over TSCH, the 6TiSCH working group was created. TSCH is based on a TDMA schedule whereby a set of time slots are used for frame transmission and reception, and other time slots are unscheduled. The latter time slots may be used by a dynamic scheduling mechanism, otherwise nodes may keep the radio off during the unscheduled time slots, thus saving energy. The minimal schedule configuration specified in [I-D.ietf-6tisch-minimal] comprises 101 time slots; 95 of these time slots are unscheduled and the time slot duration is 15 ms.

The previously mentioned CSL and RIT are also 802.15.4e modes designed for low energy.

3.6.4. Power Save Services in DECT ULE

DECT Ultra Low Energy (DECT ULE) is a wireless technology building on the key fundamentals of traditional DECT / CAT-iq [EN300] but with specific changes to significantly reduce the power consumption at the expense of data throughput [TS102]. DECT ULE devices typically operate on special power optimized silicon, but can connect to a DECT Gateway supporting traditional DECT / CAT-iq for cordless telephony and data as well as the DECT ULE extensions. IPv6 can be run over DECT ULE by using a 6LoWPAN variant [I-D.ietf-6lo-dect-ule].

DECT defines two major roles: the Portable Part (PP) is the power constrained device, while the Fixed Part (FP) is the Gateway or base station in a star topology. DECT operates in license free and reserved frequency bands based on TDMA/FDMA and TDD using dynamic channel allocation for interference avoidance. It provides good indoor (˜50 m) and outdoor (˜300 m) coverage. It uses a frame length of 10 ms divided into 24 timeslots, and it supports connection oriented, packet data and connection-less services.

The FP usually transmits a so-called dummy bearer (beacon) that is used to broadcast synchronization, system and paging information. The slot/carrier position of this dummy bearer can automatically be reallocated in order to avoid mutual interference with other DECT signals.

At the MAC level DECT ULE communications between FP and PP are initiated by the PP. A FP can initiate communication indirectly by sending paging signal to a PP. The PP determines the timeslot and frequency on which the communication between FP and PP takes place. The PP verifies the radio timeslot/frequency position is unoccupied before it initiates its transmitter. An access-request message, which usually carries data, is sent to the FP. The FP sends a confirm message, which also may carry data. More data can be sent in subsequent frames. A MAC level automatic retransmission scheme significantly improves data transfer reliability. A segmentation and reassembly scheme supports transfer of larger higher layer SDUs and provides data integrity check. The DECT ULE packet data service ensures data integrity, proper sequencing, duplicate protection, but not guaranteed delivery. Higher layers protocols have to take this into consideration.

The FP may send paging information to PPs to trigger connection setup and indicate the required service type. The interval between paging information to a specific PP can be defined in range 10 ms to 327 seconds. The PP may enter sleep mode to save power. The listening interval is defined by the PP application. For short sleep intervals (below ~10 seconds) the PP may be able to retain synchronization to the FP dummy bearer and only turn on the receiver during the expected timeslot. For longer sleep intervals the PP can’t keep synchronization and has to search for and resynchronize to the FP dummybearer. Hence, longer sleep interval reduces the average energy consumption, but adds a energy consumption penalty for acquiring synchronization to the FP dummy bearer. The PP can obtain all information to determine paging and acquire synchronization information in a single reception of one full timeslot.

Packet data latency is normally 30 ms for short packets (below or equal to 32 octets), however if retry and back-off scenarios occur, the latency is increased. The latency can actually be reduced to about 10 ms by doing energy consuming RSSI scanning in advance. In the direction from FP to PP the latency is usually increased by the used paging interval and the sleep interval. The MAC layer can piggyback commands to improve efficiency (reduce latency) of higher layer protocols. Such commands can instruct the PP to initiate a new packet transfer in N frames without the need for resynchronization and listening to paging or instruct the PP to stay in a higher duty cycle paging detection mode.

The DECT ULE technology allows per PP configuration of paging interval, MTU size, reassembly window size and higher layer service negotiation and protocol.

4. IP Adaptation and Transport Layer

6LoWPAN provides an adaptation layer designed to support IPv6 over IEEE 802.15.4. 6LoWPAN affects the energy-efficiency problem in three aspects, as follows.

First, 6LoWPAN provides one fragmentation and reassembly mechanism which is aimed at solving the packet size issue in IPv6 and could also affect energy-efficiency. IPv6 requires that every link in the internet have an MTU of 1280 octets or greater. On any link that cannot convey a 1280-octet packet in one piece, link-specific fragmentation and reassembly must be provided at a layer below IPv6 [RFC2460]. 6LoWPAN provides fragmentation and reassembly below the IP layer to solve the problem. One of the benefits from placing fragmentation at a lower layer such as the 6LoWPAN layer is that it can avoid the presence of more IP headers, because fragmentation at the IP layer will produce more IP packets, each one carrying its own IP header. However, performance can be severely affected if, after IP layer fragmentation, then 6LoWPAN fragmentation happens as well (e.g. when the upper layer is not aware of the existence of the fragmentation at the 6LoWPAN layer). One solution is to require higher layers awareness of lower layer features and generate small enough packets to avoid fragmentation. In this regard, the Block option in CoAP can be useful when CoAP is used at the application layer [RFC 7959].

