NOTICE: This is a pre-copyedited version of a contribution published in Advances on P2P, Parallel, Grid, Cloud and Internet Computing (Fatos Xhafa, Fang-Yie Leu, Massimo Ficco, Chao-Tung Yang, eds.) published by Springer International Publishing. The definitive authenticated version is available online via https://doi.org/10.1007/978-3-030-02607-3_24

Automated Risk Analysis for IoT systems

Massimiliano Rak, Valentina Casola, Alessandra De Benedictis and Umberto Villano

Abstract Designing and assessing the security of IoT systems is very challenging, mainly due to the fact that new threats and vulnerabilities affecting IoT devices are continually discovered and published. Moreover, new (typically low-cost) devices are continuously plugged-in into IoT systems, thus introducing unpredictable security issues. This paper proposes a methodology aimed at automating the threat modeling and risk analysis processes for an IoT system. Such methodology enables to identify existing threats and related countermeasures and relies upon an open catalogue, built in the context of EU projects, for gathering information about threats and vulnerabilities of the IoT system under analysis. In order to validate the proposed methodology, we applied it to a real case study, based on a commercial smart home application.

1 Introduction

Over the last few years, the Internet of Things (IoT) has become one of the prominent emerging technologies for delivering value-added services to end users.

Securing IoT systems presents a number of unique challenges that depend on many different factors, including: (i) the heterogeneity of the IoT devices (mainly programmable devices and embedded systems) that have different hardware and software constraints, (ii) the heterogeneity of communication protocols (ranging

Massimiliano Rak
Università della Campania Luigi Vanvitelli, DI, Aversa (CE), e-mail: massimiliano.rak@unicampania.it

Valentina Casola and Alessandra De Benedictis
Università di Napoli Federico II, DIETI, Napoli, e-mail: {casolav, alessandra.debenedictis}@unina.it

Umberto Villano
Università del Sannio, DING, Benevento, e-mail: villano@unisannio.it

from ad-hoc, low-power connections to wi-fi networks), and (iii) the vulnerabilities of the deployment environments, that range from smart homes [16] to critical infrastructures [4] that widely adopt distributed and remote services in the cloud. The analysis of the security issues affecting IoT systems has been object of several surveys published recently (e.g., [2, 5, 15, 1]), which have highlighted that the most critical factors are: (i) the need to continuously adapt to the environment, due to tyhe dynamic introduction and/or removal of devices, and (ii) the low power and capacity of many interconnected devices, that inhibit the adoption of complex security mechanisms. Accordingly, systems should be designed and managed by taking into account the security and the capability of each new device, which may affect the overall security level of the architecture. Unfortunately, risk analysis and security assessment are costly procedures, and they are rarely applied in systems where cost is a strict constraint (e.g., smart home systems).

In this paper, we propose a methodology aimed at automating, as much as possible, the threat modeling and risk analysis processes for an IoT system. Our approach enables to easily identify the assets to protect, their vulnerabilities and the existing related threats, the effective risks they are subject to and the countermeasures to apply in order to mitigate such risks. In particular, the proposed approach relies (i) on the ISO standard model to describe IoT systems, (ii) on an open catalogue of well-known threats affecting different assets and communication protocols to identify the threats of interest for the IoT system under analysis, (iii) on the STRIDE threat classification and on the OWASP risk rating methodology for automating the risk analysis, and (iv) on standard security control frameworks (e.g., NIST800-53 and ISO 27000) to describe the countermeasure and verify their correct implementation.

The remainder of this paper is structured as follows: in Section 2, we briefly summarize the adopted reference architecture to model IoT systems and its components. In Sections 3, we illustrate the proposed methodology to automate threat modeling and risk assessment of IoT systems. In Section 4, we provide some details on the modeling activities, by also introducing a case study home automation system used to better illustrate the methodology. In Section 5, we discuss how it is possible to automatically obtain a threat model for the system, by also giving some concrete example related to the case study. Finally, in Sections 6 and 7 some related work is presented with conclusions and future work.

2 What is an IoT system

In this paper, we adopt the ISO Reference Architecture presented in the ISO/IEC 30141 document [6] as the baseline to model IoT systems and their architecture. The ISO/IEC 30141 provides a complex reference model, including a conceptual model describing the entities involved in an IoT system and their relationships, and several architectural views. These include, among others, the *functional view*, which represents, in a technology-agnostic way, the high-level functionalities that are necessary to form an IoT system. The functional view is organized in domains: at the

bottom, there is the *Physical Entity* domain (PED), with the *Sensing & Controlling* domain (SCD) above it. The *Operation & Management* (OMD), *Application Service* (ASD) and *IoT Resource and Interchange* (RID) domains are logically positioned at the same level, on top of the *Sensing & Controlling* domain and below the *User* domain (UD).

