Security Perspectives for Collaborative Data Acquisition in the Internet of Things

Vangelis Gazis[1], Carlos Garcia Cordero[1,2], Emmanouil Vasilomanolakis[1,2], Panayotis Kikiras[1], and Alex Wiesmaier[1](✉)

[1] AGT Group (R&D), Darmstadt, Germany
{vgazis,cgarcia,evasilomanolakis,pkikiras,
awiesmaier}@agtinternational.com
[2] Technische Universität Darmstadt, Telecooperation Group, Darmstadt, Germany

Abstract. The Internet of Things (IoT) is an increasingly important topic, bringing together many different fields of computer science. Nevertheless, beside the advantages (IoT) has to offer, many challenges exist, not at least in terms of security and privacy. In addition, the large number of heterogeneous devices in (IoT) produces a vast amount of data, and therefore efficient mechanisms are required that are capable of handling the data, analyze them and produce meaningful results. In this paper, we discuss the challenges that have to be addressed, when data analytics are applied in the context of the (IoT). For this, we propose a data acquisition architecture, named *CoDA*, that focuses on bringing together heterogeneous *things* to create distributed global data models. For each layer of the proposed architecture we discuss the upcoming challenges from the security perspective.

Keywords: Internet of Things (IoT) · Security challenges · Data acquisition · Data analytics

1 Introduction

Over the last decade, the Internet of Things (IoT) has emerged as an umbrella concept for the disruptive application of advances in embedded sensors, low power wireless networking and distributed computing [26]. Its original definition envisioned a world where computers would relieve humans of the Sisyphean burden of data entry by automatically recording, storing and processing, in a proper manner, all information relevant to human activities [18]. With human involvement being the exception, Machine-to-Machine (M2M) communication is understood as a major component of the IoT portfolio of technologies. IoT represents the prime embodiment of the ongoing convergence between device-oriented sensor networks and data-oriented applications. Facilitated by the Internet portfolio of technologies, the latter utilizes data from the physical world captured by sensors to improve processes of modern life (e.g., in industrial manufacturing, health care, energy production, etc.). Not surprisingly, key industry players, as

R. Giaffreda et al. (Eds.): IoT360 2014, Part II, LNICST 151, pp. 271–282, 2015.
DOI: 10.1007/978-3-319-19743-2_38

well as prominent market analysts, have repeatedly acknowledged the importance of IoT and its economic impact [10,11,17,24].

Recently, multiple interpretations of what the (IoT) is about have been proposed [8,19,25]. The European Commission (EC) defines IoT as "things having identities and virtual personalities operating in smart spaces using intelligent interfaces to connect and communicate within social, environmental, and user contexts" [22]. The use of Web technologies in the IoT to support peer-to-peer interactions between things is referred to as the Web of Things [20]. The Telecommunication Standardization Sector of the International Telecommunication Union (i.e., ITU-T) defines IoT as a global (i.e., distributed) infrastructure for the information society, enabling advanced services by interconnecting a disparate gamut of (physical and virtual) things based on, existing and evolving, interoperable information and communication technologies [23]. Not surprisingly, security and privacy concerns are paramount in this emerging world where things may autonomously communicate and exchange information with each other in a way that is transparent to human beings.

Transparent M2M communication is the key aspect of the IoT, enabling collaboration between devices and, as a result of this collaboration, the creation of new data. Analyzing this new data becomes of crucial importance as many hidden variables can be data-mined. The endless possibilities of collaboration and the communication ubiquity between devices requires a new architectural model that would enable the processing of large amounts of data in the context of IoT. Data analytics has to be built, therefore, on top of a new architecture for accessing and gathering data, which we denominate herein as the (CoDA) architecture.

In the usual context of data-mining and analytics, CoDA becomes the bottom layers that represent the storage locations where data is gathered. In the IoT, centralized locations where all generated data is stored cannot be expected. With CoDA, it is possible to build typical data analytics tools that acquire all information through a common layer that encompasses all device communication capabilities in the IoT.