Secondly, 6LoWPAN swaps computing with communication. 6LoWPAN applies compression of the IPv6 header. Subject to the packet size limit of IEEE 802.15.4, 40 octets long IPv6 header and 8 octets or 20 octets long UDP and TCP header will consume even more packet space than the data itself. 6LoWPAN provides IPv6 and UDP header compression at the adaptation layer. Therefore, a lower amount of data will be handled by the lower layers, whereas both the sender and receiver will spend more computing power on the compression and decompression of the packets over the air. Compression can also be performed at higher layers (see Section 6.4).

Finally, the 6LoWPAN working group developed the energy-efficient Neighbor Discovery called 6LoWPAN-ND, which is an energy efficient replacement of the IPv6 ND in constrained environments. IPv6 Neighbor Discovery was not designed for non-transitive wireless links, as its heavy use of multicast makes it inefficient and sometimes impractical in a low-power and lossy network. 6LoWPAN-ND describes simple optimizations to IPv6 Neighbor Discovery, its addressing mechanisms, and duplicate address detection for Low-power Wireless Personal Area Networks and similar networks. However, 6LoWPAN ND does not modify Neighbor Unreachability Detection (NUD) timeouts, which are very short (by default three transmissions spaced

one second apart). NUD timeout settings should be tuned taking into account the latency that may be introduced by duty-cycled mechanisms at the link layer, or alternative, less impatient NUD algorithms should be considered [I-D.ietf-6man-impatient-nud].

IPv6 underlies the higher layer protocols, including both TCP/UDP transport and applications. By design, the higher-layer protocols do not typically have specific information about the lower layers, and thus cannot solve the energy-efficiency problem.

The network stack can be designed to save computing power. For example the Contiki implementation has multiple cross layer optimizations for buffers and energy management, e.g., the computing and validation of UDP/TCP checksums without the need of reading IP headers from a different layer. These optimizations are software implementation techniques, and out of the scope of IETF and the LWIG working group.

5. Routing Protocols

RPL [RFC6550] is a routing protocol designed by the IETF for constrained environments. RPL exchanges messages periodically and keeps routing states for each destination. RPL is optimized for the many-to-one communication pattern, where network nodes primarily send data towards the border router, but has provisions for any-to-any routing as well.

The authors of the Powertrace tool [Powertrace] studied the power profile of RPL. Their analysis divides the routing protocol into control and data traffic. The control plane carries ICMP messages to establish and maintain the routing states. The data plane carries any application that uses RPL for routing packets. The study has shown that the power consumption of the control traffic goes down over time in a relatively stable network. The study also reflects that the routing protocol should keep the control traffic as low as possible to make it energy-friendly. The amount of RPL control traffic can be tuned by setting the Trickle [RFC6206] algorithm parameters (i.e. Imin, Imax and k) to appropriate values. However, there exists a trade-off between energy consumption and other performance parameters such as network convergence time and robustness.

RFC 6551 [RFC6551] defines routing metrics and constraints to be used by RPL in route computation. Among others, RFC 6551 specifies a Node Energy object that allows to provide information related to node energy, such as the energy source type or the estimated percentage of remaining energy. Appropriate use of energy-based routing metrics

may help to balance energy consumption of network nodes, minimize network partitioning and increase network lifetime.

6. Application Layer

6.1. Energy efficient features in CoAP

CoAP [RFC7252] is designed as a RESTful application protocol, connecting the services of smart devices to the World Wide Web. CoAP is not a chatty protocol. It provides basic communication services such as service discovery and GET/POST/PUT/DELETE methods with a binary header.

Energy efficiency is part of the CoAP protocol design. CoAP uses a fixed-length binary header of only four bytes that may be followed by binary options. To reduce regular and frequent queries of the resources, CoAP provides an observe mode, in which the requester registers its interest of a certain resource and the responder will report the value whenever it was updated. This reduces the request response round trips while keeping information exchange a ubiquitous service; an energy-constrained server can remain in sleep mode during the period between observe notification transmissions.

Furthermore, [RFC7252] defines CoAP proxies which can cache resource representations previously provided by sleepy CoAP servers. The proxies themselves may respond to client requests if the corresponding server is sleeping and the resource representation is recent enough. Otherwise, a proxy may attempt to obtain the resource from the sleepy server.

CoAP proxy and cache functionality may also be used to perform data aggregation. This technique allows a node to receive data messages (e.g. carrying sensor readings) from other nodes in the network, perform an operation based on the content in those messages, and transmit the result of the operation. Such operation may simply be intended to use one packet to carry the readings transported in several packets (which reduces header and transmission overhead), or it may be a more sophisticated operation, possibly based on mathematical, logical or filtering principles (which reduces the payload size to be transmitted).

6.2. Sleepy node support

Beyond these features of CoAP, there have been a number of proposals to further support sleepy nodes at the application layer by leveraging CoAP mechanisms. A good summary of such proposals can be found in [I-D.rahman-core-sleepy-nodes-do-we-need], while an example application (in the context of illustrating several security

mechanisms) in a scenario with sleepy devices has been described [I-D.ietf-lwig-crypto-sensors]. Approaches to support sleepy nodes include exploiting the use of proxies, leveraging the Resource Directory [I-D.ietf-core-resource-directory] or signaling when a node is awake to the interested nodes. Recent work defines publish-subscribe and message queuing extensions to CoAP and the Resource Directory in order to support devices that spend most of their time in asleep [I-D.ietf-core-coap-pubsub]. Notably, this work has been adopted by the CoRE Working Group.

