Chapter 7
Security Challenges and Concerns of Internet of Things (IoT)

Aniruddha Bhattacharjya, Xiaofeng Zhong, Jing Wang, and Xing Li

Abstract The Internet of Things (IoT) signifies the interconnection of exceedingly heterogeneous networked entities, for instance, sensors, actuators, smart phones, etc. In accord with concrete functions, the network structure of the IoT is divided into three hierarchies: the bottom hierarchy is the sensing equipment for information acquisition; the middle hierarchy is the network for data transmission, whereas the top hierarchy is intended for applications and middleware. The uniqueness of the IoT proclaims new challenges to security requirements, dissimilar from previous technology trends. Moreover, to guarantee resilience, fail-over and recovery mechanisms must be provided to uphold operations under failure or attacks, and to return to normal operations (failure/attack mitigation). To uphold the end-to-end method, the gateway requirements to endure invisible to the communicating endpoints. The Constrained Application Protocol (CoAP) is an ideal protocol, for being used with constrained devices and low-power networking. To give more security, to the major UDP (User Datagram Protocol) well-known applications, for instance, Voice over IP/Session Initiation Protocol (VoIP/SIP), Datagram Transport Layer Security (DTLS) can run on top of UDP instead of TCP (Transmission Control Protocol). In our research, we have found that hybrid RSA (Rivest–Shamir–Adleman) algorithm can be a good one with efficiency, more security, and more privacy protected way and can work for end-to-end encryption requirements for future Internet of Everything (IoE). In general, future researches in the security issues of the IoT would mostly quintessence on the following characteristics, the open security system, individual privacy protection mode, terminal security function, related laws for the security of the IoT, etc. It is unquestionable that the security of the IoT prerequisites a series of policies, laws, and regulations, perfect security management system for mutual collocation.

A. Bhattacharjya (✉) · X. Zhong · J. Wang · X. Li (✉)
Beijing National Research Center for Information Science and Technology, Department of Electronic Engineering, Tsinghua University, Beijing, China
e-mail: li-an15@mails.tsinghua.edu.cn; zhongxf@tsinghua.edu.cn; wangj@tsinghua.edu.cn; xing@cernet.edu.cn

S. Guo, D. Zeng (eds.), *Cyber-Physical Systems: Architecture, Security and Application*, EAI/Springer Innovations in Communication and Computing,
https://doi.org/10.1007/978-3-319-92564-6_7

7.1 Introduction

The Internet of Things (IoT) signifies [1–20] the interconnection of exceedingly heterogeneous networked entities, for instance, sensors, actuators, smart phones, etc. In accord with concrete functions, the network structure of the IoT is divided into three hierarchies: the bottom hierarchy is the sensing equipment for information acquisition; the middle hierarchy is the network for data transmission, whereas the top hierarchy is intended for applications and middleware. The IPv6 and web services as major building blocks for IoT applications have formed a homogeneous protocol ecosystem, letting simple integration of IoT devices in a Low-power and Lossy Network (LLN) with Internet hosts. The uniqueness of the IoT proclaims new challenges to security requirements dissimilar from previous technology trends. Moreover, to guarantee resilience, fail-over and recovery, mechanisms must be provided to uphold operations under failure or attacks, and to return to normal operations (failure/attack mitigation). We can choose Datagram Transport Layer Security (DTLS) as our security protocol that depends on this protocol stack. Alike the security needs in traditional networks, such as the Internet, we can think about three security goals for IoT scenario [1–20]:

Authenticity: Receivers of a message can recognize their communication companions and can identify if the sender information has been forged.
Integrity: Communication companions can identify modifications to a message for the duration of transmission.
Confidentiality: Attackers cannot get information about the matters of a secured message.

DTLS fulfills these goals. The authentication is accomplished during a fully authenticated DTLS handshake and depends on an exchange of X.509 certificates comprising Rivest–Shamir–Adleman (RSA) keys. An unconstrained network (UCN) is classically signified by the Internet, while the loT comprising of a low-power wireless personal area network (LoWPAN) signifies the constrained domain. An loT gateway placed on the edge among the constrained network (CN) and the UCN adapts the communication among these two domains. Its role typically encompasses the adaptation between dissimilar protocol layer implementations. Also called a border router, it carries out protocol translations vis-a-vis end-to-end loT security. The gateway is usually an unconstrained device, which can be used for scaling down the functionalities from the UCN to the CN domain. The gateway can be used for handling security settings in peripheral constrained networks. To uphold the end-to-end method, the gateway requirements to endure invisible to the communicating endpoints. A node on the UCN can be either Hypertext Transfer Protocol (HTTP) enabled or only Constrained Application Protocol (CoAP) enabled. The communication protocols existing or being designed at the IEEE and IETF now empower a standardized protocol stack. The mechanisms founding this stack must thus empower Internet communications encompassing constrained sensing devices, while coping with the necessities of low-energy communication environments and

the aims and the lifetime of IoT applications [21–40]. In order to talk this issue for the IoT, the IETF has started the Constrained RESTful Environments (CoRE) working group, which aims at standardizing the incorporation of constrained devices with the Internet at service level. The CoRE proposal aims to permit the integration of constrained devices with the Internet, at service level. CoRE proposes the use of CoAP in constrained devices, a specialized RESTful Web transfer protocol. CoAP is a specialized web transfer protocol aimed to be used by constrained devices in loT machine-to-machine (M2M) applications. It is responsible for a client/server interaction model between application endpoints and comprises the same key functionalities of HTTP. So, CoAP can be easily interfaced with HTTP, resulting the web integration simplified while also guaranteeing M2M critical necessities, for example, built-in discovery, simplicity, multicast support, and low overhead. Yet, application layer protocols recurrently delegate security techniques to the transport layer, which benefits in attaining end-to-end security. The overhead caused by this security mechanism is very significant to the overall system performance. One such protocol is DTLS, which furthermore has inbuilt binding within CoAP. Security is fundamental for the application areas. We should take care of the basic security services, for example, confidentiality, authentication, and freshness of secret keys between two communicating entities. Information exchanged in the network requisite to be protected end-to-end. To cope with these security necessities, CoAP offers DTLS and when DTLS NoSec mode is selected, the CoAP communication could be secured using IP Security (IPSec) at the network layer in an LLN. Nevertheless, DTLS was not intended for lossy networks and constrained devices, it has appeared as a vital candidate to deliver security in loT. Nevertheless, it cannot be employed as it is, ever since it is well-thought-out to be too heavy for using in constrained environments and networks such as loT. Thus emerged numerous lightweight implementations of DTLS are there now for use in loT. Lightweight DTLS implementation could depend on employing any of the following techniques:

1. Pre-shared key (PSK)
2. Raw public key
3. Certificates

The *CoAP protocol* defines bindings to *DTLS (Datagram Transport Layer Security)* to secure CoAP messages, together with a few mandatory minimal configurations suitable for constrained environments. The acceptance of DTLS implies that security is reinforced at the transport layer, rather than being designed in the context of the application layer protocol. DTLS provides promises in terms of confidentiality, integrity, authentication, and non-repudiation for application layer communications using CoAP.

In the last section of this chapter, we have highlighted some case studies and open research issues.

7.2 Internet of Things Architectures, Properties, and Security Requirements

7.2.1 Architectures and Basic Properties

With the contextual features of Internet, the IoT [29–33, 35–45] is an emergent technology uniting EPC standard, wireless communications technology, Radio Frequency Identification (RFID) technology, and so on, to empower human to commendably solve various defies of modern society. IoT brings together and processes detailed information consisting of events and environments, by use of billions of connected things, making our life more comfortable, more productive, safer, and healthier. To explore IoT's hidden prospectives, to address many global complications, for example, energy scarcity, pollution, food, climate change, and water, along with the challenges of transportation, urbanization, and healthcare, the International Telecommunication Union (ITU) is making the IoT standardized for several years in the Telecommunication Standardization Sector (ITU-T). *ITU-T Study Group 20* was formed in recent times, to further endorse coordinated advancement of global IoT technologies, services, and applications. M2M communication technologies deliver an efficient, reliable, and secure communication platform for the almost all of 50 billion IoT devices, that are anticipated to be linked to the succeeding generation (recognized as 5G) mobile networks in 2020 and beyond. IoT is not only a confined infrastructure for interconnecting things only inside a locality (e.g., a building, an enterprise, or a city), but a global infrastructure to connect things by use of interoperable underlying communication networks. "Overview of the Internet of Things" (ITU-T Y.2060), ratified in 2012, has been enriched by a number of recommendations on IoT common framework, capabilities, use cases, and necessities. In ITU-T Y.2060, the thing has been well defined as an object of the physical world (physical thing) or the information world (virtual thing), which can be totally able to be identified and integrated into communication networks. Figure 7.1 depicts the IoT reference model detailed in ITU-T Y.2060. This layered reference model shows us a generic and universal model, with the critical functions and abilities of the IoT architecture. It has a great advantage of decreasing the implementation difficulties and endorses interoperability amid numerous IoT applications and communication technologies. The IoT reference model comprises of four horizontal layers and the common management and security capabilities allied with all layers. The application layer is the topmost layer that comprises numerous IoT applications, e.g., smart grid, intelligent transport systems, e-health, and smart home. The service and application support layer is the second layer, which comprises generic support capabilities along with application-specific support capabilities. The generic support capabilities are common abilities relevant to many applications, while the application-specific capabilities work for a particular application's necessities as their names denote. The network layer comprises the networking and transport capabilities. The networking capabilities execute the connection of things to networks and maintain that connectivity. They

