A Survey On Embedded Open Source System Software For The Internet Of Things

Mahdi Amiri-Kordestani[1]
[1] Communication and Information Research Center,
Sultan Qaboos University,
Muscat, Oman
amiri@squ.edu.om

Hadj Bourdoucen[1,2]
[1] Communication and Information Research Center,
[2] College of Engineering, ECE Dept.
Sultan Qaboos University,
Muscat, Oman, hadj@squ.edu.om

***Abstract*— Internet of Things (IoT) is an opportunity for future devices to be smarter, more robust and efficient. This opening has become available due to the continuous cost reductions of many individual systems and components such as sensors, computing devices, communications methods, the cloud and the big data paradigms. Open source software, hardware, and standards are considered as further cost-saving ways, and thus, a significant number of current products are already using open source projects as their principal elements to reduce the development time, minimize the cost, and develop more competitive products.**

However, choosing the right components for the open source system depends on multiple factors, such as the open source licensing, application complexity, required performance, hardware architecture, communication standards, external sensors, peripherals, and other parameters. This paper studies a number of the state-of-the-art open source system software projects and frameworks and identifies some of the key parameters that should be considered to select and adapt for an open source project. Moreover, a comparison of some of the favorite software, hardware and standards that target the Internet of Things is represented.

Keywords—Internet of Things; System Software; Embedded Operating System; Open Source

I. INTRODUCTION

Nearly all the leading technology and business forecasts agree that Internet of Things (IoT) is going to be a huge market and will cause a significant impact on our lives [1]–[3]. IoT is a combination of many components with different scopes that includes a variety of sensors, actuators, embedded computing systems, communications techniques, as well as various protocols, frameworks, middleware, and software systems. The IoT is frequently associated with cloud computing and the big data techniques as the primary data end points.

Open source software as specified by the Free Software Foundation (FSF) [4], allows the user to run, share, study, and modify the software. In other words, the term free here means freedom in the following abilities:

- Use the software for any purpose.
- Copy and distribute the software to other people.
- Study the algorithm and learn the technology
- Develop and improve the software.

From the hardware aspect, "Open Source Hardware Association" (OSHWA) [5] describes the open source hardware as the hardware with a publicly available design and very similar to open source software so as, anyone can examine, change, distribute, produce, and market devices based on that design. Open source hardware also provides the freedom to utilize and advance the device while sharing experiences.

There are hundreds of open source projects about the IoT and currently looking at github.com with phrase "Internet of Things" represents more than 1,800 projects. Major IoT projects and frameworks usually address a typical ecosystem that lets IoT devices communicate to the Internet, and usually to a cloud system that can store and process the data and generate advanced reports and controlling signals [6].

Based on mentioned pattern, a considerable number of open source projects are available that address one or more of the following concepts:

1. Embedded system software for IoT devices.
2. Network protocols such as Bluetooth LE, ZigBee, Thread, and 6LoWPAN techniques.
3. Application layer protocols such as CoAP and MQTT.
4. Security and privacy mechanism such as M2M authentication and cryptography (AES, SSL/TLS).
5. Higher level management, control, and reporting.
6. Application and firmware upgrade and patch management.

II. IOT APPLICATION SPECTRUM

The "Internet of Things" is a comprehensive model of all kinds of computing devices, (also referred to "*things*," "*embedded devices*," and "*smart devices*") that are somehow connected to the Internet. In theory, a device can be attached to almost any object such as vehicles, home appliances, industrial mechanical or electrical machines, and even a person to let the object to communicate to the Internet.

Some major IoT applications [7] are buildings and home automation, smart cities, smart industry or manufacturing, wearables and healthcare devices, as well as automotive. TABLE 1, indicates a number of major IoT fields, a typical application, and some average communication distance ranges.

TABLE 1. TYPICAL IOT APPLICATIONS AND THEIR RANGE

Field	Application	Range
Wearables	Healthcare, WBAN	<2m
Automotive	Infotainment	<5m
Smart homes	Intelligent buildings	<100m
Smart industries	Smart control	<10 Km
Smart cities	Smart traffic control	<100 Km
Smart grid	Power distribution	100+ Km

A major market for IoT applications is the home appliances that make smart home [8] to be the most popular IoT application among other fields. The smart home includes many available commercial products such home automation systems, home monitoring, and security systems as well as academic research titles such as intelligent homes and buildings to reduce the overall consumed energy.