In addition to the work within the scope of CoAP to support sleepy nodes, other specifications define application layer functionality for the same purpose. The Lightweight Machine-to-Machine (LWM2M) specification from the Open Mobile Alliance (OMA) defines a Queue Mode whereby an LWM2M Server queues requests to an LWM2M Client until the latter (which may often stay in sleep mode) is online. LWM2M functionality operates on top of CoAP.

oneM2M defines a CoAP binding with an application layer mechanism for sleepy nodes [oneM2M].

6.3. CoAP timers

CoAP offers mechanisms for reliable communication between two CoAP endpoints. A CoAP message may be signaled as a confirmable (CON) message, and an acknowledgment (ACK) is issued by the receiver if the CON message is correctly received. The sender starts a Retransmission TimeOut (RTO) for every CON message sent. The initial RTO value is chosen randomly between 2 and 3 s. If an RTO expires, the new RTO value is doubled (unless a limit on the number of retransmissions has been reached). Since duty-cycling at the link layer may lead to long latency (i.e. even greater than the initial RTO value), CoAP RTO parameters should be tuned accordingly in order to avoid spurious RTOs which would unnecessarily waste node energy and other resources. On the other hand, note that CoAP can also run on top of TCP [I-D.ietf-core-coap-tcp-tls]. In that case, similar guidance applies to TCP timers, albeit with greater motivation to carefully configure TCP RTO parameters, since [RFC6298] reduced the default initial TCP RTO to 1 second, which may interact more negatively with duty-cycled links than default CoAP RTO values.

6.4. Data compression

Another method intended to reduce the size of the data units to be communicated in constrained-node networks is data compression, which allows to encode data using less bits than the original data representation. Data compression is more efficient at higher layers, particularly before encryption is used. In fact, encryption

mechanisms may generate an output that does not contain redundancy, making it almost impossible to reduce the data representation size. In CoAP, messages may be encrypted by using DTLS (or TLS when CoAP over TCP is used), which is the default mechanism for securing CoAP exchanges.

7. Summary and Conclusions

We summarize the key takeaways in this document:

a. Internet protocols designed by IETF can be considered as the customer of the lower layers (PHY, MAC, and Duty-cycling). To reduce power consumption, it is recommended that Layer 3 designs should operate based on awareness of lower-level parameters rather than treating the lower layer as a black box (Sections 4, 5 and 6).

b. It is always useful to compress the protocol headers in order to reduce the transmission/reception power. This design principle has been employed by many protocols in 6Lo and CoRE working group (Sections 4 and 6).

c. Broadcast and non-synchronized transmissions consume more than other TX/RX operations. If protocols must use these ways to collect information, reduction of their usage by aggregating similar messages together will be helpful in saving power (Sections 2 and 6.1).

d. Saving power by sleeping as much as possible is used widely (Section 3).

8. Contributors

Jens T. Petersen, RTX, contributed the section on power save services in DECT ULE.

9. Acknowledgments

Carles Gomez has been supported by the Spanish Government, FEDER and the ERDF through projects TEC2012-32531 and TEC2016-79988-P.

Authors would like to thank the review and feedback from a number of experts in this area: Carsten Bormann, Ari Keranen, Hannes Tschofenig, Dominique Barthel, Bernie Volz and Charlie Perkins.

The text of this document was improved based on IESG Document Editing session during IETF87. Thanks to Ted Lemon and Joel Jaeglli for initiating and facilitating this editing session.

10. IANA Considerations

This document has no IANA requests.

11. Security Considerations

This document discusses the energy efficient protocol design, and does not incur any changes or challenges on security issues besides what the protocol specifications have analyzed.

12. References

12.1. Normative References

[Bluetooth42] Bluetooth Special Interest Group, "Bluetooth Core Specification Version 4.2", December 2014, <https://www.bluetooth.org/en-us/specification/adopted-specifications>.

[EN300] ETSI, "Digital Enhanced Cordless Telecommunications (DECT); Common Interface (CI)", March 2015, <https://www.etsi.org/deliver/etsi_en/300100_300199/30017501/02.06.01_60/en_30017501v020601p.pdf>.

[fifteendotfour] IEEE Computer Society, "IEEE Std. 802.15.4-2015 IEEE Standard for Local and metropolitan area networks--Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)", 2015, <https://standards.ieee.org/findstds/standard/802.15.4-2015.html>.

[G9959] International Telecommunication Union, "Short range narrow-band digital radiocommunication transceivers - PHY and MAC layer specifications, ITU-T Recommendation G.9959", January 2015, <http://www.itu.int/rec/T-REC-G.9959>.

[IEEE80211v] IEEE, "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, Amendment 8: IEEE 802.11 Wireless Network Management.", February 2012.

[MSTP] ANSI/ASHRAE, "Addenda: BACnet -- A Data Communication Protocol for Building Automation and Control Networks, ANSI/ASHRAE Addenda an, at, au, av, aw, ax, and az to ANSI/ASHRAE Standard 135-2012", July 2014, <https://www.ashrae.org/File%20Library/docLib/StdsAddenda/07-31-2014_135_2012_an_at_au_av_aw_ax_az_Final.pdf>.

[NFC] NFC Forum, "NFC Logical Link Control Protocol version 1.3, NFC Forum Technical Specification", March 2016.

[oneM2M] oneM2M, "oneM2M specifications", <http://www.onem2m.org/technical/published-documents>.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, DOI 10.17487/RFC2460, December 1998, <https://www.rfc-editor.org/info/rfc2460>.