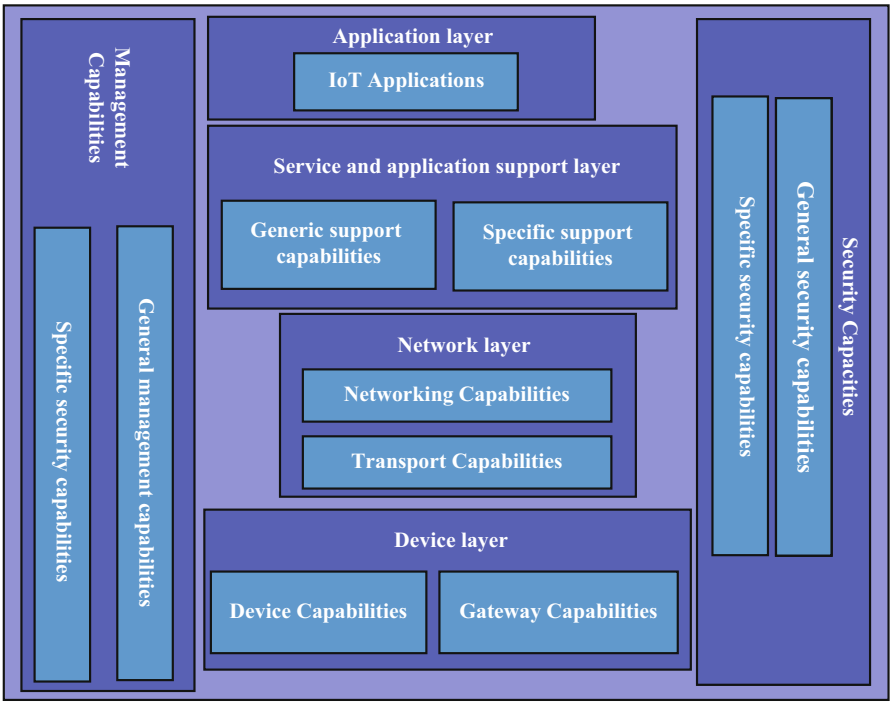

Fig. 7.1 Internet of Things (IoT) reference model

comprise functions for resource allocation, routing, mobility management, access control, etc.

Likewise, the transport capabilities comprise functions for transporting IoT application data along with control and management instructions. The device layer at bottom comprises a collection of device capabilities and gateway capabilities. The device capabilities empower things to interact with a network straight or via a gateway. They are comprised of ubiquitous sensor networking functions. Likewise, the gateway capabilities comprise privacy protection, security, and protocol translation functions to allow resource-constrained IoT devices empowered with heterogeneous wireless technologies, such as Zigbee, Bluetooth, and WiFi, to be connected securely through a network. Management and security capabilities are also considered as generic and specific capabilities. The generic management capabilities comprise device management functions such as software update, network topology management, status monitoring and control, and traffic, congestion control, and remote activation. The generic security capabilities comprise integrity protection, privacy protection, access control, authentication, confidentiality, and authorization, etc.

The essentials of IoT security [29–33, 35–44] include information sensing with high safety, trustworthy data transfer and information control with high safety. The

security system of the IoT can be classified into three layers, *the Sensor Layer Security, the Network Layer Security, and the Application Layer Security*. *Firstly*, any object in this earth is having connection to the Internet. So, it is well understood that nodes will communicate effortlessly with each other. *Secondly*, sensing at any time anywhere in all place like an all-round sensing, resulting in identification of any object connected in IoT automated, no manual intervention is needed. *The third* is intelligent processing. Intelligence control, self-feedback, and automation, etc., characterize the intelligent processing. This security framework is depicted in Fig. 7.2. In general, we always have to take care about three characteristics in IoT. The *first one* is entirely perception. To make clearer, to gain access to the information of object anywhere at any time by use of various means, like RFID, sensors and two-dimensional code. The *second one* is reliable delivery. To make clearer, it is to send information of the object correctly at real time through incorporating

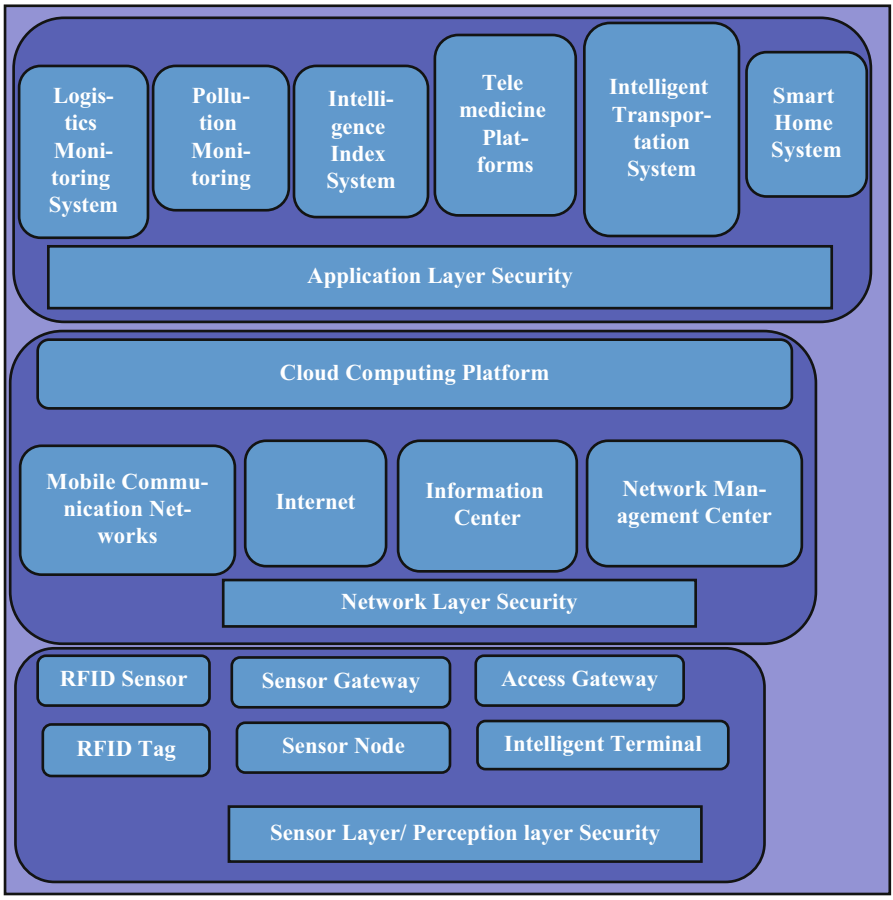

Fig. 7.2 Security framework of IoT

telecommunication network and Internet. The *third one* is intelligent processing, analyzing, and processing massive data and executing intelligent governing power on objects by the usage of cloud computing, fuzzy recognition, and other intelligent computing techniques. The most frontend layer is Sensor Layer or Perception Layer, which is mainly responsible for information collection and so it is well understood that it has one of the most significant roles in the security of IoT.

So, now let us have some highlights on Security issues in sensor layer. If we consider the case of traditional network, sensor nodes in IoT positioned in an unattended environment, there are some new characteristics in sensor network.

1. Wireless link signal strength is very feeble

Sensor nodes spread data to each other primarily by wireless network and most of them can work well in longtime environments and with low-power environment. The disturbing waves usually affect the wireless communication's signal. So, it is obligatory to not to transfer information by wireless network.

2. Node is visible

In the wireless data communication, hidden terminal and exposed terminal problems are most prominent problems, as wireless channel is an open and shared channel. For better understanding, let us consider an example, when we use RFID technology in sensor layer, the object which embedded an RFID chip will be censored not only by its owner but also by others. So this way, we can understand that the sensor node is the best place for all kind of attackers.

3. The network topology is dynamic

Locations of IoT node frequently change from one place to another. In comparison with traditional Transmission Control Protocol (TCP)/IP network, all network monitoring technologies or cyber defense technologies have to deal with more complex network data, more exactingly real-time demand in the scenarios when 50 billion IoT devices will be connected.

4. Computing capacity, storage capacity, and energy are limited

Typically, IoT node is a product of low-power consumption. Most vulnerable issues are that their computing capacity, storage capacity, and energy are limited. So, it is well understood that our present security technologies of traditional network cannot shift to IoT effortlessly.