Herein, we highlight and elaborate upon the challenges associated to the application of data analytics (e.g., model learning, data fusion, etc.) in the IoT. We propose a data acquisition architecture, named *CoDA*, that enables heterogeneous *things* to be brought together and support the creation of distributed global data models useful for data-mining. Furthermore, for each layer of the proposed architecture we discuss major challenges from a security perspective.

The remainder of this paper is organized as follows. In Sect. 2 we discuss how a data analytics architecture has to be adapted for the IoT. Subsequently, in Sect. 3, we propose a sub-architecture, named CoDA, that addresses specific IoT challenges in the data acquisition layer, and discuss the respective security challenges. Finally, Sect. 5 concludes this paper and discusses future directions.

1.1 Related Work

Established in 2012, oneM2M is a partnership among major ICT standards development organizations around the world [28,29]. The founding SDOs include the

Association of Radio Industries and Businesses (ARIB), the Telecommunication Technology Committee (TTC) of Japan, the Alliance for Telecommunications Industry Solutions (ATIS), the Telecommunications Industry Association (TIA), the China Communications Standards Association (CCSA), the European Telecommunications Standards Institute (ETSI) and the Telecommunications Technology Association (TTA) of Korea. Currently numbering approximately 200 members, oneM2M is developing joint specifications and technical reports for the M2M service layer. oneM2M is also liaised with major industry alliances (e.g., Open Mobile Alliance, BroadBand Forum) and Internet standardization bodies (e.g., IETF, IEEE). Candidate Release 1 of the oneM2M specifications was recently published. These specifications define the functional architecture in terms of logical elements, their functional capacities and their interfaces. They also define the oneM2M core protocol and its technology bindings to the HTTP and CoAP protocols, as well as device management enablers that incorporate the respective standards from the BroadBand Forum (BBF) [16] and the Open Mobile Alliance (OMA) [27]. Collectively, these form a common services layer for a wide range IoT applications.

The Routing Over Low power and Lossy networks (ROLL) working group of IETF is developing a routing architectural framework for the IPv6 protocol tailored to resource-constrained devices (e.g., embedded devices). The IETF Constrained RESTful Environments (CORE) working group is developing a framework for resource-oriented applications intended to run over the IP protocol on resource-constrained networks (e.g., M2M networks based on the IEEE 802.15.4 standard [21]).

Regional Standards. The European Telecommunication Standards Institute (ETSI) published Release 1 of its M2M standard on November 2011. These introduce an M2M platform for M2M service providers where IoT applications are supported through platform agnostic interfaces [12–14]. They thus define a system architecture that enables integration of a diverse range of M2M devices (e.g., sensors, actuators, gateways, etc.) into an end-to-end platform. The latter offers to applications a standard interface for accessing data and services made available over these (typically last mile) devices.

Starting with Release 10, 3GPP has included support for (device terminated and device originated) Machine Type Communications (MTC). The objective is to ensure that 3GPP network installations will support M2M applications deployed on a very large scale [1]. 3GPP2 has assessed the traffic impact of M2M applications on the cdma2000 network infrastructure (e.g., huge population of communicating devices, low traffic volume per device, etc.) [2] and amended its specifications accordingly. M2M work in 3GPP2 is now aligned to the M2M work in 3GPP and the M2M architecture work done in ETSI as part of an access-agnostic architecture [2–4].

In the Telecommunications Industry Association (TIA), Engineering Committee TR-50 M2M on Smart Device Communications (SDC) has developed an M2M framework. The latter abstracts the technological details of underlying transport networks (wireless, wireline, etc.) and provides a convergence layer for M2M applications [31].

The ATIS M2M Focus Group (FG) has addressed the M2M, Smart Grid and Connected Vehicle markets, with M2M understood as an horizontal layer spanning across multiple vertical domains (e.g., Smart Grid, Connected Vehicle). The ATIS M2M work is currently integrated to the 3GPP MTC work program [7] and further advanced there. ATIS has addressed carrier portability issues (e.g., waiving the need for SIM card swapping, remote reconfiguration, etc.), management procedures (e.g., provisioning, billing, etc.), transport (e.g., peering) and security issues of IoT applications.