[RFC4944] Montenegro, G., Kushalnagar, N., Hui, J., and D. Culler, "Transmission of IPv6 Packets over IEEE 802.15.4 Networks", RFC 4944, DOI 10.17487/RFC4944, September 2007, <https://www.rfc-editor.org/info/rfc4944>.

[RFC6206] Levis, P., Clausen, T., Hui, J., Gnawali, O., and J. Ko, "The Trickle Algorithm", RFC 6206, DOI 10.17487/RFC6206, March 2011, <https://www.rfc-editor.org/info/rfc6206>.

[RFC6282] Hui, J., Ed. and P. Thubert, "Compression Format for IPv6 Datagrams over IEEE 802.15.4-Based Networks", RFC 6282, DOI 10.17487/RFC6282, September 2011, <https://www.rfc-editor.org/info/rfc6282>.

[RFC6298] Paxson, V., Allman, M., Chu, J., and M. Sargent, "Computing TCP’s Retransmission Timer", RFC 6298, DOI 10.17487/RFC6298, June 2011, <https://www.rfc-editor.org/info/rfc6298>.

[RFC6550] Winter, T., Ed., Thubert, P., Ed., Brandt, A., Hui, J., Kelsey, R., Levis, P., Pister, K., Struik, R., Vasseur, JP., and R. Alexander, "RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks", RFC 6550, DOI 10.17487/RFC6550, March 2012, <https://www.rfc-editor.org/info/rfc6550>.

[RFC6551] Vasseur, JP., Ed., Kim, M., Ed., Pister, K., Dejean, N., and D. Barthel, "Routing Metrics Used for Path Calculation in Low-Power and Lossy Networks", RFC 6551, DOI 10.17487/RFC6551, March 2012, <https://www.rfc-editor.org/info/rfc6551>.

[RFC6775] Shelby, Z., Ed., Chakrabarti, S., Nordmark, E., and C. Bormann, "Neighbor Discovery Optimization for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)", RFC 6775, DOI 10.17487/RFC6775, November 2012, <https://www.rfc-editor.org/info/rfc6775>.

[RFC7228] Bormann, C., Ersue, M., and A. Keranen, "Terminology for Constrained-Node Networks", RFC 7228, DOI 10.17487/RFC7228, May 2014, <https://www.rfc-editor.org/info/rfc7228>.

[RFC7252] Shelby, Z., Hartke, K., and C. Bormann, "The Constrained Application Protocol (CoAP)", RFC 7252, DOI 10.17487/RFC7252, June 2014, <https://www.rfc-editor.org/info/rfc7252>.

[RFC7668] Nieminen, J., Savolainen, T., Isomaki, M., Patil, B., Shelby, Z., and C. Gomez, "IPv6 over BLUETOOTH(R) Low Energy", RFC 7668, DOI 10.17487/RFC7668, October 2015, <https://www.rfc-editor.org/info/rfc7668>.

[TS102] ETSI, "Digital Enhanced Cordless Telecommunications (DECT); Ultra Low Energy (ULE); Machine to Machine Communications; Part 2: Home Automation Network (phase 2", March 2015, <https://www.etsi.org/deliver/etsi_ts/102900_102999/10293902/01.01.01_60/ts_10293902v010101p.pdf>.

12.2. Informative References

[AN079] Kim, C., "Measuring Power Consumption of CC2530 With Z-Stack", September 2012, <http://www.ti.com/lit/an/swra292/swra292.pdf>.

[ContikiMAC] Dunkels, A., "The ContikiMAC Radio Duty Cycling Protocol, SICS Technical Report T2011:13", December 2011, <https://www.mysciencework.com/publication/download/2f406d3c4cc1eda32a234f7a1ad2cc3b/7eb199e4f8b00857e21af2b7d2b31c0d>.

[I-D.bormann-lwig-7228bis] Bormann, C., Ersue, M., Keranen, A., and C. Gomez, "Terminology for Constrained-Node Networks", draft-bormann-lwig-7228bis-01 (work in progress), May 2017.

[I-D.ietf-6lo-dect-ule]
Mariager, P., Petersen, J., Shelby, Z., Logt, M., and D. Barthel, "Transmission of IPv6 Packets over DECT Ultra Low Energy", draft-ietf-6lo-dect-ule-09 (work in progress), December 2016.

[I-D.ietf-6man-impatient-nud]
Nordmark, E. and I. Gashinsky, "Neighbor Unreachability Detection is too impatient", draft-ietf-6man-impatient-nud-07 (work in progress), October 2013.

[I-D.ietf-6tisch-architecture]
Thubert, P., "An Architecture for IPv6 over the TSCH mode of IEEE 802.15.4", draft-ietf-6tisch-architecture-12 (work in progress), August 2017.

[I-D.ietf-6tisch-minimal]
Vilajosana, X., Pister, K., and T. Watteyne, "Minimal 6TiSCH Configuration", draft-ietf-6tisch-minimal-21 (work in progress), February 2017.

[I-D.ietf-core-coap-pubsub]
Koster, M., Keranen, A., and J. Jimenez, "Publish-Subscribe Broker for the Constrained Application Protocol (CoAP)", draft-ietf-core-coap-pubsub-02 (work in progress), July 2017.

[I-D.ietf-core-coap-tcp-tls]
Bormann, C., Lemay, S., Tschofenig, H., Hartke, K., Silverajan, B., and B. Raymor, "CoAP (Constrained Application Protocol) over TCP, TLS, and WebSockets", draft-ietf-core-coap-tcp-tls-09 (work in progress), May 2017.