The functionalities identified in the functional view are implemented by suitable components included in the *system view*: for example, controlled and sensed physical objects belong to the PED, while sensors, actuators, gateways and local control systems belong to the SCD.

All the concepts involved in an IoT system are reported in the *conceptual model*, which describes the main IoT entities and their relationships. The *IoT Device* is the entity that bridges between real-world *Physical Entities* and the other digital entities in the system, and interacts with other entities through one or more *networks*. An IoT Device can be either a *Sensor*, able to monitor a physical entity and transform some of its characteristics into a digital representation that can be communicated, or an *Actuator*, able to act on one or more properties of a physical entity on the basis of received commands. The *Service* entity represents a set of distinct capabilities implemented by one or more software components that is directly accessed by a digital user. An *Application* is a service that offers a collection of functions that can be accessed by a human user to perform a task. In the IoT context, it implements the functionalities typical of the application domain (eHealth, smart home, etc.). The *IoT Gateway* is a digital entity that connects one or more IoT Devices to a wide-area network. The IoT Gateway typically interacts with IoT Devices through short-range networks, and with Services through high-bandwidth networks. Both IoT Gateways and Services use a *Data Store*, which holds data relating to the IoT system, either derived from IoT devices or resulting from services acting on IoT device data.

As illustrated later in the paper, we will adopt these concepts and components as the basis to model any IoT system and to perform the risk analysis and security assessment.

3 Automated Risk Analysis Methodology

As shown in Figure 1, our risk analysis methodology comprises four main steps, namely *Modeling*, *Asset Threats Identification*, *Risk Analysis* and *Security Controls Identification*.

In the **Modeling** step, the target IoT system is analyzed in order to identify the architectural assets and their relationships, and is first modeled based on the ISO reference model discussed in the previous section (*ISO System Model generation* sub-step). In particular, the specific components of the IoT system under analysis are first mapped onto the entities of the ISO conceptual model, and then a technology-dependent system view is built for the system according to the ISO guidelines. Then, the ISO-compliant model is automatically translated into another formalism, i.e., the *MACM* graph-based formalism introduced in [13] (*MACM generation sub-step*),

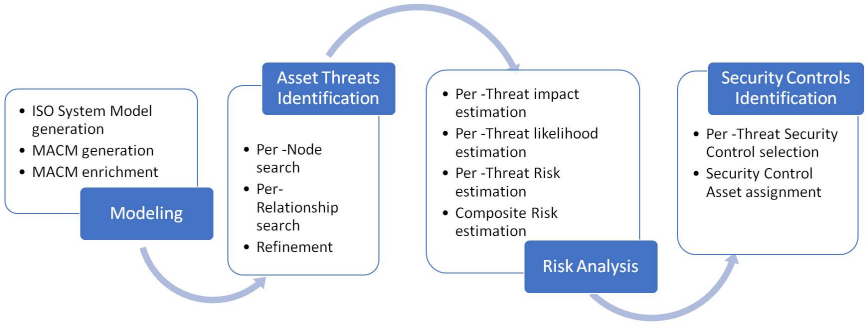

Fig. 1 IoT Automated Risk Analysis Methodology

which enables to easily represent system components, their relationships and security features, and to perform an automated assessment of the security of a system by means of suitable graph manipulations. In the *MACM enrichment sub-step*, the MACM model of the target system is enriched with additional information, obtained by querying the human assessor and aimed at identifying the threats potentially affecting each asset of the system. In particular, the questions posed to the assessor are useful to identify the specific type of asset, where needed (e.g., is a network asset a radio network, LAN or a WAN?, is a network asset a wired or a wireless network? is a service asset a web-based service?, is an IoT device an open-platform device?, etc.), the type of protocol used in a communication (e.g., XMPP, Zigbee, TLS/SSL, IP, HTTP, HTTPS, TCP), the role of a node in a communication protocol (e.g. server node, client node, peer node,...).

Based on gathered information, in the **Asset Threat Identification** step all relevant threats are first identified for each node and each relationship in the graph (*Per-Node Search sub-step* and *Per-Relationship Search sub-step*). This set is then refined based on the answers given by the assessor in the *Refinement sub-step*, in order to identify the threats that are actually relevant to the target IoT system.