2 Data Analytics Architecture for the IoT

A generic framework of an IoT data analytics framework needs to deal with a high degree of heterogeneity, e.g., in terms of storage facilities of the underlying infrastructure, data types and representation formats, processing modes, and, analytic algorithms. These concerns are reflected in its stratification which includes four layers, i.e., *Scalable Analytics, Analytics Enabler, Scalable Processing* and *Data Acquisition*. Figure 1 shows an overview of a generic IoT data analytics architecture.

2.1 Scalable Analytics

The *Scalable Analytics* layer encompasses the range of use cases that employ data analytics in the context of IoT (i.e., the vertical domains of IoT data analytics). As seen in Fig. 1 these include (but are not limited to) critical infrastructure

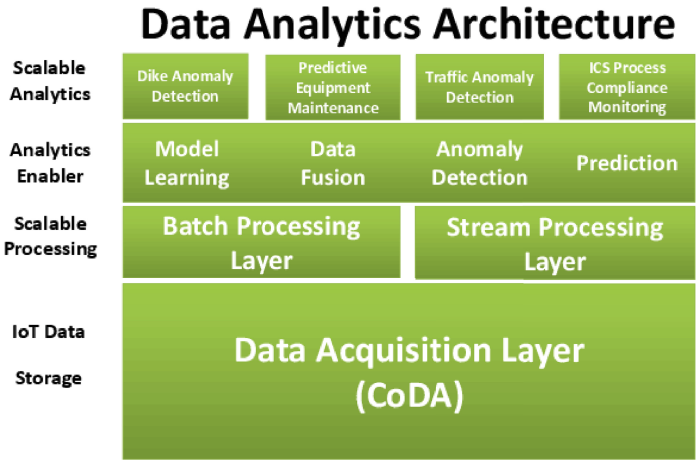

Fig. 1. High level view of a generic data analytics architecture for the IoT

monitoring (e.g., dike anomaly detection), industrial IoT (e.g., predictive equipment maintenance, ICS process compliance monitoring) and Smart City (e.g., traffic anomaly detection). However, in the rest of this paper we focus on the subsequent layers, i.e., the specific mechanisms that are required in order to provide the *Scalable Analytics* with the necessary capabilities and relevant data.

2.2 Analytics Enabler

The *Analytics Enabler* layer provides the hosting environment for the deployment and execution of analytic algorithms, along with their associated model data. It provides mechanisms to deploy/undeploy a packaged analytic algorithm in a particular hosting environment, e.g., an Amazon Machine Image, an OSGi container. To this end, it realizes capability negotiation mechanisms to match the requirements and constraints of a particular analytic algorithm to those of the available hosting environments. Provisioning a deployed analytic algorithm with the necessary model data is also supported by the capability negotiation mechanism as part of the post-deployment and pre-execution phases. Finally, it includes a workflow engine for scheduling the execution of compositions of analytic algorithms and the coordination of their input/output dependencies.

From the perspective of analytics function, the majority of IoT applications, including security and privacy ones, are based on a combination of the following capacities:

1. *Model Learning*, where a formal representation of (some aspects of) the real world is built from recorded measurements of relevant phenomena in the real world. For instance, IoT traffic management applications typically feature the building of a model of observed traffic patterns.
2. *Data Fusion*, where multiple pieces of data (which may be of different types and modalities) are combined to render data of better quality, e.g., in terms of accuracy in data values. For instance, observations from different sensors that are proximal to each other can be combined to provide a more accurate observation about a phenomenon in their particular location.
3. *Real-Time Anomaly Detection*, where (real-time or near real-time) measurements of particular phenomena in the real world are contrasted to values estimated from a model (e..g, one that has been developed through a model learning process for those particular phenomena, formulated analytically, or, a combination of both approaches) and any observed deviations are reported. This is particularly relevant and frequent for administrative levels tasked with real-time control of critical assets, e.g., Demand/Response control processes in a Smart Grid environment or traffic management processes in the context of Smart City.
4. *Prediction*, where (possibly real-time) measurements of particular phenomena in the real world are applied to a model (that has been developed through a model learning process for those particular phenomena) to generate a forecast (e.g., forecasted demand for electricity in a particular urban area).