So, now let us have some highlights on security technology in sensor layer.

1. Encryption mechanism

Point-to-point encryption and end-to-end encryption are two uppermost forms of cryptographic applications in traditional network. From the IoT framework, generally it can be seen that, the node of sensor layer, is low speed CPU, for instance, single chip system. So, for good security, we need to use large storage and high power for Encryption and Decryption but here we cannot use large storage and high power. So, Encryption technique in IoT should be very much lightweight.

2. Access control

Access control mechanism in IoT is very special and differs than normal networks. In our TCP/IP network, a "person" used to give approval to access the system but in IoT, it is "machine." So, it prerequisites to assign and transfer sharing data in a self-determined method between node and node.

3. Authentication mechanism of nodes

At receiver end, the authentication mechanism is used to make sure of the real identity of sender and make sure whether the data is altered for the duration of the transmission. It is very obligatory, for IoT architecture, to make sure that the true node is working, Encryption mechanism can make the data confidential by encoding the data, and it can stop intruder from stealing and altering crucial information by use of data encryption.

In other opinion, sometime we say that the Sensor Layer as the Perception Layer. So, the functions are totally same, it is like another name only. So, the perception layer is primarily responsible to capture and gather, distinguish, and identify objects' information in physical world. The layer consists of laser scanner, cameras, GPS, sensors, RFID tags, literacy device, and so on.

The second layer is the network layer as shown in Fig. 7.2. This layer is used to transmit and process information acquired by the perception layer or sensor layer. Also, this layer is responsible for delivering reliable communication support to the application layer.

The top level is the application layer as shown in Fig. 7.2. This layer is used to process intelligently massive amount of data, data accumulated from numerous sources with various types and interactive display. The layer uses cloud computing, data mining, middleware business management, and so on, for the control and management of objects' information. We have to look for very coordinated association of the information technology and the industry-specific technology for the upcoming and development of the application layer.

Now, at the time of building the security architecture of the IoT, which is used to resolve the security difficulties, now we are facing it from the bottom layer to the top one of the IoT system. Some of the most concerning security problems among the security difficulties are information acquisition security, information transmission security, information processing security, physical security, and so on. So, when we are designing the security architecture, we have to take care about vulnerabilities in every layer. Figure 7.3 depicts one kind of ideal security architecture of the IoT. As explained in Fig. 7.3, the whole system consists of four layers, which work layer-wise. It works for the physical security of terminal equipments positioned in perception layer and local data storage, the protection of wireless transmission of sensor networks, the security of computer networks and mobile communication transmission, and the data service security on application layer.

Ever since the terminal equipment, like RFID tags, being used for identifying entities and all sorts of low-cost sensors, being used to observe objects' status modification or alteration positioned at the perception layer, is mostly restricted by

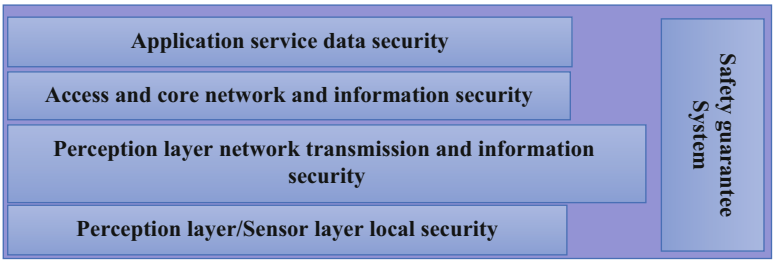

Fig. 7.3 Security architecture of the IoT

the constrained computing resources and mass positioning but with unverified status in positioning environment, those terminal ends are highly in danger to several kinds of attacks.

The network layer security as per its function is categorized into two types, based on the access layer and the core layer transmission. The core network transmission security problem has a complete security protection ability due to its traditional benefits of network information safety. It also has the traditional network security dangers and defenselessness. Moreover, quantitative scale of nodes positioned in the IoT is gigantic, which an attacker can explode very easily to initiate a denial of service (DoS) attack and block network finally. On the contrary, the access layer offers access to heterogeneity and it yields foremost security vulnerability owing to dissimilar media switching technologies and the location management technology. It has wireless or wired multiple access methods. Furthermore, the openness of wireless interface in wireless mobile communication transmission offers malicious individuals with tapping wireless channel, along with that gives chance for capturing even deleting, inserting, retransmitting, and modifying messages communicated through radio interface with the intention of fake user identification or for identification of deceived server. There are severe potential reasons for privacy leakage of information due to different requirements for the same data, the number of systems, multiform data, numerous applications' integration, and various sources in Application Layer of the IoT. The application layer also has another security issue like shielding users' privacy from unsolicited access to personal information, while those users have right of entry to the application service platform for carrying out identity authentication.

The existing security ways and measures for each distinct layer in the IoT are independent of each other, so it is well understood that it is not adequate to offer security assurance for the whole IoT application. One example can be that certification is the identification among different levels in the traditional authentication technique, for that reason, the authentication positioned at the network layer is independent of that to be done at the application layer. So, it is vivid that there is no relationship among the two types of authentications. But, in the IoT environs, business applications and network communications are connected very closely as they work altogether. So, well-understood matter is that it is hard to make them

work independently. So, an example for better understanding is that the security prerequisite of privacy protection is not only dependent on a certain level of the IoT, but in practice it also includes each one of it. Moreover, from a design aspect, the IoT network security architecture does not imply to communicate between devices or any articles. So, in a nutshell, we should replan the security architecture of the IoT. This replanned security architecture should improve the security shielding procedures in the application process and in the development of the IoT.

Designing the security architecture of the Trusting IoT is to facilitate information security protection for data transmission, sensor data security, and tag privacy. Also, another goal is carrying out an intensive systematic research on the transmission and information security of the core network founded on the IoT or networking business security of the IoT. The security architecture of creditable IoT will bring together trusted computing into the architecture to build a chain of trust from the perceived source, and associate the network and service platform, sensor node, with the trust relationship. So, it is well understood that it adds safety techniques in each layer, which is different from the existing network security system. As an outcome, the new architecture could offer the solid theory basis and application guide for the application of the IoT. Also, this architecture is credible and controllable material network architecture, which promotes the networking applications and development.

The essential tactic to make a real-time trusted IoT is to consider three layers as shown in Fig. 7.2. Also, after taking perception/sensor, network, and application layers, for making the Trusting IoT, the resulting most important outcome is the much more improvement in the cyber security. Some examples of this architecture can be as follows.

At first, one of the well-known ciphers, the ECC algorithm can be embedded into tags. The reason is that it will execute the privacy protection to shield the data from modification, usage, duplication, or illegal access. Also, we can use the CPK, which is a known identity-based authentication. It will help us to resolve the mass and fast authentication at the sensor/perception layer.

Second, for the network layer, for providing identification authentication, we can embed the CPK-specific communication chip into the wired or wireless-oriented communication equipment. So, this eliminates the needs of a trusted third party certification. Also along with this, transport code authentication can be used to launch data integrity and confidentiality for communicating data encrypted. Here, the cryptographic power is not less as the key size is not less than 256 bits in the process of data transmission encryption. So, as a nut shell, the above methods can be anticipated to implement an identity-based data transmission encryption. Also, it does not need a third party among entities labeled on identifier.

At last, the trusted access control can be used for the application layer. This trusted access control is used to avoid the illegitimate incursion and safeguard users' unique legitimacy when they log into and necessitate services. Also, this trusted access control can be used also to track main performance, for instance, the operation of business, conforming events for guaranteeing the act of operating non-repudiation and identification of actual operator. And then, to accomplish a

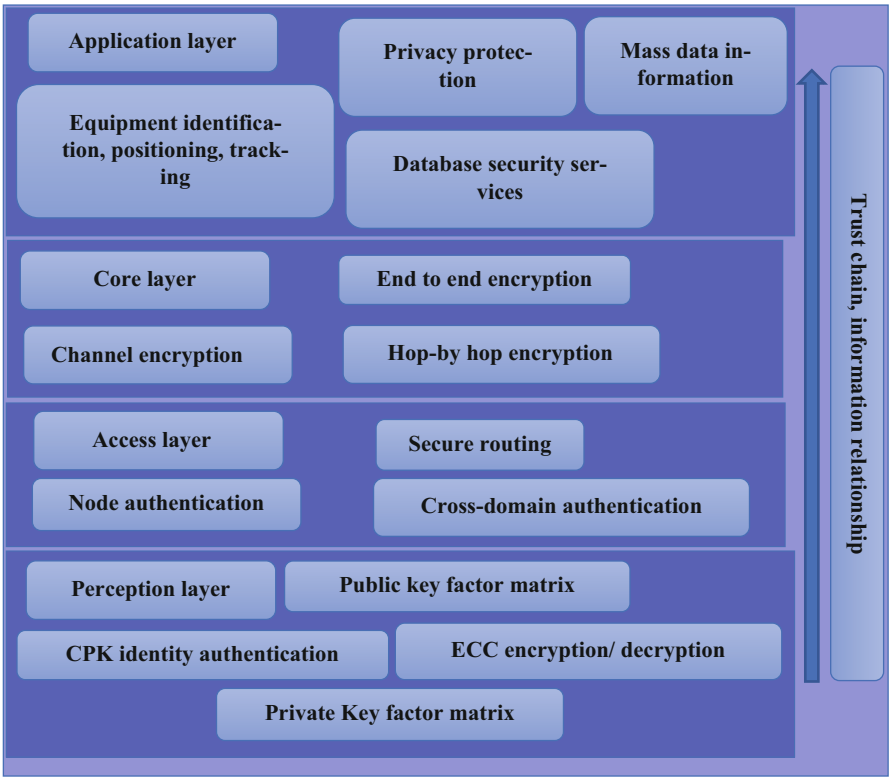

Fig. 7.4 Creditable security architecture of IoT

trusted and safe runtime in open and unsafe network environs, the authentication on code, trusted thread, and process can be accomplished. Furthermore, trusted database can be used to execute data access mutual authentication, for more defense for the network layer. As a result, the creation of whole defense is built to make sure space safety and manageability of the IoT's field. Figure 7.4 has shown the Trusting IoT's security architecture.