[I-D.ietf-core-resource-directory]
Shelby, Z., Koster, M., Bormann, C., Stok, P., and C. Amsuess, "CoRE Resource Directory", draft-ietf-core-resource-directory-11 (work in progress), July 2017.

[I-D.ietf-lwig-crypto-sensors]
Sethi, M., Arkko, J., Keranen, A., and H. Back, "Practical Considerations and Implementation Experiences in Securing Smart Object Networks", draft-ietf-lwig-crypto-sensors-04 (work in progress), August 2017.

[I-D.kovatsch-lwig-class1-coap]
Kovatsch, M., "Implementing CoAP for Class 1 Devices", draft-kovatsch-lwig-class1-coap-00 (work in progress), October 2012.

[I-D.rahman-core-sleepy-nodes-do-we-need]
Rahman, A., "Sleepy Devices: Do we need to Support them in CORE?", draft-rahman-core-sleepy-nodes-do-we-need-01 (work in progress), February 2014.

[Powertrace]
Dunkels, Eriksson, Finne, and Tsiftes, "Powertrace: Network-level Power Profiling for Low-power Wireless Networks", March 2011, <https://core.ac.uk/download/pdf/11435067.pdf?repositoryId=362>.

Authors' Addresses

Carles Gomez
Universitat Politecnica de Catalunya
C/Esteve Terradas, 7
Castelldefels 08860
Spain

Email: carlesgo@entel.upc.edu

Matthias Kovatsch
ETH Zurich
Universitaetstrasse 6
Zurich, CH-8092
Switzerland

Email: kovatsch@inf.ethz.ch

Hui Tian
China Academy of Telecommunication Research
Huayuanbeilu No.52
Beijing, Haidian District 100191
China

Email: tianhui@ritt.cn

Zhen Cao (editor)
Huawei Technologies
China

Email: zhencao.ietf@gmail.com

6lo
Internet-Draft
Intended status: Standards Track
Expires: March 29, 2015

G. Rizzo, Ed.
AJ. Jara, Ed.
A. Olivieri
Y. Bocchi
HES-SO
MR. Palattella
SnT/Univ. of Luxembourg
L. Ladid
SnT/Univ. of Luxembourg/IPv6 Forum
S. Ziegler
C. Crettaz
Mandat International
September 25, 2014

IPv6 mapping to non-IP protocols
draft-rizzo-6lo-6legacy-02

Abstract

IPv6 is an important enabler of the Internet of Things, since it provides an addressing space large enough to encompass a vast and ubiquitous set of sensors and devices, allowing them to interconnect and interact seamlessly. To date, an important fraction of those devices is based on networking technologies other than IP. An important problem to solve in order to include them into an IPv6-based Internet of Things, is to define a mechanism for assigning an IPv6 address to each of them, in a way which avoids conflicts and protocol aliasing.

The only existing proposal for such a mapping leaves many problems unsolved and it is nowadays inadequate to cope with the new scenarios which the Internet of Things presents. This document defines a mechanism, 6TONon-IP, for assigning automatically an IPv6 address to devices which do not support IPv6 or IPv4, in a way which minimizes the chances of address conflicts, and of frequent configuration changes due to instability of connection among devices. Such a mapping mechanism enables stateless autoconfiguration for legacy technology devices, allowing them to interconnect through the Internet and to fully integrate into a world wide scale, IPv6-based IoT.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

This Internet-Draft will expire on March 29, 2015.

Table of Contents

1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2. Introduction

The Future Internet and the IPv6 protocol enable a new generation of techniques for accessing the network, which extend the Internet seamlessly to personal devices, sensors, home appliances, enabling the so called ''Internet of Things'' (IoT). One of the key issues which presently hampers the development of IoT and limits its potential is the lack of an efficient common framework for the integration among the vast and diverse set of protocols and technologies which compose it. Current sensors and their application environments employ a large set of technologies which lack efficient interoperability. Some associations of manufacturers have been formed to build a common technological framework in specific application domains, e.g. KNX for building automation (http://www.knx.org/), ZigBee (ZigBee Alliance) (http://www.zigbee.org/), and protocols such as X10 and CAN. Such frameworks are based on very different architectures, and the protocols which compose them are generally not interoperable. Finally, most of these technologies were designed in a context of small and local networks, with limited capabilities, and they were not conceived for integration within the Internet. One of the ideas at the basis of the IoT is the constitution of a common set of protocols which enables the interaction between devices through the Internet. By enabling interaction through the Internet, new services could be conceived and implemented, increasing the value produced by the IoT infrastructure. The adoption of a common framework may make more economically convenient its deployment, and foster the development of new smart environments (buildings, cities, etc), ultimately making possible the full realization of the potential of the IoT. As deployment of new sensors is typically expensive, it is unthinkable of putting to disuse an installed set of sensors, once a new set of devices (typically, IPv6 enabled) is deployed. This is not an uncommon case, as the set of deployed legacy devices (sensors, actuators) is to date very large. Rather, mechanisms are needed to integrate legacy devices into a common IoT platform, in order to include them in all the present and future services (e.g. devices and services directory, localization services, etc) which will be implemented on the IoT. For these reasons, many designers of the Internet of Things are focusing on building such common access and communications framework. All the proposals (e.g. CoAP, RESTful Web services) presently under discussion are based on IPv6. This has important implications on the addressing of the devices. Indeed a