TABLE 2. COMPARISON BETWEEN SOME MAJOR IOT CONNECTIVITY STANDARDS

Standard	Native IP	2.4 GHz	Mesh	Approximate Range	Data Rate	Example Application
Wifi 802.11 b/g/n	Yes	Yes	No [a]	35-250 m	600 Mbps (802.11n)	Home, Industry
Bluetooth Smart (LE)	No	Yes	No	50-100 m	1 Mbps	Home, Wearables
ANT+	No	Yes	Yes	30 m	12.8/60 Kbps	Home, Wearables
ZigBee (802.15.4)	No	Yes [b]	Yes	20 m	20 to 250 kbps	Home, Industry
Thread (802.15.4)	Yes	Yes	Yes	20 m	250 kbps	Home, Industry
Z-Wave	No	No [c]	Yes	30 m	9.6/40/100 Kbps	Home, Industry
Sub-Ghz	n/a [c]	No	n/a [c]	100+ Km	n/a [c]	Smart City, Grid
Cellular	Yes (4G)	No	No	35 Km (GSM), 200 Km (HSPA)	50-100 Mbps (LTE)	Smart City, Grid

a. 802.11s also adds the support for mesh networks.

b. ZigBee can operate at 2.4 GHz as well as 868 MHz in EU and 915 MHz in US.

c. Z-Wave operates at 868.42 MHz in EU and 908.42 MHz in US, other countries vary.

d. Sub-Ghz specification depends on the implementation.

Wearable [9] and healthcare devices are the second most popular application of the IoT. A wearable is a small computing device running on a battery that usually includes some microelectromechanical sensors (MEMS) to monitor vital signs such as motions, temperature, heart rate, blood pressure, and others. Some typical wearable applications are for entertainment, fitness, smart watches, as well as healthcare and monitoring devices. As an example, a healthcare device can monitor vital signs and transfer the captured data to an upper system such on the Internet (probably a cloud system) through network gateways or other communication mechanisms such as cellular networks. The health monitoring device and the management system can alert both the person as well as corresponding doctors in case of difficulties and emergencies.

The solutions such as smart cities [8] expand the IoT domain to a wider area. Some major topics are about greater management of traffics, water, waste, and the city security. Some specialized topics such as smart grids [10] expand the IoT to a much wider and to a country or state. The idea behind smart grids is to monitor both sides of electricity consumers and suppliers to optimize the power distribution and increase the efficiency and safety.

IoT applications in industry such as smart industry, smart manufacturing or industrial Internet [7], introduces the combination of electrical and mechanical machines, advanced data analytics and people at work to increase the overall performance and productions. The combination of devices with network sensors and artificial intelligence can better serve the industries to gain productions.

Another trending application for IoT is automotive industry [7]. Some common patterns are vehicle to vehicle (V2V) communications, vehicle to roadside communications, and the network inside vehicles that links all sensors, cameras, the navigation system, and other embedded devices to communicate and collaborate with each other.

III. OPEN SOURCE IOT SOFTWARE

As discussed above, the IoT applications have a broad range of domains, and they incorporate many network standards as represented in TABLE 2, with different hardware architectures as illustrated in TABLE 3 and consequently, there are many system software and projects for various needs. The system software which is also called operating system for larger computing devices and firmware for smaller embedded systems is usually a type of program that is designed to run on hardware directly. A number of open source system software is discussed here, and a comparison is shown in TABLE 4.

A. *Collaborative projects*

Many important open source projects follow the type of collaborative projects and are governed by a well-known organization such as the Linux Foundation and a list of corporate members from various industries. Linux Foundation as a key organization in open source is a non-profit organization with an aim to promote, support and improve Linux and some key open source projects. Currently, there are more than 50 projects listed on the organization’s website, and some of those projects are directly targeting the IoT domain such as IoTivity, Tizen, Automotive Grade Linux, Yocto, and Zephyr.

IoTivity [11] is an open source framework of the connectivity standards for IoT that was originally developed by Intel to allow devices to communicate to the Internet and to each other. Another very similar open source project was the AllJoyn framework by Allseen Alliance which is currently merging with IoTivity in a unified framework. Organizations such as Linux Foundation, Open Connectivity Foundation (OCF), AllSeen Alliance and many major companies such as Intel, Microsoft, Samsung, LG, Qualcomm, Mediatek, CISCO, GE, Canon, Sharp and others are all involved in this huge project.