7.2.2 Main Security Requirements and Their Sub-Components

If we try to review security requirements from the domain of the IoT, then we have to consider also the correlated areas of IT and their necessities in the context of the properties of the IoT. For that, we can classify the requirements into five groups: *Network Security, Identity Management, Privacy, Trust, and Resilience*. These five key security necessities together with their sub-components are depicted in Fig. 7.5.

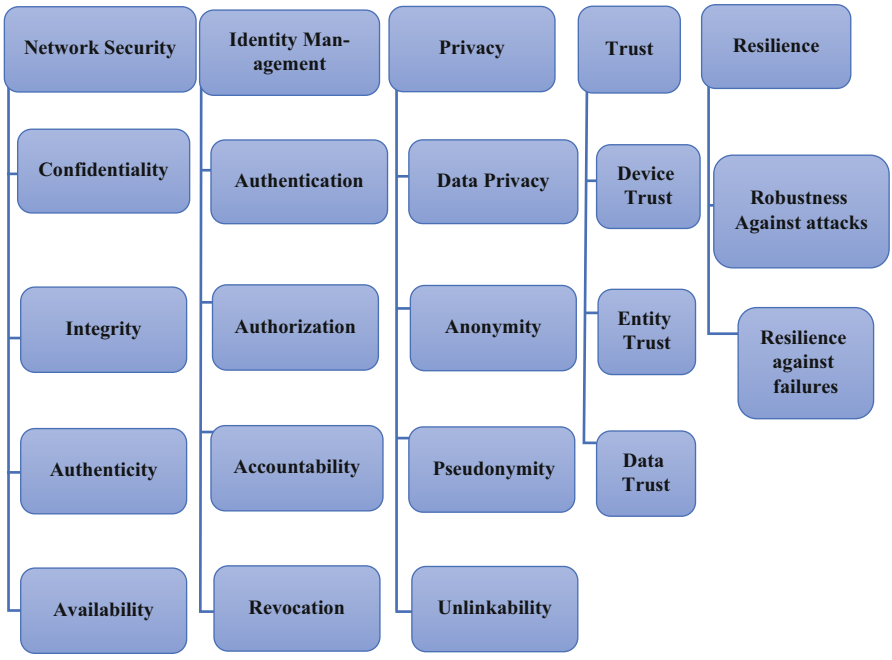

Fig. 7.5 Main security requirements and their sub-components

Table 7.1 IoT properties and security requirements: the "*" symbols represent the level of influence in a scale from one (low) to three (high)

	Network security	Identity management	Privacy	Trust	Resilience
Uncontrolled environment	*	*	*	**	*
Heterogeneity	*	**	*	**	*
Scalability	*	*	**	*	***
Constrained resources	**	*	**	*	*

In addition, Table 7.1 illustrates the association among the numerous IoT properties and the security necessities. It is well understood, for network security, that the constrained resources have the strongest connection. It is for the reason that, mainly due to the constrained resources, there are some restrictions to implement traditional security mechanisms, e.g., cryptography in IoT. Heterogeneity of the IoT mostly has influence on the identity management. Privacy is commonly linked with scalability and the constrained resources, as limitations are posed to the technology candidates that can be utilized. Additionally, the uncontrolled environs and the heterogeneity of the IoT have a big effect on trust. Also, resilience is straightly connected to the need of the IoT for scalability.

Let us now discuss the five requirements in detail as shown in Fig. 7.5.

1. *Network Security*: We can split this requirement into confidentiality, authenticity, integrity, and availability. When we are considering the IoT security architecture, we need the architecture that necessitates the architectures, which deal with the heterogeneity of things. So, it is well understood that interconnecting things may necessitate confidentiality. For example, it should be able to stop eavesdropping the sensitive information via Internet transmission. We already have this for our Internet transmission to fulfill this requirement, for instance, IPSec and Transport Layer Security (TLS). Nevertheless, overhead may exceed the resource constraints of things and therefore dedicated secure network stacks for the IoT exist in this era. We have taken care about authenticity, as it offers evidence that a connection is established with a legitimate entity. Integrity makes sure to detect if any data is lost or modified during transmission. The integrity can be obligatory in the absence of authenticity to detect and recover failures also. But, IoT scenarios need some different, like it may necessitate transactional integrity, like, critical infrastructures, so we can take the architectures as well. Availability makes sure that the connectivity of a thing or service continues, in the scenario of link failures. For that reason, IoT architectures should guarantee that link handover is possible.
2. *Identity Management*: Identity management is really a big challenge in the IoT, as we can have 50 billion devices by 2020 and then another challenge is the complex relationship between users, devices, services, and owners. Henceforth, we have to pay more and more attention to accountability or non-repudiation, authentication, and authorization including revocation. Also, if the abilities of direct authentication to the devices exceed, then user provisioning option should be there, meaning that a user with her/his service credentials can be able to provisioning many devices. Henceforth, ways and means to claim ownership and have control over devices are obligatory. Inside the IoT scenarios, interactions may stretch through numerous domains but our existing authorization solutions, e.g., Kerberos, presume a single domain that enfolds services, owners, users, and devices. Consequently, resolutions for federated authorization that works for non-trusted devices permit the delegation of access through many domains and offer swift revocation. An example can be for broken or rogue devices, it is obligatory. One of the big challenges in IoT is Accountability, for the reason of the magnitude of reuse of data, services, and devices also for many purposes. It makes sure that every action is obviously bound to an authentic entity. Therefore, accountability's obligation is to pact with massive amounts of actions, delegation of access to entities that span continuous derivation of data along with organizational domains.
3. *Privacy*: As the involvement of citizens is increasing day by day in IoT and ubiquitous data collection, e.g., in smart home scenarios, is also increasing day by day, so as an outcome, Privacy is now one of the most dominant challenges in the IoT. Data privacy actually ensures the confidential data transmission. Like a stored data record, it requisites not to uncover undesired properties, for instance, the individual's identity. So, it is very well understood that this requisite is a big challenge in the IoT, due to the reason that many sensing devices have to bring

together personal information. Actually, huge amount of such data turn out to be Personally Identifiable Information (PII), when combined together and this data recognize a person. There are some models which can "anonymize" these kind of data records. But, we have seen that they are insufficient. Addition to that, models to defend this data privacy under data exchange among domains are rather uncharted and complex for implementing it. The property of a single person not being recognizable as the source of data or an action is called as Anonymity. Anonymity is anticipated in the IoT on every occasion, when a persons' identity is not obligatory to fulfill the data minimization laws (Directive 95/46/EG). Along with that, it is anticipated to dismiss preconceptions that arise with data collection in the IoT. It is very hard to attaining anonymity, due to the reason that wearable and mobile devices may disclose PII, for example, IP addresses and location unwittingly. In the present time, we have technologies like anonymous credentials and onion routing, but it may not balance appropriately with the IoT. To trade-off anonymity with accountability, the best tactics should be Pseudonymity. In pseudonymity, actions of a person are allied with a pseudonym, which is nothing but a random identifier, instead of an identity. Pseudonyms can be used in multi-purpose. An example can be connecting several activities of the same person or offering elegant degradation of anonymity for abuse cases. Also, pseudonyms may give resolution for the issues like privacy and accountability concerns in the IoT. Only, standardized resolutions that accompany several domains are obligatory. As definite actions of the same person must not be connected together, so we can say that unlinkability qualifies pseudonymity. Unlinkability defends the profiling in the IoT. Although pseudonyms may resolve unlinkability. One of the examples can be a dissimilar pseudonym being used for each action, cross-implications with anonymity, in specific unidentified meta-data, remain a challenge. In addition, some entity can every time link every pseudonym to a person. So, it is well understood that it can thus also link all activities of that person.