common addressing at the device level is mandatory, in order to implement true Machine to Machine (M2M) communications without Portal Servers, which would make the whole system difficult to integrate and scale. The present document focuses on the network layer aspects of such IPv6 based integration. At the network layer, a mechanism which assigns an IPv6 address to each device is needed, to solve the addressing problem. In this document, we propose a new mechanism for the users and devices to map the different addressing spaces to a common IPv6 one. Our proposed mechanism solves several issues posed by some of the mappings adopted so far. Such mapping makes it possible for every device from each technology to operate through a common framework based on IPv6 and protocols over IPv6 such as RESTFul WebServices and Constrained Application Protocol (CoAP). For each technology, the proposed mechanism maps technology-specific features to a set of fields defined within the IPv6 address. This allows the location and identification of the devices in a multi-protocol card, or in any gateway or Portal Server.

2.1. Examples

In this subsection, we present two examples which help understanding the importance of adopting a common IPv6 based framework for interaction between things, and the need for legacy devices to be individually addressable through IPv6.

2.1.1. Example 1 - Building automation systems and IoT

The IoT is composed by a very large set of devices, which is poised to grow exponentially in the near future. For this reason, a directory service is needed, which offers the possibility to individuate a specific device or set of devices, with given capabilities or within a given geographical region. Let us assume such directory lists devices with their IPv6 addresses, and their function (say a temperature sensor, or a mobile phone, etc). For instance, let us consider the case of someone willing to build a map of temperatures in a given geographical region. Such directory service would allow retrieving the list of available devices within that region, each with its own IPv6 address. Assume some of those devices are legacy, non IP based temperature sensors and part of a given building automation system. Assume also that such system manages several of those temperature sensors. Even if such system would be reachable via IP, without having those sensors individually listed in the directory and appearing as ''autonomous'' things, which can be polled directly, one should resort to techniques for retrieving the temperature reading of those sensors which are specific of that building automation technology. This would make more complex the implementation of such a temperature map.

Instead, by having the building automation system expose each sensor as an IPv6 enabled device, the whole set of temperature sensors would be accessible in a homogeneous way, greatly simplifying the task.

2.1.2. Example 2 - KNX and demand-side management

KNX is a standardized (EN 50090, ISO/IEC 14543), OSI-based network communications protocol for intelligent buildings. Among the devices typically managed through KNX, we find:

- Lighting control systems;
- Heating/ventilation and air conditioning devices;
- Shutter/blind and shading control systems; and
- Energy management and electricity/gas/water metering devices.

KNX devices do not support IP. Therefore, in order to connect a KNX home network to the Internet, a gateway (KNXnet/IP router) is necessary. Other technologies for home automation are available nowadays, in which each smart device (air conditioners, washing machines, etc) supports IPv6. Let us consider a scenario in which an utility company offers an agreement to a fraction of its clients. In exchange for a cut on the energy bill, the utility company gains direct control over some appliances at the premises of the client. In this way, by powering off some of those devices in periods when the production cost of power are very high, the utility company realizes potentially high savings.

In order to implement this, the utility company sends commands to a set of devices under its direct control. For recently installed devices, the utility can assume that they support IPv6, and some application layer protocols such as CoAP. Therefore a command to switch off a device would use the IPv6 address to identify the device, and the application layer protocol to send the actual command. But for KNX devices, the command should have another format: the IPv6 address should be the one of the router bridging the IPv6 and the KNX networks, and upper layers protocols should take care of identifying the specific device inside the KNX home network to whom the command should be sent. Having to format a specific query for each specific home automation protocol adds a level of complexity which translates into higher costs of implementation and maintenance of such a service.

3. Reference System

In this section we describe a reference system where the IPv6 mapping is used. Such a system includes:

1. A set of networks running non-IPv6-compatible technologies, each with one or more hosts connected. Such networks generally use different OSI layer 3 protocols, or they may adopt a technology which does not have any layer 3 protocol.

2. A proxy, which hosts the IPv6 mapping functionality. Such device is typically connected to each of the legacy protocols networks, and it accesses the Internet via the IPv6 protocol. Such IPv6 addressing proxy performs all the necessary conversions and adaptations between IPv6 and the (local) networking protocol of the legacy technologies, in a way which depends on the specific legacy technology considered. This proxy makes use of the IPv6 mapping mechanism in order to transforms the native addressing to IPv6 Host ID and vice versa in a way that depends on the legacy technology.

Though in what follows we will describe the proposed mapping with reference to such a system, the main ideas behind it are more general, and they apply to settings others than the one of reference presented here.

4. Issues addressed through the 6TONon-IP mapping mechanism

In this section we highlight the main open issues regarding assignment of IPv6 addresses to devices which do not support IPv6 or IPv4, and we describe a set of desirable properties for a mechanism for automatic assignment of IPv6 addresses to such devices, which we name henceforth 6TONon-IP. In Appendix A of RFC 4291, a method is described for creating modified EUI-64 format Interface Identifiers out of links or nodes with IEEE EUI-64 Identifiers, or with IEEE 802 48-bit MACs. Moreover, for technologies having other link layer interface identifier, some possible mapping methods are sketched, leaving for each legacy protocol the possibility to define its own mapping method.