Tizen [12] is a Linux based distribution and a competitor to both Google’s Android and Apple’s iOS that can address a broad range of embedded devices, including smartphones, tablets, TVs, players, cameras, printers, home appliances, and wearables. Samsung is the major player of Tizen project and has merged its Bada project into Tizen. The other leading members from silicon and telecom industries are Intel, NEC, Fujitsu, Huawei, Panasonic, Sprint, Vodafone and others.

Automotive Grade Linux (AGL) [13] is an embedded Linux project to create open source applications and solutions with members from the automotive, telecom, and silicon industries. The primary purpose for AGL is In-Vehicle-Infotainment (IVI) systems, but further applications such as digital instrument panels and telematic systems are also planned to be added. There is also an AGL Linux distribution that collects some of the software that has been originally developed for other platforms such as Tizen IVI and GENIVI in a Linux distribution.

The Yocto [14] is an embedded Linux project that provides a framework to create a highly customized Linux distribution. Some well-known embedded and commercial grade projects such as Mentor Embedded Linux and Wind River Linux are based on specific builds of Yocto Linux project.

Zephyr [15] is somehow a very different developing project from Linux Foundation that provides an embedded and real-time operating system (RTOS) for microcontrollers with concentrated resources that cannot use the standard Linux kernel or they require specialized features such as real-time or safety specifications. Intel (Wind River Systems) originally developed Rocket kernel for embedded and IoT devices. Later, Intel donated the Rocket kernel to Zephyr project to cooperate with other members. Zephyr source code as illustrated in TABLE 4 has a more permissive license compared to other open source RTOS projects such as FreeRTOS as well as the original Linux kernel source code.

TABLE 3. ARM ARCHITECTURES AND EXAMPLE OF IOT APPLICATIONS.

Architecture	Bits	ARM Profile	Suitable IoT devices	Common cores	New cores
ARMv8-A	32/64	Application	Ultra-high-performance, full OS	Cortex-A53, A57, A72	Cortex-A35, A73
ARMv8-A	32	Application	High-performance, full OS	New	Cortex-A32
ARMv8-R	32	Real-time	Real-time, High-performance, RTOS	New	Cortex-R52
ARMv8-M	32	Controller	High-performance, RTOS	New	Cortex-M23, M33
ARMv7-A	32	Application	High-performance, full OS	Cortex-A5, A7, A8, A9, A15	Cortex-A17
ARMv7-R	32	Real-time	Real-time, RTOS	Cortex-R4, R5, R7	Cortex-R8
ARMv7E-M	32	Controller	Low-power, RTOS plus DSP	Cortex-M4, M7	n/a
ARMv7-M	32	Controller	Low-power, RTOS	Cortex-M3	n/a
ARMv6-M	32	Controller	Ultra Low-power, RTOS	Cortex-M0, M0+, M1	n/a

(Full OS stands for full feature operating systems such as Linux, RTOS stands for bare-metal applications or real-time operating systems)

B. Android ecosystem

Google is a key company regarding the IoT field because it is directing the Android software as the base software for many smartphones. Meanwhile, it has the infrastructure and the required experiences for running big data models and cloud services. Currently, Google is offering two products that target the IoT, the Weave [16] as a communications platform and Android Things (formerly Brillo) [17] as an Android-based operating system as well as necessary frameworks for high-end products very similar to Android operating system for smartphones.

Google Weave is a competitor of IoTivity framework and provides a communication and controlling mechanism based on HTTPS for controllable devices such as lights and wall switches. Weave has two components; "*Weave Device SDK*" and "*Weave Server*." The device SDK is an open source library (libiota) written in C language for maximum portability and can be integrated with IoT devices to let them communicate with the Weave servers. Currently, the SDK can only support HVAC controller, light, outlet, television, and wall switch as IoT device types, and there is a promise to add other kinds to this project. The other component of the Weave platform is the server side "*Weave Server*" that presents secure methods to store the states, device registration mechanism, and command broadcasting. The server is already integrated with other Google cloud services.

Android Things follows Android Open Source Project (AOSP) patterns and software that is designed to support high-end IoT devices, sensors, and wireless networking while claiming much lower amounts of storage, memory, and power compared to AOSP. The Android Things also has Android ecosystem and utilizes the existing development tools, APIs, and resources that also provide low-level I/O controls and required libraries for some sensors, display controllers, and embedded modules.