There are additional analytics capacities involved in other IoT applications, however, these are either more of an elementary nature (e.g., data clustering), or, feature in a small niche of IoT applications (e.g., simulation) [6,15].

2.3 Scalable Processing

The *Scalable Processing* layer realizes the computational capabilities required by the *Analytics Enabler* layer. It provides computational resources, e.g., CPU cycles, thread pools, to support the execution of each particular analytic algorithm. Different computing modalities are supported, depending on the real-time profile of the data fed to the analytic algorithm.

- *Batch Processing* where input data is not subject to a real-time requirement, e.g., historical data at rest. Typically this is achieved through the Map/Reduce paradigm of computing supported by the popular Hadoop framework that achieves scalability under a bulk mode of computation.
- *Stream Processing* where input data is subject to a real-time requirement, e.g., streaming real-time data. Commonly, this regards the treatment of events (where an event describes the occurrence of a significant situation in terms of attribute-value pairs) in a so-called Complex Event Processing (CEP) framework founded on the Event-Condition-Action (ECA) paradigm. With different levels of significance attached to each event and with unpredictable times of occurrence for each event, this mode of processing fits the requirements of a process in control of assets of importance, e.g., operational procedures occurring within a Smart Grid context.

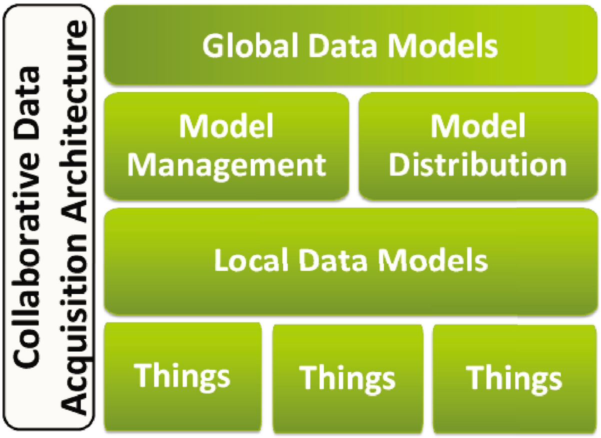

Fig. 2. Different layers of the envisioned Collaborative Data Acquisition architecture

2.4 Data Acquisition

The *Data Acquisition* layer describes all the processes that are required to produce meaningful data starting from the heterogeneous layer of *things* to the global data models. We argue that, in the context of IoT, this layer is of particular importance as it has to deal with the shortcomings of generating data from a large number of diverse and low-resourced devices. Therefore, we extensively discuss this layer and propose a distinction of four additional sub-layers in the upcoming Sect. 3.

3 CoDA: Collaborative Data Acquisition Architecture

In this section we describe our vision of a collaborative data acquisition architecture for the IoT, as well as the respective security challenges. Figure 2 provides an overview of the so-called CoDA architecture.

3.1 Data Acquisition Architecture

The IoT is composed of multiple objects, or *things*, that are capable of generating data constantly. Each such object analyzes and creates data according to its particular capabilities. That is to say, each different thing generates data that concerns only the phenomena it is capable of sensing, i.e., it is equipped with sensory instruments for. The result heterogeneity of the data complicates matters when different processes need to access a collection of all the data generated by things. To enable processes to access heterogeneous data, it is important to develop an architecture capable of conveying each thing with the ability of communicating data among itself and requesting processes in a common format.

Once mechanisms for acquiring and collecting data are in place, more advanced computations (by *things*) can be performed on top of the data easily. The IoT promises the availability and generation of vast amounts of data. If this data is easily accessible following security and user guidelines, services will be able to leverage *things* efficiently and produce valuable information. However, data acquisition is the fundamental and core aspect of any service that could be conceived and provided in the IoT.