In the present document, we propose a mapping mechanism which enables stateless address autoconfiguration for legacy technologies, and which exploits some protocol specific identifier such as link layer interface identifiers, and the like. The proposed mapping mechanism addresses the following issues:

1. Protocol identification: For the legacy protocols to which the mapping described in RFC 4291 does not apply, a mechanism is

needed to map an IPv6 address to the right legacy protocol. This feature is necessary in case of devices which operate as proxy for more than one legacy technology at the same time.

2. Inter protocol aliasing: Without a mechanism for identifying the legacy protocol from the host part of the IPv6 address, address conflicts are possible among devices belonging to different legacy protocols. For instance, this may happen when the link layer interface identifier is the same for two devices belonging to different technologies. As several legacy technologies are characterized by a small addressing space, address conflicts are not so unlikely.

3. Conflicts between IPv6 mapped legacy technology addresses and addresses derived from (modified or not) EUI-64 format interface identifiers.

4. Intra-protocol aliasing: As several legacy technologies are characterized by a small addressing space, it is not unlikely to have two legacy devices, mapped to IPv6 addresses with the same network ID (for instance, in the case in which they belong to two separate networks of the same technology, both connected to a same proxy), and with a same interface identifier, and mapping therefore to a same IPv6 address.

Moreover, the following is a list of desirable properties for a 6TONon-IP mapping:

1. Consistency: A host should get the same IPv6 address every time it connects to a same legacy network, assuming that the configuration of all the other devices in that network remains unchanged. This allows avoiding to advertise a new address every time the host reconnects. This feature might be particularly important for devices which are not always "on", or which are not permanently connected.

2. Local Uniqueness: For devices which have an IPv6 address with a same network part, the host part should be unique for each host. This property allows avoiding address conflicts.

3. Uniqueness within the whole Internet: Coherently with the IoT vision, the host part of an IPv6 address associated to a host should be unique within the whole Internet.

Depending on the specific legacy protocol, there might be protocol specific limitations to the satisfaction of these properties. In particular, for those protocols which do not have an interface identifier which is unique, properties 1) and 2) cannot be fully

satisfied. Indeed, no mapping can solve address conflicts which take place inside a legacy protocol network. When legacy protocols have a interface identifier which is unique, this can be used to produce a unique host part of an IPv6 address, and its uniqueness would guarantee the satisfaction of properties 1), 2) and 3).

5. 6TONon-IP Mapping Method

In this section we describe the proposed strategy for forming IPv6 addresses from legacy protocol information, and the address format that derives from it. We assume that (one or more) 64 bits Network ID prefixes are given to the mapping function, which therefore computes the 64 bits of the Host ID part of the address (IPv6 interface identifier), in order to form a full IPv6 address.

The input of the proposed mapping function consists in the interface identifier of the legacy protocol.

In the proposed mapping method, the resulting Host ID part (IPv6 interface identifier) is composed by six fields, as shown in Figure 1:

- A Technology ID field (11 bits), containing a code which identifies the specific legacy protocol. This field is split into two parts, one of 6 bits, and another of 5 bits.

- U/L bit (1 bit), in order to keep compatibility with the mapping EUI-64 [RFC4291]. The U/L bit is the seventh bit of the first byte and is used to determine whether the address is universally or locally administered. This bit is set to "0", in order to indicate local scope, analogously to what proposed in [RFC4291]. This choice prevents address conflicts with IPv6 interface identifier generated from IEEE EUI-64 identifiers or IEEE 48-bit MAC identifiers.

- A Reserved field (4 bits). This field can be used in the future for the identification of different interfaces for a same technology (in the same subnetwork).

- Technology Mapping field (32 bits), which maps the interface identifier of the legacy protocol. For those protocols for which the IID is not larger than 32 bits, this field contains the 32 bits of the IID. For IID which are larger than 32 bits, a hashing function is used instead of direct mapping. In particular, some hashing algorithms such as CRC-32 are suggested. Hashing satisfies the requirements of consistency and uniqueness within a subnet with a very high probability, which depends on the hashing

algorithm used. This field is split into two parts, one of 8 bits, and another of 24 bits.

- The fourth and fifth bytes are both set to to "0x00", in order not to conflict with EUI-64 interface identifiers.

The resulting format of the Host ID part of the IPv6 address obtained from the mapping is indicated in Figure 1.

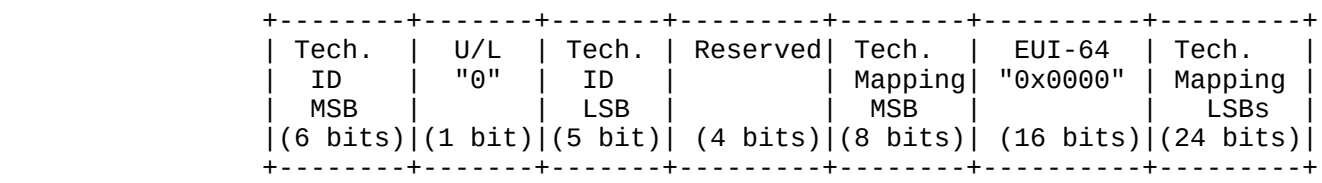

Figure 1: general format of the host ID part for legacy protocols

6. Examples

In this section we illustrate the proposed mapping method by applying it on some examples.

6.1. Example 1 - EIB/KNX

We assume the legacy protocol is EIB/KNX. This device has two kind of addresses: On the one hand, a logical address for management of group operations, and on the other hand, an individual address for identification of the device in the topology.