In the **Risk Analysis** step, an estimation of the risk associated with each identified vulnerability is computed as the combination of the likelihood that the vulnerability is exploited and the resulting impact, as devised by the Owasp Risk Rating Methodology [12] (*Per-Threat Likelihood estimation*, *Per-Threat Impact estimation* and *Per-Threat Risk estimation sub-steps*). The risk values are then used to evaluate the overall risk severity with respect to the STRIDE threat categories proposed by Microsoft [9], i.e., Spoofing, Tampering, Repudiation, Information-Disclosure, Elevation-of-Privileges (*Composite Risk estimation sub-step*).

Finally, in the **Security Controls Identification** step, a list of possible countermeasures, in terms of security controls (belonging to a standard framework such ad the NIST Security Control Framework [11]), is selected (*Per-Threat Security Control selection sub-step*) and mapped to the assets to be protected (*Security Control Asset assignment sub-step*). The identified security controls are then included in the

system architecture to refine and finalize the design, with a subsequent update of the model.

It is worth noting that the above process is almost fully automated, thanks to the availability of a *threat catalogue* that suitably maps threats to assets and to security controls in order to enable the *Asset Threat Identification* and the *Security Controls Identification* steps, respectively. The catalogue was developed in the context of the FP7 SPECS project and H2020 MUSA project, it is available on line[1] and is continuously enriched when new threats and vulnerabilities are discovered. As said, a human intervention is needed only in the *Modeling* phase, to build the initial model of the system and to reply to the questions that help refine the model. In this regard, it is worth mentioning that also the questionnaire used for model refinement is part of the threat catalogue, as questions are directly mapped to assets and threats.

4 Modeling

As anticipated, the *Modeling* step of the proposed methodology relies upon the MACM formalism, which was introduced in the context of the security assessment of cloud applications [13], and that has been extended in this paper to include IoT-specific aspects and automate the assessment of IoT systems' security.

The original version of the formalism enables to represent the typical components and relationships of a cloud environment, by defining specific node types to model cloud services (i.e., *IaaS*, *PaaS* and *SaaS* node types) and providers (i.e., the *CSP* node type), and by considering relationships like *use*, *host* and *provide*. The MACM IoT extension introduces further node types and relationships by leveraging the concepts included in the ISO standard briefly described in section 2. In particular, we introduced the node types *IoTDevice*, *IoTGateway*, *Network*, *Entity*. The *use* relationship has been extended to specify that any Software-as-a-Service (SaaS) node can use any IoTDevice node. A property may specify the protocol adopted for such interactions and other protocol-related features. Even the *host* relationship, which was originally adopted to describe an IaaS service hosting any SaaS or PaaS service, has been extended to specify that an IoTGateway may host a SaaS or PaaS component. Finally, we added the *connect* relationship, which links any physical system (IaaS resource, IoTGateway or IoTDevice) to the network infrastructure it is connected to. It is worth noting that, in the IoT environment, different and not connected networks may be involved, due to the short-range communications typically existing among devices.

A case study: The MicroBees home automation system.

In order to illustrate the proposed approach, we will consider a home automation system built by exploiting the Microbees IoT technology [8]. MicroBees offers a set of components devoted to offering simple and cheap home automation functionalities. Such components interact via radio by means of a custom protocol, and are

[1] www.bitbucket.org/cerict/sla-model

coordinated by a dedicated gateway that adopts cloud services to offer advanced user interface and improved automation capabilities. The end user interacts with the system through a mobile phone, by accessing the cloud services that communicate with the *GateBee* component. MicroBees offers four different devices, namely *WireBee*, able to monitor different physical features, *SenseBee*, acting as both a sensor and an actuator, *GateBee*, which is the central gateway that receives commands and data and communicates with SenseBee and WireBee via wireless, and *SecureBee*, which is able to track any object moving in a physical environment.