Simple data collection services would already benefit from such a data acquisition architecture; however, services that embody an advanced analytic function, e.g., machine learning, would benefit the most. The standard flow of operations of such services (and machine learning algorithms in particular) requires, first and foremost, access to data.

This architecture proposes a four layer approach at making data available without requiring a central location to collect all data. The four layers are *things*; *local data models*; *model management and distribution*; and finally the *global data models*. The layer of *things* is concerned with the actual data generated by each individual set of *things*. The second layer of local data models is responsible for representing data from the lower layer (the *things* layer) so that it can be easily managed (i.e., it provides a convergence layer to enable management procedures). The third layer is a composition of two components, a component for managing the generated local data models and another component for distributing the models. The last layer, the global data layer, encompasses a collection of all data models that is easily accessible by users or service providers.

As previously stated, this architecture does not require a centralized unit to collect all data in one single location. Each thing is responsible for generating local data models that are shared among other *things* in their vicinity. Neighborhoods of *things* are created, e.g., by leveraging standard neighbour discovery protocols such as IPv6, where the same type of information is shared among themselves. An additional benefit of the local sharing of data is the facilitation of increased data availability and improved performance in accessing the data (i.e., as a distributed cache).

When a user queries, for instance, one particular data type, the user is able to issue such queries to any thing and this thing will be able to redirect the query to the right owner of the data.

In what follows we describe the components of the CoDA architecture as well as the security concerns that are entailed in the implementation of this architecture.

3.2 Introduction to the CoDA Architecture

We present an architecture that enables *things* to provide platform independent access to their data while also following security and user requirements. Figure 2 shows the different layers of CoDA.

Things. Each component in the IoT is essentially a *thing*. The bottom most layer of the architecture encompasses groups of *things*. Regardless of the *things* being exactly the same, this layer groups them by the data they produce. A group of things is a collection of *things* that produce the same data types.

Each group of *thing* is able to generate the same data conforming to specifications and the specific characteristics of the incorporated sensors. However, for the resulting communication to be effective, a common ground for the data generation process is required. This also requires standardized protocols for communication so that convenient data access is possible. These result in a multitude of *things* producing different data represented by the same common output format. In addition, each device in this layer requires a secure and unique identifier, e.g., a common secret, that enables interactions between owners and other *things* to be structured with the appropriate level of security [9,30]. Simultaneously, every sensor needs to be able to support more than one consumer for the data it generates. Lastly, *things* are able to determine their owner, e.g., through a common secret. Determining ownership relationships allows *things* to choose the data to share.

Local Data Models. The second layer of the CoDA architecture is a data normalization component . The layer consists of models, standards and formats which specify the way and form of communication for all *things*. This is necessary due to the fact that by default the data produced by *things* would not necessarily conform to any particular standard. The normalization component effectively provides the required data convergence.

Data Management and Model Distribution. When each thing is able to produce meaningful output, the question that arises is how to distribute data efficiently. Due to the fact that in the IoT a vast amount of devices and sensors are interconnected, we envision distributed and P2P techniques being utilized in this level (in contrast to centralized approaches that would not scale). This can be realized by adopting mechanisms from the P2P and the wireless mesh networks areas [5], which allows *things* to create arbitrary communication links among each other.

Furthermore, in the basis of the existence of a basic communication overlay, a distribution layer in parallel is of high importance. At this point, the main challenge that needs to be addressed is determining which *things* should communicate with each other. A combination of the type of *things* as well as the knowledge of a common secret can provide an initial way to deal with this.

Global Data Models. The global data model layer is based on the fact that local models can be aggregated via the utilization of the aforementioned layers to form a mixed model. This has a twofold objective. First, it enables *things* to expand their capabilities through collaboration. For instance, a sensor can make use of data gathered from other sensors to perform complex tasks. Second, various services can query this layer to receive mixed information from the interaction of multiple *things*.