4. *Trust*: One of the crucial prerequisites in the IoT is Trust. The reason is that in reality it is dependable on qualitative data, along with that it is highly distributed also. We can classify the Trust into data trust, entity trust, and device trust. Data trust takes place in the IoT in a dual manner. At first as we know, data come out from several and potentially illegitimate devices. Henceforth, trusted data need to come out from illegitimate sources. It can be done like by applying data aggregation and machine learning techniques. It is well understood that due to the reason that a priori trust in devices cannot at all times be established, so device trust is really a big challenge. It cannot be established due to many reasons, like for high dynamics and cross-domain relations. Henceforth, methods, for instance, trusted computing (for standardized devices) along with computational trust, are obligatory to constitute device trust. Furthermore, every entity may consider trust in a device in a different way. So, as an outcome, IoT architectures have to work with non-singular views of trust. Anticipated behavior of participants, for example, persons or services, is referred as Entity trust in IoT. As we know, device trust can be constituted via trusted computing.

But, planning such methods to introduce into device trust, e.g., via behavioral attestation, is much more challenging and experimental. Another issue is that new data is always obtained from IoT services. One example can be by integrating diverse types of data, we can get new derived data. So, as an outcome, a new trust assessment is obligatory for these newly generated data. Solutions can be in many ways like via computational trust.

5. *Resilience*: One of the most important necessities of IoT is resilience and robustness against attacks and failures. The reason of these attacks and failures are uniting of scale of the IoT in terms of devices. Architectures have to be built up in such a way that it should be able to offer way to adeptly select services according to their robustness (failure/attack avoidance), transmission paths, and things. Moreover, to safeguard fail-over, recovery mechanisms and resilience, the architecture must be able to uphold operations in case of failure or attacks. Also, the architecture should be designed to return to normal operations (failure/attack mitigation).

For end-to-end communication, we have some security solutions at corresponding layers of stack used in that end-to-end communication. So, let us have some highlights on those security solutions.

7.2.2.1 Link Layer: IEEE 802.15.4 Security

The IEEE 802.15.4 protocol is used as link layer in the 6LoWPAN networks. 802.15.4 Link layer security is the current security resolution for the IoT. The node which is being used for communication process needs to be trusted in the link layer. As we know in the link layer, several numbers of nodes along with multiple numbers of hops can be used for communication. A key has to be well defined before the communication starts. This key has a very big role, it is actually always been used for defending all the particular communication going on, in the communication cycle. So, it is well understood that if this key is compromised, then the security of the whole layer is totally gone. Another highlight is that unwanted alteration at individual hop can be discovered by the per-hop security procedure. Data integrity has to be offered for all the hops security measures, with the 6LoWPAN networks. As we know that link layer security has a big disadvantage, it can only provide security in the communication among two adjacent nodes. Still, it is one of the flexible preferences as it can be used with several protocols at any layer, which is above the link layers.

7.2.2.2 IP Security: Network Layer

The IPSec protocol is able to offer security for the network layer. Most relevant thing is that this offers end-to-end security with replay protection, integrity, confidentiality, and authentication. Another advantage is that the IPSec protocol can

be used with several transport layer protocols, for instance, HTTP, User Datagram Protocol (UDP), CoAP, and TCP. IPSec, being a network layer security solution, its security is shared by all the applications, in a running state on a particular device.

7.2.2.3 Security for Transport Layer

IPSec has robustness problem in case of web protocols, and it really lacks robustness. In Transport Layer, TLS or its predecessor Secure Sockets Layer (SSL) is used generally. TLS protocol has solo use over stream-oriented TCP. So, it is well understood that it is not a great technique for wireless communication. The connection-oriented TLS protocol has solo use; it is used in over stream-oriented TCP. But, the problem is that it is not the favored technique of communication for embedded smart objects. Datagram TLS is actually a special protocol, and it is an adaptation of TLS for UDP. DTLS actually can guarantee the end-to-end security of dissimilar applications. It can protect DoS attacks, as it can use the cookies in the web protocol domain. But, again DTLS can be used with the UDP protocols. Thus, it is imperious to make use of the DTLS for offering end-to-end security with IoT.

7.2.2.4 Network Security

As we know that the network is vulnerable to the network attacks, so it is well understood that these attacks can compromise the security. There are many Intrusion Detection Systems (IDS), which are able to detect impostors and malicious happenings in the network. Also, Firewalls are obligatory to block unauthorized access to networks. 6LoWPAN networks of the IoT are susceptible to several attacks from the Internet and from inside the network. So, as a whole it is well understood that it is easier to compromise the wireless domain resource-constrained IoT world than our present regular Internet. So, it is most urgent to develop unique IDS for developing a more complete security for IoT-enabled devices.

7.2.2.5 Data Security in the Internet of Things World

It is well understood that various network security mechanisms make the network communication secure. Our next big issue is how to safeguard the data that IoT devices have stored. We know that the stored data in the IoT devices can be private and sensitive and prerequisites to be secured. IoT world will encompass many tiny nodes which will be resource constrained. So, it is very well understood that the biggest issue is the difficulty to safeguard each of these billions of devices physically or by the use of Trusted Platform Modules (TPMs). Generally, in IoT security, we first take care about setting up of security services at basic level, including authorization, availability authentication, integrity, non-repudiation, and

confidentiality. The present multitude of control protocols for the IoT systems is the Zigbee standard though the security architecture of IoT is in a high progression.

We know that IoT embeds dissimilar kind of sensors into a diversity of goods in reality, so it is very easily perceived that the application of IoT encompasses a lot of private information about users, for example, location, personal information, etc. Now, the reality is that on one side, we presume that the service suppliers to provide the most correct outcomes with our own provided information. On another side, we anticipate that our highly solicited personal privacy can be secured from illegal access. The present-day, privacy shielding procedures consist of space encryption, anonymous space and time, location camouflage, and so on. The role-based access control (RBAC) method in the architecture of IoT defends the security of information to level. But, it is not a complete solution, as it has some insufficiency on identity falsification, information revelation, and other attacks. Another disadvantage of RBAC is that it prerequisites to accumulate huge amounts of information in the database. One good option is Privacy protection based on cryptography. It encompasses homomorphism encryption technique and secure multi-calculation technique and so on. In addition, we prerequisite additional computing resources to add these techniques. Another good result we get after using the K-anonymity technology is that here attackers cannot detect the definite target. But, the disadvantage of this technique is that it does not have any mechanism to protect individual information, as a result, we have to wait for the number of objects attaining K in the group.

Privacy homomorphism was first projected by Rivest in 1978. Many scholars, after that, projected several encryption schemes. But, we have seen that these schemes either only have homomorphism on multiplication (RSA algorithm) or only on addition (IHC algorithm), so these are with limited options of security. But, a very few have homomorphism both on multiplication and addition. But, the biggest problem with these very few algorithms is that we cannot use it in real-time scenario, due to their security flaws. In the past, we have seen that the deterministic privacy homomorphism can be broken in polynomial time. In 2009, a mathematical object based on ideal lattice to understand fully homomorphism algorithm by working together with the encrypted data in this particular way was proposed by Graig Gentry, a researcher from IBM. But, due to the synchronization efficiency improvement, it was not put in real-time use, but it was really a big innovation in fully homomorphism area. Cryptographic community's one of the most enlightened research areas is now homomorphism technology. It permits direct working out on encrypted ciphertext, devoid of decryption and likewise, the result is alike to the ciphertext of the plaintext computation. When we use the privacy homomorphism technology in IoT, a diversity of services are offered to the users, devoid of decrypting users' secret data. So, in a nutshell, a good resolution of personal data security in IoT will be a *personal secrecy protection policy model* that depends on homomorphism encryption. This type of model can advance the efficiency of encryption algorithm. We can enjoy the suitability of the services, although the service providers cannot decrypt the ciphertexts of private information.

Nowadays, IoT extends itself from “anywhere, anyhow, anytime” computing to new extent, we can say the new one as “anything, anyone, any service.” More and more use of IPv6 protocols are there now, for interconnecting present series of computers, along with the smart objects those are in development in the area of Wireless Sensor Networks (WSNs). Most excitingly, merging of IoT-based systems into the kingdom of Internet is going to make a huge change in the direction of future. We can now think that the world with full of IoT-based objects and unified communication is going from end to end through these objects on the IPv6 platforms. So, we know that to make the IoT infrastructure trustable and reliable, the infrastructure needs to be able to offer confidentiality, device and data integrity protection, authentication, privacy protection, transaction auditing, access control, authorization, etc. NFV is a beneficial tool for permitting us to enforce dissimilar levels of security necessities for having a perfect match with the criticality of the services offered in each logically isolated network partition. In the same way, we can use the gateways to impose strict security actions to separate a user-premise network (e.g., a human-body area network (biological sensor networks) used for healthcare) from illegitimate outside domains. Here, it is well understood that resource-constrained user devices are defended by the gateways from illegitimate access. Also, the gateway defends resource-constrained user devices, from being compromised by a mischievous outer entity.