C. ARM ecosystem

ARM ecosystem is mostly focused on the system software for low-cost and low-power Cortex-M processors. There are many types of hardware devices, and hence there are also many different system software products. For instance, in ARM Cortex processor family, there are three profiles of processors to address various applications as ARM Cortex-A, Cortex-R, and Cortex-M.

ARM Cortex-A family as represented in TABLE 3, is a full-featured application processor with many components such as memory management unit (MMU) and is usually used in high-end devices such as smartphones, tablets, and netbooks. Cortex-A can handle large applications and full feature operating systems such as Android, Linux, and Windows. ARM Cortex-R is designed to be a deterministic processor and to address real-time and functional safety applications. The Cortex-M is the smallest member of the family with much lower cost, performance and power consumption that is more suitable for many IoT devices. As an example, an ultra-low power ARM Cortex-M0+ can run off a battery for years [18], and is still powerful enough for many IoT devices.

As discussed earlier, Android Things (Brillo) is suitable for high-performance processors such as ARM Cortex-A, and x86 architectures. Meanwhile, the Zephyr project from

Linux Foundation is designed for low-performance controllers such as Cortex-M that lack features such as MMU.

mbed is the official ARM solution designed to address the embedded devices and the Internet of Things. The "*mbed OS*" and ARM's upcoming product "*mbed Cloud*" are the two main components of this project [19]. mbed OS is an open source system software (Apache 2.0 license) that targets Cortex-M microcontrollers, and incorporates a tiny RTOS based on CMSIS-RTOS RTX, security features (TLS), many connectivity standards, and device drivers for popular sensors in a modular form. mbed OS supports an extensive range of communication standards such as Bluetooth LE, 6LoWPAN, Ethernet, WiFi, RFID, Cellular, Thread, and LoRa LPWAN. mbed OS is designed to be compatible with upcoming mbed Cloud product from ARM that will add some cloud features such as mbed Cloud Provision, Connect, and Update services to mbed ecosystem.

D. Community system software projects

Other than mentioned extensive ecosystems, many individual projects have turned into key components for IoT devices. Contiki [20] is a well-known system software for both academic research and industrial usage that initially is designed to target the low-power and low-cost microcontrollers such as TI MSP430x, Atmel AVR, and newer TI SimpleLink family. Contiki and its native simulator Cooja, have been used as research vehicles for many wireless sensor network projects. Contiki is best known for the complete and efficient implementation of essential network standards such as IP, UDP, TCP, as well as low-power standards such as 6LoWPAN, RPL, and CoAP.

RIOT [21], [22] is another open source (LGPL) and partially POSIX compatible system software with real-time scheduling support that targets a broad range of 8-bit, 16-bit and 32-bit microcontrollers. RIOT is designed to be energy-efficient and modular with very low memory requirements. RIOT was initially developed as a research project by FU Berlin, INRIA, and HAW Hamburg and follows the microkernel type, somehow inherited from FireKernel [23] project that was targeting wireless sensor networks. RIOT supports the essential features such as memory allocations and multi-threading for C and C++ programming languages as an advantage. RIOT already includes some of common networking protocols and standards such as IPv6, 6LoWPAN, UDP, RPL, CoAP, and CCN-Lite.

Apache Mynewt [24] is a new developing system software project and is a real-time and has a modular and open-source architecture designed for low-performance IoT devices. Mynewt targets ARM Cortex M0 to 4 and RISC-V architectures with a plan to extend the hardware support to MIPS architecture. Mynewt targets low-cost devices that have to operate for an extended period with limited power, memory, and flash storages. It is a collaborative open-source project with Apache License 2.0 governed by the Apache Software Foundation. Mynewt includes NimBLE [25] as the first fully open source implementation of Bluetooth Low Energy (BLE) 4.2 as an advantage to the other projects.

There are also many other Linux-based embedded distributions such as Ubuntu Snappy Core, and Ostro that try to address high-performance devices. These are not covered in this paper.

TABLE 4. COMPARISON BETWEEN ARM MBED, ANDROID THINGS, AND SOME COMMUNITY PROJECT.