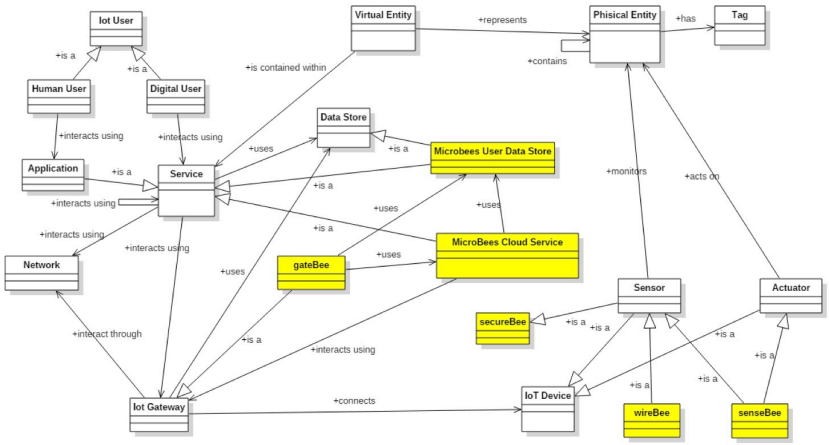

Fig. 2 Microbees Reference Architecture in the ISO model

Figure 2 shows the mapping of Microbees components onto the ISO concepts introduced before. In order to analyze a concrete home automation system, let us consider a simple deployment consisting of four different Actuator devices, controlling Garden lights, Entrance lights, Kitchen lights and Thermostat, respectively, and one Sensor, i.e., the Thermometer. The ISO-compliant system model of such system is depicted on the left of Figure 3, while on the right the corresponding MACM model is reported.

5 Risk Analysis Automation

As anticipated, the *Asset Threat Identification*, *Risk Analysis* and *Security Controls Identification* steps of the methodology introduced in Section 3 can be automated thanks to a threat catalogue, which includes several well-known threats collected from available literature, suitably mapped to the assets identified by the ISO standard and classified based on the related STRIDE category. As shown in Figure 4, which reports an extract of the catalogue, we collected several information for each

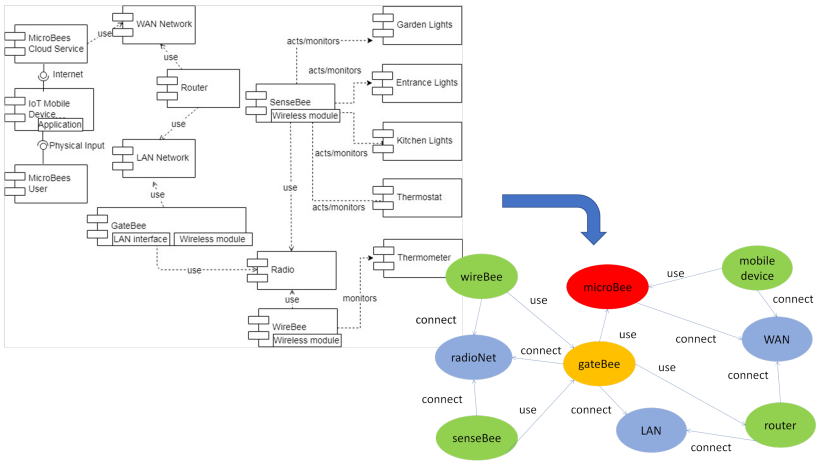

Fig. 3 MicroBees home automation system - System View and MACM model

threat, including the specific type of asset to which it applies, the weakness in the asset configuration that may lead to the threat exploitation, and the security controls that should be enforced as a countermeasure (we currently support the security controls suggested by the NIST framework [11]). Moreover, we also collected information on well-known threats targeting the communication protocols, in order to provide more detailed results during the *Asset Threat Identification* step. We currently support ethernet, IP, TCP, TLS, XMPP, OAUTH, zigbee, and bluetooth, and we are continuing updating the threat collection.

Threat	Description	ISO	Type	STRIDE	Weakness	control
Eavesdropping	An adversary can easily retrieve valuable data from the transmitted packets that are sent	network	radio	Information Disclosure	Lack of Transport Encryption, Channel Accessible by Non-Endpoint	AC-4, AC-16, AC-17, SC-7, SC-8, SC-10, SC-12, SC-13, SC-17, IA-2, IA-7
Data Leakage	An adversary can access to local data of the asset	iotdevice, iotgw, software	peer, client, server, cms	Information Disclosure, Spoofing	Memory Access, Insufficient Authentication, Insufficient Authorization, Insecure Software	AC-7(2), AC-19, IA-3, IA-3(1), SA-18, SC-41, IA-5, SC-8, SI-2, RA-5(1)
Message Modification	An adversary can simply intercept and modify the packets' content meant for the base station or intermediate nodes	network	radio	Information Disclosure, Spoofing, Tampering	Channel Accessible by Non-Endpoint, Lack of Transport Encryption	AC-16, AC-17, SC-8, SC-13, SC-17, IA-2, SC-23, SC-38, SC-40, SA-18

Fig. 4 An extract of the table of threats collected per each type of component

Starting from the MACM representation of the IoT system under analysis, we are able to automatically build a custom threat model associated with the system

by performing suitable queries to the catalogue. To provide an example, we report in Table 1 a small extract of the results we obtained from the analysis of the case study home automation application (the extract is very small due to an existing non-disclosure agreement with Microbees).