4 Security Challenges

Due to the unattended environment in which the *things* operate and the resource constrained nature of these devices in terms of computational capabilities, memory size, and available energy, it is not possible to just employ security schemes coming from regular computers or ad hoc wireless networks domain. Challenges in securing the IoT go beyond the standard C-I-A model. IoT security must and be lightweight to run in constrained devices, scalable to billions of devices, supporting heterogeneous devices, work in an unattended manner, and supporting decentralized solutions.

4.1 Data Acquisition

In order to implement the CoDA architecture in a secure way, many challenges need to be addressed [30]. The two main problems in the IoT is the absence of centralized points of service and the existence of numerous low-resource devices with power limitations. As a result, resource intensive mechanisms such as asymmetric cryptography or centralized architectures, like PKI infrastructures, are limited. Nevertheless, as not all *things* require strong security properties, more lightweight symmetric cryptographic techniques can be utilized. Moreover, sophisticated trust models need to be built to enable the deployment of access control limitations between owners and *things*.

It is necessary to distinguish in the layer of *things* who are the owners of a *thing* and manage the respective access control list of each. A common shared secret between the owners and a *thing* can be utilized along with unique identifiers. Moreover, the first two layers provide the necessary encryption services for guarantying the confidentiality and integrity of the generated data.

The model management and distribution layer need to be scalable, and resilient against attacks and failures. This means that the overlay has to function even when a number of sensors might not be working. Regardless of the utilized overlay, the architecture must provide sufficient capabilities to distinguish trusted *things* and foreign *things*. Finally, the global data models must ensure proper access control enforcement, and authentication for the *things* and owners.

A high level realization of the security challenges in our proposal can be envisioned by the utilization of a complex multi-layer symmetric-based encryption scheme. Multi-layer in this context refers to the various levels of trust that *things* must accommodate for a flexible and secure interaction among themselves and the owners. For instance, devices can have more than one owner, spanning up to an entire organization, and also different protocols might be used for their interaction.

4.2 Privacy and Trust

In general and especially in the IoT, privacy is more than just keeping personal information confidential. Examples for other privacy aspects that have to be considered are given in the following. *Identity privacy* refers to disclosing a user's identity if and only if needed and keep it secret otherwise. *Query privacy* refers to data retrieval without revealing to the sender (or any other party) which data was received. *Location privacy* means hiding a user's location to the reasonable extent whenever possible. *Footprint privacy* aims at minimizing a user's linkable (meta) data volume.

Similar to privacy, trust is in general, especially in the context of IoT, more than just relying on third parties. Four different areas of trust can be distinguished. *Device trust* refers to the user's confidence to interact with reliable sensors, as well as assessing the accuracy of the produced data. Sensors that are not handled according to specifications have the risk of producing inconsistent or wrong data. *Processing trust* reflects the need to deal with correct and meaningful data. *Connection trust* embodies the desire to exchange data only with the intended partners. *System trust* refers to the ability of using the interaction between *things* to confidently accomplish complex tasks.

4.3 Attacks

Just as any other network, an IoT network is subject to different types of attacks. Examples for attack categories that have to be considered are given in the following. *Physical attacks* on devices, e.g., destroying, analyzing, and/or reprogramming them. *Service disruption attacks* on routing, localization, etc. *Data attacks* such as traffic capture, spoofing, and similar. *Resource-consumption and denial-of-service (DoS) attacks* on (remote) resources and/or services.

5 Conclusion

The IoT is a prevalent concept that aims to permeate common things with the possibility of offering and consuming services. Each *thing* is capable of generating data through its sensors or by consuming services of other *things*. To handle the large amount of data that will be generated, a common architecture needs to be designed that will enable efficient and simplified communication between *things*.

We highlighted the challenges brought by the adoption of data analytic applications in the IoT and proposed the *CoDA* data acquisition architecture. *CoDA* enables heterogeneous *things* to be brought together efficiently in support of distributed global data models for data analytic applications. Our presentation of *CoDA* focused on security concerns, as these are paramount under a global data model. We foresee extensions of our work in the refinement of *CoDA* in the direction of privacy preserving approaches that are generically applicable to data analytics applications.