So, in a nutshell, to have a proper secure and privacy protected proliferation of IoT services, we need architectures which are entailed with customized security and privacy levels. These all above literatures give us a wide-ranging overview of many open issues with future directions in the IoT security field. In precise, the secured IoT necessitates compliance with well-defined security and privacy strategies, privacy for users and things, confidentiality, access control, and trustworthiness among devices and users.

7.3 Constrained Application Protocol: Application Layer Connection-Less Lightweight Protocol for the Internet of Things

7.3.1 Constrained Application Protocol

The CoAP is a standard web transfer protocol. This CoAP is an ideal protocol, for being used with constrained devices and low-power networking. For M2M applications, it is an ideal choice. Some of the examples can be smart energy and building automation. The CoAP runs over UDP, resulting in non-reliable message transport. Another highlighting point is that it is not session based, along with that the CoAP can tackle loss or delayed delivery of messages. CoAP offers a request/response communication model among application end points. It also has built-in discovery of services and resources support. The CoAP comprises

Fig. 7.6 Constrained Application Protocol (CoAP) message format

significant conceptions of the Web, such as extensible header options, URIs, and RESTful interaction, etc. CoAP's special ability is that it can effortlessly interface with HTTP for incorporation with the Web, at the same time, meeting specialized necessities, for instance, and simplicity for constrained environments, very low overhead and multicast support. CoAP message structure is shown in Fig. 7.6.

The first byte encompasses the protocol version Ver, a type field T, and TKL. The T is a type field consisting of basic message type information. TKL represents the size in bytes of the Token field. Then, we have the Code field. The Code field encompasses more specific message type information. Then, we have Message ID field. The Message ID field is a unique ID. The work of this unique ID is to track messages and distinguish likely duplications. To match request and response messages, the optional Token field can be used. The value of this Token must be produced at random, and in addition to that, it should be unique for each request. The field varieties are in between 0 and 8 bytes in size. These varieties of field are actually for making CoAP more robust to battle the IP-spoofing attacks. We should use this just in case security is not offered at the transport layer. Moreover, more than a few dissimilar CoAP options have been well defined. Now, it is possible to state a list of them in line with a Type–Length–Content scheme. At last part of the structure of CoAP message, it has the Payload field. As we know, the IETF CoRE working group has projected the CoAP as a new application-level protocol for constrained devices. But, astonishingly, the CoAP has no security measures, but nowadays, research works have projected positioning the DTLS or IPSec protocols to offer a secure CoAP.

7.3.2 Constrained Application Protocol–IP Security

We know that IPSec is a layer three protocol. It is ideal for use with IPv6, but later stage, it is now can be used for IPv4. It can protect application and transport layers' applications but good thing is that it is not an application-dependent protocol.

The reason for this independence is that the IPSec is integrated into the kernel, resulting in transparency to the applications. For the reason of this transparency, TLS and Secure/Multipurpose Internet Mail Extensions (S/MIME) [RFC 3851] can be used by IPSec. The IPSec can offer various security services like: Limited Traffic Flow Confidentiality, Anti-Replay mechanism, Access Control, Confidentiality, Connectionless Integrity, and Data origin Authentication. One way to use IPSec, to secure the CoAP transactions, can be Encapsulating Security Payload Protocol [RFC 2406] (IPSec-ESP). It can be a special case, if the hardware provisions encryption at layer 2 (it is the situation with some IEEE 802.15.4 radio chips). Another way can be the 6LowPAN extension, for using the IPSec with AH [RFC 2402] or ESP.

There are some issues with IPSec. *First* point is that basically the IPSec and DTLS were not considered for the constrained environs. At that time, the constraints were not considered in the IPSec/DTLS designs. *Second* point is that IPSec has been identified with problems for making use of Network Address Translation (NAT) and/or Port Address Translation (PAT). *Third* point is that performance of the network gets worse when communicating small packets, as the encryption procedure of IPSec produces a large overhead. *Fourth* point is that security association (SA) has an issue in IoTs, i.e., the mobility. The Security Parameter Index (SPI), Destination IP Address, and Security Protocol Identifier identify the SA, uniquely. Now, in this case, the issue is that if a node alters its IP address afterward the formation of the SA, then new SA prerequisites to be formed, which will give unnecessary performance degradation. *Fifth* point is that IPSec is inserted in the IP stack, so any alterations will have the need of kernel level. *Sixth* point is that Configuring/Managing/Troubleshooting IPSec and Internet Key Exchange (IKE) are very much composite tasks. It is well understood that an enormous number of constrained devices are taking part in the network. Any wrong configuration of security parameters of IPSec could give security holes or performance problems. *Seventh* point is that every scenarios/nodes cannot be supported by IPSec. Simply to understand, the support of IPSec for multicast communication is problematic. Last but not the least, as per the CoAP's draft, it is promising to use IPSec (ESP) with layer-2 encryption hardware. It provisions the use of AES-CBC (128-bit keys).

A comparison of IPSec and DTLS in various security dimensions is described in Table 7.2.

Also apart from the above issues, the DTLS and IPSec are not the most enhanced resolutions, to offer proper protection to CoAP for many reasons. The reasons are, at *first*, IPSec and DTLS necessitate extra messages, to work for the security parameters and form the security associations (SAs). But, the overhead and drain out of the resources of the constrained devices will be increased much more. This is very problematic for the mobile types of the devices in the IoTs, as new AS prerequisites to form every time the device in mobility. The *Second point* is that if we think about the environs of the communication among two dissimilar networks, the ideal security resolution is dependent on either IPSec or DTLS, which point towards the existence and provision of these protocols, in both the source and destination networks. But, this ideal idea cannot be realistic in many circumstances, particularly

Table 7.2 A comparison of IPSec and DTLS in various security dimensions

Security dimension	IPSec	DTLS
Access control	No	No
Authentication	Yes	Partially server only
Non-repudiation	Yes/No, as per the authentication method. PKI not supported by constrained devices	Yes/No, as per the authentication method. PKI not supported by constrained devices
Confidentiality	Yes	Yes
Communication security	Yes	Yes
Integrity	Yes	Yes
Availability	Mitigation—no full defend	Yes—stateless cookie
Privacy	No	No

when we think about the fact that the IPSec protocol has a compatibility problem with firewalls throughout the networks. *Third* issue is that both IPSec and DTLS count on the IKE and the Extensible Authentication Protocol (EAP), for setting up the secure association and sometimes any other. So, it is well understood that this points towards that all constrained devices' vendors requisite to support these additional protocols (IKE and EAP). *Fourth* point is that the IPSec and DTLS are aimed at securing connections among two static and remote devices. So, the IPSec and DTLS attempt to offer the most possible secure connection among the two ends, devoid of the QoS, the network trustworthiness, or any other restrictions on the end devices' considerations. But, in the environs of the constrained environment, there is a need for more dynamic and sensible actions that think about the constrained type of the end devices at the time of negotiating the security parameters. The *fifth* point is that the IEEE 802.15.4 specification describes that the payload should be 127 bytes as whole. So, if we use the DTLS as security protocol, to defend CoAP exchanges, 13 bytes (out of the 127 bytes of IEEE 802.15.4 frame) has to be assigned for DTLS record. Also, 25 bytes has to be used for link layer addressing information, and 10 bytes for 6LowPAN addressing; along with that 4 bytes for CoAP header. So, as an outcome, only 75 bytes are available, for application layer payload. But, it is not sufficient space for communicating the actual data. Subsequently, one big piece of data (bigger than 75 bytes) will use additional resources from the nodes and the network itself. The reason is that it will be broken into several pieces and will be sent twice. Hence, some header compression mechanisms are good solutions, at the exact cases where needed. The compressing and decompressing necessities are the reason, for more constraints to the nodes and network resources. The *Sixth* point is, in the case of DTLS, that some applications might necessitate security services to be more and more customized in relation to the applications' or scenarios' requirements. Nevertheless, if the security were applied as per the requirements of the application or scenario, it would offer to decrease the usage of resources existing and definitely would increase the network enactment. *Last but not the least*, in the Internet draft of "Datagram *Transport Layer Security in Constrained Environments*," the authors point out seven prospective problems, correlated to

DTLS protocol, if employed in constrained environs. The authors also have pointed out some projected workaround, to resolve these problems. Still, much of work is required, to make the DTLS perfect for making it a good and prospective security resolution for IoTs.

The *Secure CoAP (S-CoAP)* is a secure variant of CoAP. In S-CoAP, the security technique is actually an integrated part of the protocol itself. With S-CoAP, security measures will be integrated into the plain CoAP transactions. So, one of the good features is that it will have its own compromise stage that thinks through the limits of the constrained devices. The S-CoAP prerequisites to offer security for normal connection setup, in addition to that, for the case of mobility also. So, in a nutshell, the advantage is that the security will be an integral part of the CoAP protocol. It is well understood that this security is offered by other standards, so the S-CoAP should be capable to function across numerous sites and networks.