System Software	Provider	License	Example Hardware	Integrated Cloud
Android Things	Google	Apache 2.0/GPLv2	High-end processors such as x86, ARM Cortex-A (32/64)	Google Weave
mbed OS	ARM	Apache 2.0	ARM Cortex-M	mbed Cloud
Contiki OS	Community; Thingsquare	3-clause BSD	ARM Cortex-M, MSP430, AVR, x86	Thingsquare
RIOT OS	Community; FU Berlin	LGPLv2.1	ARM Cortex-M, MSP430, ARM7, AVR, x86	n/a
Zephyr	Collaborative by Linux Foundation	Apache 2.0	ARM Cortex-M, x86, ARC, NIOS2	n/a
Apache Mynewt	Collaborative by Apache Foundation	Apache 2.0	Cortex-M, RISC-V	n/a

IV. RESEARCH OPPORTUNITIES

Communication standards, security, and privacy are a number of significant research areas for IoT devices [2], [26], [27] because the connected things are not always general purpose computers, but a variety of embedded devices that can communicate to each other and to the Internet. For instance, an insecure device can compromise and allow unauthorized users to gain necessary access to perform the monitoring or controlling the device or a system and collect private data or disturb the owner.

Security risks for IoT devices can be caused by different reasons [27] such as the lack of experience regarding security concerns, the absence of aftersales bug fixes or updates, problems during software upgrades, constrained hardware interfaces and resources, and even backdoors and malware created during the development or manufacturing.

Recent research on some IoT devices [28] reports that 90% of embedded devices including their ecosystem software collect some personal information and 70% of them communicate through unencrypted network services. About, 60% of the system software of investigated devices could be upgraded through the network without using any data encryption or protection mechanisms. Moreover, some update images (firmware and patches) where standard Linux file system images could easily be mounted and altered by ordinary users.

V. Conclusions

The Internet of Things will bring shortly many embedded smart things to our life. Many open source components such as system software projects can be employed to accelerate the development speed and reduce the overall cost of future devices.

Some primary concerns in choosing an open source system software other than technical specifications (such as support for hardware, functionality, reliability, performance, network connectivity and standards) are the type of software license, project governance, founding members, commercial and community supports, and finally the size of users.

A number of major ecosystems have been introduced and studied in this paper. Among all those projects, the IoTivity from Linux Foundation has an active and massive community from reputable companies with a permissive license without commercial components.

Google Android Things in combination with Android operating system, Google cloud infrastructure, and other cloud services is also a turnkey solution that will empower lots of high-performance devices in the near future but will lock solution providers to Google ecosystem.

The other ecosystem is ARM mbed OS in combination with mbed Cloud as a chip to the cloud solution that can target low-performance MCU devices but will also lock solution providers to mbed Cloud ecosystem.

Contiki, Zephyr, Mynewt, and RIOT introduced in this paper (summarized in TABLE 4) are open source system software without integrated commercial cloud services that can be utilized as essential components for larger ecosystems systems.

Contiki and RIOT are both mature system software projects for IoT devices. Contiki has an extensive networking stack and a strong community that is supported by a number of large companies. There is also an integrated commercial cloud solution from the Thingsquare that extends and connects this project to the cloud. It is worth noting that RIOT better supports high-level languages such as C and C++ and has a better memory management with a multi-thread support but offers a restricted "LGPLv2.1" software license compared to the "3-clause BSD" license used by Contiki OS.

Zephyr and Mynewt are two emerging collaborative projects from Linux Foundation and Apache Foundation with Apache 2.0 as a permissive open source license. Both projects are quickly growing by adding hardware supports, software, and connectivity features and they will capture more interest in the future from IoT vendors.

Acknowledgment

This work is part of the research that is conducted at the "*Free and Open Source Software Laboratory*" at Communication and Information Research Center (CIRC), Sultan Qaboos University, Muscat, Oman.

References

[1] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Futur. Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, 2013.

[2] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[3] J. Manyika *et al.*, "The Internet of Things: Mapping the value beyond the hype.," *McKinsey Global Institute*, 2015. [Online]. Available: http://www.mckinsey.com/business-functions/digital-mckinsey/our-insights/the-internet-of-things-the-value-of-digitizing-the-physical-world. [Accessed: 03-Jan-2017].

[4] "What is free software?," *Free Software Foundation*. [Online]. Available: https://www.gnu.org/philosophy/free-sw.html. [Accessed: 24-Dec-2016].

[5] "Open Source Hardware (OSHW) Definition," *Open Source Hardware Association*. [Online]. Available: http://www.oshwa.org/definition/. [Accessed: 24-Dec-2016].

[6] H. Derhamy, J. Eliasson, J. Delsing, and P. Priller, "A survey of commercial frameworks for the Internet of Things," in *2015 IEEE 20th Conference on Emerging Technologies Factory Automation (ETFA)*, 2015, pp. 1–8.