References

1. 3GPP. Ts 23.887; study on machine-type communications (mtc) and other mobile data applications communications enhancements (2013)
2. 3GPP2. S.r0141-0; study for machine-to-machine (m2m) communication for cdma2000 networks (2010)
3. 3GPP2. X.p0067-0; machine to machine (m2m) architecture and enhancements study for cdma2000 networks (2012)
4. 3GPP2. X.s0068-x; cdma2000 network enhancements for m2m (2012)
5. Alcaraz, C., Najera, P., Lopez, J., Roman, R.: Wireless sensor networks and the internet of things: Do we need a complete integration? In: 1st International Workshop on the Security of the Internet of Things (SecIoT 2010) (2010)
6. Future Internet Assembly: Internet of Things: an early reality of the Future Internet. Technical report, European Commission (2009)
7. ATIS. Assessments and Recommendations (2013)
8. Atzori, L., Iera, A., Morabito, G.: The internet of things: a survey. Comput. Netw. **54**(15), 2787–2805 (2010)
9. Atzori, L., Iera, A., Morabito, G., Nitti, M.: The Social Internet of Things (SIoT) when social networks meet the Internet of Things: concept, architecture and network characterization. Comput. Netw. **56**(16), 3594–3608 (2012)
10. Cisco. The Internet of Things (2013)
11. Ericsson. More than 50 billion connected devices (2013)

12. ETSI. Ts 102 689; machine-to-machine communications (m2m); m2m service requirements (2013)
13. ETSI. Ts 102 690; machine-to-machine communications (m2m); functional architecture (2013)
14. ETSI. Ts 102 921; machine-to-machine communications (m2m); mia, dia and mid interfaces (2013)
15. Future Media Internet Task Force: Future Media Internet Research Challenges and the Road Ahead. Technical report, European Commission (2010)
16. BroadBand Forum. Tr-069; cpe wan management protocol (2013)
17. Gartner. The Internet of Things (2013)
18. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of things (iot): a vision, architectural elements, and future directions. Future Gener. Comput. Syst. **29**(7), 1645–1660 (2013)
19. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of things (IoT): A vision, architectural elements, and future directions. Future Generation Computer Systems (2013)
20. Guinard, D., Trifa, V., Mattern, F., Wilde, E.: From the internet of things to the web of things: resource-oriented architecture and best practices. In: Uckelmann, D., Harrison, M., Michahelles, F. (eds.) Architecting the Internet of Things, pp. 97–129. Springer, Heidelberg (2011)
21. IEEE Standards Association. IEEE Standard for Local and metropolitan area networks Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs), September 2011
22. D G INFSO. Internet of Things in 2020: A Roadmap for the Future, INFSO D.4 Networked Enterprise & RFID and INFSO G.2 Micro & Nanosystems in co-operation with RFID Working Group of the European Technology Platform on Smart Systems Integration (EPOSS) (2008)
23. ITU-T. Recommendation itu-t y.2060, overview of the internet of things, June 2012
24. McKinsey. Disruptive Technologies (2013)
25. Miorandi, D., Sicari, S., De Pellegrini, F., Chlamtac, I.: Internet of things: Vision, applications and research challenges. Ad Hoc Netw. **10**(7), 1497–1516 (2012)
26. National Intelligence Council NIC. Disruptive Civil Technologies: Six Technologies with Potential Impacts on US Interests out to 2025, April 2008
27. OMA. Oma device management v1.2 (2013)
28. oneM2M Partners. oneM2M Partnership Agreement, July 2012. http://www.onem2m.org/docs/oneM2M_Partnership_Agreement.pdf (Accessed 10 February 2014)
29. oneM2M Partners. oneM2M Homepage, January 2014. http://www.onem2m.org/index.cfm (10 February 2014)
30. Roman, R., Najera, P., Lopez, J.: Securing the internet of things. Comput. **44**(9), 51–58 (2011)
31. TIA. Tia-4940.005: Smart device communications reference architecture (2012)