7.4 Datagram Transport Layer Security Overview and Supporting Constrained Application Protocol

7.4.1 Datagram Transport Layer Security Protocol

The DTLS protocol is UDP based. The DTLS comprises of four protocols: the Handshake protocol, Alert protocol, the Change Cipher Spec protocol, and the Record protocol. The DTLS protocol offers message fragmentation at the Handshake layer. This enables the DTLS to get rid of message fragmentation in the network layer. These fragmented packets bring many problems, like data loss rate increases and unnecessary delays made by packet retransmission. So, it results in worse LLN conditions. The main burden to a memory-constrained device is to reunite a fragmented message packet, due to the reason that devices have to retain fragmented pieces of the message in the buffer unless until all the pieces reach. To resolve these issues, the DTLS In Constrained Environments (DICE) standard WG was shaped. Nevertheless, definite solutions have not been projected yet. So, it is a well-known thing that to decrease the load on memory of the devices used in making IoT environs, *lightweight DTLS* was projected. *Lightweight DTLS* is able to decrease the DTLS code size, for decreasing the burden on constrained memory of a device. Another way to reduce the load can be by decreasing the transmitted message size by compressing the DTLS header.

The CoRE WG projected TLS_PSK_WITH_AES_128_CCM_8, as a basic cipher suite of DTLS to decrease difficulties like packet fragmentation and loss and delay in an LLN. But, here we have one limitation. Here, the PSK is a necessary thing, due to the reason that if it is not there then the devices cannot make use of this cipher suite. To resolve this issue, Gerdes and Bergmann projected a system, in which a ticket is generated. After executing a DTLS handshake among delegators, each delegator produces a ticket. The CoAP server and the CoAP client execute

the DTLS handshake using the ticket. A PSK is encompassed in the ticket. So this way, key circulation is made likely to form PSK-based DTLS channel among nodes. Here, the security policy has not been determined in advance. To have the network efficiency, we can decrease added header data, due to message fragmentation. So this way, we can decrease the packet loss rate and delay. URI based on a CoAP communication environment having a RESTful structure is a good practical approach. Let us now discuss some of the issues about attacks on the above kind of system.

7.4.1.1 Secure Service Manager Spoofing Attack

If an attacker is the secure service manager (SSM), then the most dangerous thing is that the attacker can acquire all the information about the session, due to the reason of delegating the DTLS handshake. So, there is a chance that the encrypted data among end nodes can be exposed to the attacker. A good solution can be the use of PSK_DN (which is shared among the SSM and a constrained device in the bootstrapping phase). This is a perfect solution for protecting from SSM spoofing attack. The good reason for this protection of the SSM spoofing attack is that data is encrypted by use of PSK_DN and then sent, and the attacker cannot deceive a constrained device and cannot get the right to use the encrypted data.

7.4.1.2 Semi End-to-End Security

We have to ensure end-to-end security. The SSM can acquire all session information, by just delegating the DTLS handshake. As we know, the encrypted session information is sent to a constrained device instantly, but the SSM does not do the accumulation of session information. So, it is well understood that end nodes joining in the DTLS communication will encrypt and decrypt data themselves only. The SSM is only responsible for the data relay after sending the session information to the constrained device. In this kind of system, the executor of the encryption and decryption is the end node, in the DTLS communication. There is one obligatory thing: the SSM must trust the preregistered device, for example, smart phone of user. So, as an outcome, we can get an end-to-end security (semi end-to-end security exactly) definitely.

7.4.1.3 Denial of Service

The devices setting up IoT have low CPU performance and a small amount of memory. So, it is a well understood fact that sending a DTLS handshake request message to these low-memory and low-performance devices can seem to be a DoS attack, even supposing that the request is from a legitimate user. Another case is if an attacker transmits a DTLS handshake message straight to a constrained device with

conditions in the LLN, then as an outcome, the devices become more dangerous. So, we can understand that the SSM benefits to resolve the DoS issue by delegating the DTLS handshake. The SSM stops constrained devices from receiving a lot of messages directly.

7.4.1.4 Single Point of Failure

Numerous methodologies applying delegation can give a single point of failure (SPOF). It is one of the utmost predictable, but serious difficulties in security field. We know that the SSM has a significant role of delegating DTLS handshake in place of numerous CoAP sensors. So, it is well understood that if the SSM is negotiated or fails, then all the sensors under the SSM cannot create a secure session with client or server, which are outer of the LLN.

A well-defined trust manager can somehow protect such an SPOF issue. The trust manager has the option to choose alternative authentic device, as a new SSM. Then, he/she can broadcast associated information to his sensors. Only thing is that the SSM should be a resource rich device in smart home or smart building (e.g., smart healthcare devices, etc.). Another way can be virtually applied SSM in cloud system. It is harder to compromise a virtual SSM in Cloud, as it is operated and supervised by security manager, compared to attack a home device or smart phone, which is operated by its usual user. One highlighting point is that here, a secure registration method between the SSM and IoT devices controlled by the SSM is there. Moreover, another supposition is that the secret key, which is common for both SSM and its devices, cannot be compromised. Future research can be on designing and implementing a concrete secure system, with additional mechanisms including key revocation, secure bootstrapping, trust management, and so on.

7.4.1.5 Fragmentation Attacks

A packet fragmentation mechanism is a good resolution for dissimilar MTU size among Internet and LLN. An IPv6 adaptation layer, 6LowPAN, has a provision with a method to fragment large IPv6 packets into small frame. Normally, sensing data and control data for actuators can be small in size. Though, DTLS handshake message is bigger in size than the maximum frame size of LLN, for instance, IEEE 802.15.4 (i.e., 127 bytes). Particularly, DTLS fragmentation is unavoidable at the fourth flight of DTLS handshake. The reason is that it encompasses comparatively large size of certificate of server and key exchange message. We can send 27 DTLS fragmented datagrams in case of using TLS_ECDHE_ECDSA_WITH_AES_128_CCM_8 with Raw Public Key Certificate. Significant transmission overhead is the outcome from these fragmented datagrams for the reason that the header is added to each of the frames. But, some other critical issues are that, due to the deficiency in authentication mechanism at 6LowPAN layer, it gives chance for attackers to try buffer reservation attack,

fragment duplication attack, and fragmentation attacks. An attacker eavesdrops and modifies a fragmented frame in the middle of the wireless multi-hop link, to lunch the fragment duplication attack. At the time of receiving, the Target node cannot identify the altered frame. So, as an outcome, the attacker's just a single forged frame can stop successful reassemble execution of the target node. Additionally, the target node requisites to abandon all frames in the buffer and awaits for retransmission once more, resulting in the DoS attack. We know that the first frame retains a memory space for reassembling the original packet and it is indicated in the header (i.e., datagram size field) at the target node. Also, the buffer reservation attack exploits this fact. The attack can be very simple, like the attack can be done by sending a forged start frame encompassing large number in the datagram size field.

A good option with a good efficiency can be a scheme, which uses the SSM to delegate the DTLS handshake phase. For the constrained network like an LLN, network overhead, and delay and loss problems, due to fragmented handshake message packets, are resolved by delegating the handshake. For the constrained device, the device need not retain the fragmented handshake packets, in the buffer until receiving all of them. In addition, DTLS communication devoid of any source code for a DTLS handshake can be used by a constrained device. Here, the end-to-end security is definite, as data encryption and decryption are done in the end node. Also, its more important feature is that the system can tackle an SSM spoofing attack and DoS attacks on a constrained device. Another highlight is that the SSM and the constrained device are tangibly distinct but can virtually be considered as one system in a trusted relation with a shared key. This shared key is a pre-shared. So, in a nutshell, this kind of scheme can benefit for deploying constrained devices in a secure manner in constrained environments.

7.4.2 Supporting Constrained Application Protocol

The DTLS protocol is nothing but an improved type of the very popular TLS protocol [RFC 5246]. To give more security, to the major UDP well-known applications, for instance, Voice over IP/Session Initiation Protocol (VoIP/SIP), DTLS runs on top of UDP instead of TCP. This is a key difference. The DTLS offers automatic key management, confidentiality, authentication, and data integrity. It also provisions wide range of dissimilar cryptographic algorithms. As per the CoAP's draft, CoAP describes four security modes with the intention of achieving the security services, which is obligatory. They are: NoSec, PreSharedKey, RawPublicKey, and Certificate. In case of NoSec mode, the packets are transferred usually as UDP datagrams over IP. The CoAP scheme indicated this as coap://. In case of all other three security modes, security is attained by DTLS and the scheme is indicated by coaps://.