The mapping will be focused for the individual address. This includes an Area ID (4 bits), Line ID (8 bits), and Device ID (8 Bits). An example, is the value 0x1/0x01/0x01 for a sensor connected in the Area ID 0x1, Line ID 0x01, and Device ID 0x01.

We apply a hash (CRC-32) to the sequence 0x10101. The result is 0xDEA258A5.

Let us assume that EIB/Konnex Technology ID is "0". Thereby, the IPv6 interface identifier is "0000:DE00:00A2:58A5", considering the documentation network 2001:db8::/32. The final IPv6 address for the legacy device is "2001:db8::DE00:A2:58A5".

The address is presented in the Figure 2.

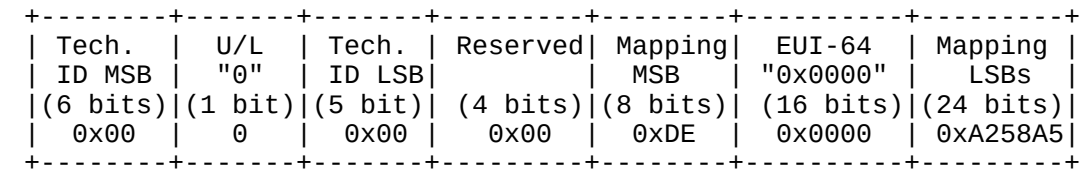

Figure 2: EIB/KNX example: the IPv6 interface identifier.

6.2. Example 2 - RFID

We assume the legacy protocol is RFID. Each RFID device is identified by its Electronic Product Code (EPC), whose length may vary from 96 to 256 bits. Let us assume to have an RFID device whose EPC is given by 01.23F3D00.8666A3.000000A05 (12 bytes). Let us assume that the RFID technology ID is "1".

We apply a hash (CRC-32) to the sequence 0x0123F3D008666A3000000A05. The result is 0xA93AFFA0.

Thereby, the IPv6 interface identifier is "0004:A900:003A:FFA0", considering the documentation network 2001:db8::/32. The final IPv6 address for the RFID tag is "2001:db8::400:A900:3A:FFA0".

The address is presented in the Figure 2.

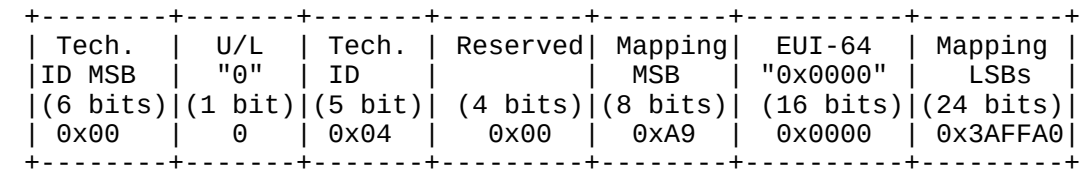

Figure 3: RFID example: the IPv6 interface identifier.

7. IANA Considerations

Not yet defined.

8. Security considerations

The proposed mapping mechanism, being based on mapping proprietary protocol ID, results in such ID being incorporated in the final IPv6 address, exposing this piece of information to the Internet. The concern has been that a user might not want to expose the details of the system to outsiders. For such concern, which holds also for MAC address mapping into EUI64 addresses, please refer to appendix B in [RFC4942].

9. Acknowledgements

The authors wish to acknowledge the following for their review and constructive criticism of this proposal: Robert Cragie. Thanks to the IoT6 European Project (STREP) of the 7th Framework Program (Grant 288445), and the colleagues who have collaborated in this work. In particular, Antonio Skarmeta from the University of Murcia, Peter Kirstein and Socrates Varakliotis from the University Colleague London, and Sebastien Ziegler from Mandat International.

10. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, February 2006.

[SENSORS] Jara, A., Moreno-Sanchez, P., Skarmeta, A., Varakliotis, S., and P. Kirstein,, "IPv6 Addressing Proxy: Mapping Native Addressing from Legacy Technologies and Devices to the Internet of Things (IPv6)", Sensors 13, no. 5, 6687-6712, 2013, 2013.

Authors' Addresses

Gianluca Rizzo, Ed.
HES-SO Valais
Technopole 3
Sierre, Valais 3960
Switzerland

Phone: +41-76-6151758
Email: gianluca.rizzo@hevs.ch

Antonio J. Jara, Ed.
HES-SO Valais
Technopole 3
Sierre, Valais 3960
Switzerland

Email: jara@ieee.org

Alex C. Olivieri
HES-SO Valais
Technopole 3
Sierre, Valais 3960
Switzerland

Email: Alex.Olivieri@hevs.ch

Yann Bocchi
HES-SO Valais
Technopole 3
Sierre, Valais 3960
Switzerland

Email: yann.bocchi@hevs.ch

Maria Rita Palattella
University of Luxembourg
4, rue Alphonse Weicker
Interdisciplinary Centre for Security, Reliability and Trust
Luxembourg

Phone: (+352) 46 66 44 5841
Email: maria-rita.palattella@uni.lu

Latif Ladid
University of Luxembourg / IPv6 Forum
4, rue Alphonse Weicker
Interdisciplinary Centre for Security, Reliability and Trust
Luxembourg

Phone: (+352) 46 66 44 5720
Email: latif@ladid.lu

Sebastien Ziegler
Mandat International
3 rue Champ Baron
1209 Geneva
Switzerland

Email: sziegler@mandint.org

Cedric Crettaz
Mandat International
3 rue Champ Baron
1209 Geneva
Switzerland

Email: iot6@mandint.org