In each raw, we reported the asset to protect (system component), the associated threat along with the related STRIDE category, and the security controls to enforce in order to mitigate the risk of having a threat realized. As said, the threats were identified by taking into account both the type of involved assets and the protocols adopted for communication.

Table 1 Threat and Security Control identification for the MicroBees deployment

Asset	Threat	STRIDE	Security Control
GateBee	Data Leakage	Information Disclosure, Spoofing	IA-3, IA-3(1), SA-18, SC-41, IA-5, SC-8, SI-2, RA-5(1)
Network	Message Modification	Information Disclosure, Spoofing, Tampering	AC-17 , SC-8, IA-2(13), SC-23
IoT Device	Data Leakage	Information Disclosure, Spoofing	IA-3, IA-3(1), SA-18, SC-41, IA-5, SC-8, SI-2, RA-5(1)
Service	Compromised	Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service	IA-9, SA-18, AC-2, AC-1, AC-7, AC-9, IA-5, SC-8, IA-5(1), SI-2, RA-5(1)

6 Related Work

The problems highlighted by the recent breaches mentioned in this paper have boosted the search of manufacturers and researchers for reliable and secure architectures of IoT devices and networks. Unfortunately, nowadays the picture is far from complete and a lot of further work will be necessary. As a matter of fact, the term IoT covers many different technologies and various application domains, and a single reference architecture is likely to be not adequate for all conceivable environments and applications. As a consequence, there is a great variety of different solutions, and the terms adopted vary from one technological solution to the other. The problem of the use of architecture standards for the industrial Internet and connectivity in the IoT is discussed in the paper [18].

Among the open IoT architectures it is worth mentioning the Industrial Internet Reference Architecture (IIRA), Internet of Things Architecture (IoT-A), the Standard for an Architectural Framework for the Internet of Things (IoT), advanced by the IEEE P2413 WG, the ETSI High Level Architecture for M2M, and the ISO

Internet of Things Reference Architecture (IoT RA - ISO/IEC WD 30141) [6]. In addition, standardization efforts have been published in the form of white-papers by main vendors (e.g., Microsoft, SAP, Intel). Similarly, in the academic world, a few survey papers [2], [19] have proposed definitions of IoT systems and outlined the main research issues. Not all these architecture proposals include security considerations. When it is present, security spreads across multiple architectural layers, and this is a very weak model, as pointed out in [10].

As regards the literature centered on IoT system security, Alaba *et al.* [1] propose an IoT security taxonomy that takes into account application, architecture, and communication. The paper also proposes a set of typical threats and vulnerabilities of the IoT heterogeneous environment and proposes possible solutions for improving the IoT security architecture. Zarpelao [20], instead, surveys the intrusion detection techniques useful in the IoT context, pointing out the difficulties of the adoption of such strategies for low power and performance devices.

The papers [17] and [15] outline IoT security challenges in multiple security domains (e.g., authentication, access control, privacy, etc.) proposing an interesting overview of security threats in IoT. Finally, The paper [14] proposes a systematic view of IoT, identifying the main elements together with their interactions and the main actors together with their relationships in the IoT context. Then, the security challenges in respect for each element and actor identified are pointed out.

The risk analysis approach presented in this paper is original, in that nothing similar has never been pursued in the literature. A notable exception is the work presented in [7], which follows a technique with some point sin common with the one presented in this paper, as it relies on the use of graph and graph databases to evaluate a risk profile of a system configuration. The main difference is that Lewis uses simple empirical risk metrics and threshold values, while our method relies on a catalogue gathering information about threats and vulnerabilities.

7 Conclusions and Future Work

In this paper, we introduced a methodology devoted to automating, as much as possible, the threat model definition and risk analysis execution for IoT systems. Our approach relies upon the definition of a model of the system under analysis that is compliant with state of art IoT standards, and on the execution of an almost fully automated process that enables to identify the threats affecting system assets and involved communication protocols, to evaluate related risk, and to identify the countermeasures that should be applied in order to mitigate existing risk. In future works, we plan to extend the technique in order to support automated security assessment of an IoT system, by taking into account what each component is able to provide and by evaluating, in an automated way, if the introduction of a new (possibly faulty) component may affect the security of other assets of the system. Moreover, we plan to adopt the framework and solutions proposed in [3] in order to automate the penetration testing of such systems.