[7] *Internet of Things Applications - From Research and Innovation to Market Deployment (River Publishers Series in Communications)*. River Publishers, 2014.

[8] D. Miorandi, S. Sicari, F. De Pellegrini, and I. Chlamtac, "Internet of things: Vision, applications and research challenges," *Ad Hoc Networks*, vol. 10, no. 7, pp. 1497–1516, 2012.

[9] A. Pantelopoulos and N. G. Bourbakis, "A Survey on Wearable Sensor-Based Systems for Health Monitoring and Prognosis," *IEEE Trans. Syst. Man, Cybern. Part C (Applications Rev.*, vol. 40, no. 1, pp. 1–12, Jan. 2010.

[10] V. C. Gungor *et al.*, "Smart Grid Technologies: Communication Technologies and Standards," *IEEE Trans. Ind. Informatics*, vol. 7, no. 4, pp. 529–539, Nov. 2011.

[11] "IoTivity project," *Linux Foundation*. [Online]. Available: https://www.iotivity.org/about. [Accessed: 24-Dec-2016].

[12] "TIZEN," *Linux Foundation*. [Online]. Available: https://www.tizen.org/about. [Accessed: 24-Dec-2016].

[13] "Automotive Grade Linux," *Linux Foundation*. [Online]. Available: https://www.automotivelinux.org/about. [Accessed: 24-Dec-2016].

[14] "Yocto Project," *Linux Foundation*. [Online]. Available: https://www.yoctoproject.org/about. [Accessed: 24-Dec-2016].

[15] "Zephyr Project," *Linux Foundation*. [Online]. Available: https://www.zephyrproject.org/about.

[16] "Google Weave," *Google*. [Online]. Available: https://developers.google.com/weave/. [Accessed: 24-Dec-2016].

[17] "Android Things," *Google*. [Online]. Available: https://developer.android.com/things/index.html. [Accessed: 24-Dec-2016].

[18] "Microchip-Atmel: SAM L MCUs," *Microchip - Atmel*, 2017. [Online]. Available: http://www.atmel.com/products/microcontrollers/ARM/SAM-L.aspx. [Accessed: 06-Jan-2017].

[19] "ARM mbed," *ARM*. [Online]. Available: https://www.mbed.com/en/. [Accessed: 24-Dec-2016].

[20] A. Dunkels, O. Schmidt, N. Finne, J. Eriksson, F. Österlind, and N. T. M. Durvy, "The Contiki OS: The Operating System for the Internet of Things," *Contiki OS*, 2011. [Online]. Available: http://contiki-os.org/. [Accessed: 24-Dec-2016].

[21] "RIOT: The friendly Operating System for the Internet of Things," *RIOT*. [Online]. Available: http://www.riot-os.org/. [Accessed: 24-Dec-2016].

[22] E. Baccelli, O. Hahm, M. Gunes, M. Wahlisch, and T. C. Schmidt, "RIOT OS: Towards an OS for the Internet of Things," in *2013 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2013, pp. 79–80.

[23] H. Will, K. Schleiser, and J. Schiller, "A real-time kernel for wireless sensor networks employed in rescue scenarios," in *2009 IEEE 34th Conference on Local Computer Networks*, 2009, pp. 834–841.

[24] "Apache Mynewt: An OS to build, deploy and securely manage billions of devices," *Apache Software Foundation*, 2017. [Online]. Available: https://mynewt.apache.org/about/. [Accessed: 06-Jan-2017].

[25] "NimBLE Introduction," *Apache Software Foundation*, 2017. [Online]. Available: http://mynewt.apache.org/network/ble/ble_intro/. [Accessed: 06-Jan-2017].

[26] S. Li, L. Da Xu, and S. Zhao, "The internet of things: a survey," *Inf. Syst. Front.*, vol. 17, no. 2, pp. 243–259, 2015.

[27] F. Baker *et al.*, "Internet of Things (IoT) Security and Privacy Recommendations," *Broadband Internet Technical Advisory Group*, 2016. [Online]. Available: https://www.bitag.org/report-internet-of-things-security-privacy-recommendations.php.

[28] "Internet of things research study," *Hewlett Packard Enterprise Development LP*, 2015. [Online]. Available: https://www.hpe.com/h20195/v2/GetPDF.aspx/4AA5-4759ENN.pdf. [Accessed: 01-Jan-2017].