Acknowledgements This authors would like to thank Lorenzo Russo and Maria Teresa Diana for their valuable contribution in the validation of the methodology.

References

1. Alaba, F.A., Othman, M., Hashem, I.A.T., Alotaibi, F.: Internet of Things security: A survey. Journal of Network and Computer Applications **88**(December 2016), 10–28 (jun 2017)
2. Borgia, E.: The internet of things vision: Key features, applications and open issues. Computer Communications **54**, 1–31 (2014). https://doi.org/10.1016/j.comcom.2014.09.008
3. Casola, V., Benedictis, A.D., Rak, M., Villano, U.: Towards automated penetration testing for cloud applications. In: 2018 IEEE 27th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE). pp. 24–29 (June 2018)
4. Casola, V., Esposito, M., Mazzocca, N., Flammini, F.: Freight train monitoring: A case-study for the pshield project. Proceedings - 6th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, IMIS 2012 pp. 597–602 (2012)
5. Guo, J., Chen, I.R., Tsai, J.J.: A survey of trust computation models for service management in internet of things systems. Computer Communications **97**, 1–14 (2017). https://doi.org/10.1016/j.comcom.2016.10.012
6. ISO: Internet of Things Reference Architecture (IoT RA) **ISO/IEC CD 30141** (2016)
7. Lewis, M.: Using graph databases to assess the security of thingernets based on the thingabilities and thingertivity of things. In: Living in the Internet of Things: Cybersecurity of the IoT - 2018. pp. 8 (9 pp.)–8 (9 pp.). IET (2018). https://doi.org/10.1049/cp.2018.0008
8. MicroBees: The MicroBees web site (2018), https://www.microbees.com/
9. Microsoft Corporation: The STRIDE Threat Model (2016), https://docs.microsoft.com/en-us/previous-versions/commerce-server/ee823878(v=cs.20)
10. Minoli, D., Sohraby, K., Kouns, J.: IoT security (IoTSec) considerations, requirements, and architectures. In: 2017 14th IEEE Annual Consumer Communications & Networking Conference (CCNC). pp. 1006–1007. IEEE (jan 2017). https://doi.org/10.1109/CCNC.2017.7983271
11. National Institute of Standards and Technology: SP 800-53 Rev 4: Recommended Security and Privacy Controls for Federal Information Systems and Organizations. Tech. rep. (2013)
12. OWASP: The OWASP Risk Rating Methodology Wiki Page (2016), https://www.owasp.org/index.php/OWASP_Risk_Rating_Methodology
13. Rak, M.: Security assurance of (multi-)cloud application with security SLA composition. Lecture Notes in Computer Science **10232**, 786–799 (2017)
14. Riahi Sfar, A., Natalizio, E., Challal, Y., Chtourou, Z.: A roadmap for security challenges in the Internet of Things. Digital Communications and Networks **4**(2), 118–137 (2018). https://doi.org/10.1016/j.dcan.2017.04.003
15. Roman, R., Zhou, J., Lopez, J.: On the features and challenges of security and privacy in distributed internet of things. Computer Networks **57**(10), 2266–2279 (2013)
16. Schiefer, M.: Smart home definition and security threats. In: 2015 Ninth International Conference on IT Security Incident Management IT Forensics. pp. 114–118 (2015)
17. Sicari, S., Rizzardi, A., Grieco, L., Coen-Porisini, A.: Security, privacy and trust in Internet of Things: The road ahead. Computer Networks **76**, 146–164 (jan 2015)
18. Weyrich, M., Ebert, C.: Reference architectures for the internet of things. IEEE Software **33**(1), 112–116 (2016). https://doi.org/10.1109/MS.2016.20
19. Xu, L.D., He, W., Li, S.: Internet of Things in Industries: A Survey. IEEE Transactions on Industrial Informatics **10**(4), 2233–2243 (nov 2014). https://doi.org/10.1109/TII.2014.2300753
20. Zarpelão, B.B., Miani, R.S., Kawakani, C.T., de Alvarenga, S.C.: A survey of intrusion detection in Internet of Things. Journal of Network and Computer Applications **84**, 25–37 (2017). https://doi.org/10.1016/j.jnca.2017.02.009