Now, let us discuss some issues of the *DTLS supporting the CoAP*. At *first*, multicast communication is not offered by DTLS protocol, but it is an essential

part of CoAP protocol and main feature in IoTs. *Second* thing is that the DTLS handshake protocol is not protected at all; anytime it can be attacked by the exhaustion attack of the resources of battery-powered device, may be with the stateless cookie also. So, it is well understood that as an outcome, the nodes could not work properly in the network and make interruption to the whole communication. *Third*, bitmap window can defend the DTLS from replay attack, but still the nodes have to obtain the packets first, then process and occasionally even forward them also. This attack could make the network flooded. So, good resolution can be filtering proxy, for instance, 6LoWPAN Border Router (6LBR). Also, one point in this resolution is that the possibility of running this kind of filtering on a 6LBR cannot protect all situations. Furthermore, handling the replied packets is energy consuming. *Forth* issue is that Handshake phase is strongly defenseless, ever since no end-host has been authentic to the other end-host. *Fifth* issue is that DTLS's security advantages do not match with the CoAP. For example, the loss of a message in-flight necessitates the re-communication of all messages in-flight. But, if all messages in-flight are communicated together in a single UDP packet, its good, but more, resources are obligatory for dealing with large buffers. Additionally, if CoAP client prerequisites Internet access, which essentials the CoAP/HTTP mapping process, then it is well understood that the DTLS handshake process will be a big issue. Mainly, it is not clear if a partial mapping among TLS and DTLS can be accomplished. This topic could also be more complex, since a CoAP client would not be capable to distinguish which device has started the request. Last but not the least, CoAP messages have two transactions (one round-trip); one message starting at the client (request) and the other starting from the server (response). If DTLS is used in these two transaction processes, then we need four round trips, three round trips for DTLS (40–50 Bytes) and additional one round trip for CoAP. It should be before CoAP's actual contents are exchanged.

Distributed IoT applications can use the CoAP at the application layer, with the intention of regaining the resources from sensing devices and in case of the autonomous communications, among WSN and Internet devices. CoAP can be used to empower the application layer RESTful communications with these sensing platforms. So, this can be one of the foundations for the forthcoming great future of future IoT applications. So, it is well understood that the security in case of the CoAP has a major importance. The existing CoAP specification accepts DTLS (Datagram Transport Layer Security) at the transport layer security, for the purpose of transparent secure CoAP communications at the application layer. DTLS offers end-to-end security. But, in actuality, DTLS has a conflict with one functionality designed in CoAP which is the usage of proxies, to help communications among the Internet and WSN communication domains. Another prospect for DTLS for CoAP necessitates the use of public key authentication by use of ECC (Elliptic Curve Cryptography), for the purpose of the authentication and key agreement.

The *handshake* is a big issue for the end-to-end security. The reason is that, after completion of the authentication and key negotiation, the end-to-end security implementation issue can be resolved in the sensing device very efficiently with AES/CCM encryption. We know that the transparent interception and mediation of

DTLS also give us advantages, other than permitting the ECC encryption to make provision for high security with CoAP. The end-to-end security's one of the key components can be the DTLS handshake. It permits for mutual authentication and key agreement, within communicating both the parties. But, it takes some more load, due to its high computational costs, so we should try to offload such costly computations. But, when we are thinking this, we prerequisite to support sensing devices for moving freely in between several WSN domains. We have to take care about the matter that, in the environs of a given IoT application, CoAP resources that exist on sensing devices are securely reachable. The reachability with security should be regardless of the present location of the device. In parallel, there should not be any changes for CoAP and DTLS as maintained on such devices.

7.5 Case Studies and Open Research Issues

At first, let us highlight the ongoing projects and consider them as our case studies. The European Union is working on *Butler* (European Union FP7 project) [46–48]. This project facilitates the expansion of secure and smart life assistant applications, along with the security and privacy necessities. Also, this work has developed a mobile framework. The smart applications which are targeted are like smart home/smart office, smart mobility/smart transport, smart health, smart shopping, and smart cities. Another European Union project is *EBBITS* (EU FP7 project) [47]. This project works for an IDS, by use of latest IPv6 over 6LoWPAN devices. Ever since, 6LoWPAN protocol is defenseless to wireless and Internet protocol attacks. This projected IDS framework comprises a monitoring system and a detection engine. The *Hydra project* [49] has projected a middleware for Network Embedded Systems. This middleware is founded on a Service-Oriented Architecture (SOA). Hydra considers the distributed security concerns and social trust within the middleware constituent. Hydra is designed for P2P communication and diagnostics, architecture formed on Semantic Model and the Device and Service Discovery. Another project which is to increase the user trust is *uTRUSTit* [50]. *uTRUSTit* stands for Usable Trust in the IoT (EU FP7 project). It is actually a trust feedback toolkit to potentially increase the user trust. It empowers the system manufacturers and system integrators to express the security ideas. It agrees to create effective decisions on the trustworthiness. *iCore* is another EU project. *iCore* [51] has a management framework with very significant security protocols/functionalities. These protocols/functionalities are having relation with the ownership and privacy of data and the access to objects. This management framework has three levels of functionality: virtual objects (VOs), composite virtual objects (CVOs), and functional blocks. The *iCore* solution can be part of various smart environs, like supply chain management, smart office, smart transportation, and ambient-assisted living.

Now, another very well-known DARPA project is *HACMS* [52]. It stands for High Assurance Cyber Military Systems. This project actually has tried to have

patch of the security vulnerabilities of IoT. This project has taken account of drones, medical equipment, and military vehicles. HACMS provides the seeds for future security protocols and achieves sufficient standardization and security.

NSF, *National Science Foundation*, has a *multi-institutional project* [53]. This project is actually working for the security in the cyber-physical systems. This multi-institutional project is working on several solutions, like trying to discover the efficient resolutions, finding novel network architectures and networking conceptions, trying to invent new communication protocols. Also, they are bearing in mind about the trade-offs between mobility and scalability, technical challenges, trusted data, the integrity along with authentication, trust models, and use of network resources on mobile environments. The EU, China, and Korea are working together in a project called *FIRE* [54, 55]. It stands for *Future Internet Research and Experimentation*. The *FIRE* works for discovering resolutions, for the setting out of IoT technologies in numerous application areas, like medical and health service, urban management, social security, people livelihood, and public safety. They are also trying to give proper focus on intellectual property right, privacy, and information security. Another EU and Japanese collaborative project is *EUJapan ICT Cooperation project* [47]. They have already made the common global standards, to make sure about seamless communications and shared ways to accumulate and have right to use the information. They are also trying to confirm the highest security and energy efficiency standards.

In 1999, the Auto-ID Laboratory of Massachusetts Institute of Technology has introduced us *Thought of "the Internet of things"*. Then, in 2005, we had the "*ITU Internet Reports: The Internet of Things*." We need to develop the security structural design of the IoT, for the reason of offering information security defense for tag privacy, sensor data security, and data transmission, etc. We need very deep systematic research on the transmission and information security of the core network, depending on the IoT or networking industry security of the IoT. We have seen that recent works are simply adding safety methods in each layer. But, this is not at all sufficient. We have seen that, depending on the *privacy homomorphism*, the computational insufficiency of traditional algorithms is enhanced to make sure users' personal privacy security. It is one of the milestone ideas. But, the homomorphism technology presently is not matured enough as required. Now, the homomorphism algorithm is capable of offering the complete integer operations. Still nowadays, it is comprehensive to the real region that the security comes out to a big issue. Also, another disadvantage is that very few homomorphism properties are held by the privacy homomorphism. So, we need more developed homomorphism, which can be extensively used in IoT. We have worked on multilayer *Hybrid RSA*-based solution [45, 56–59] for personal messaging for more efficiency and strong security and privacy as shown in Fig. 7.7. Our Hybrid RSA scheme now works for human messaging, and in later stage this Hybrid RSA cipher [45, 56–59] can be used for Internet of Everything (IoE) for end-to-end encryption with high efficiency and high security with authentication and privacy protection.

In generic, the security actions to be taken for IoT denote to the basic facility of security services comprising availability, authentication, authorization, non-

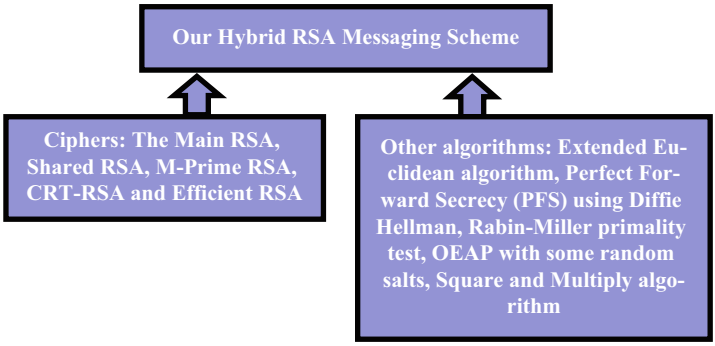

Fig. 7.7 Our Hybrid Rivest–Shamir–Adleman (RSA) scheme

repudiation, confidentiality, and integrity. The security structural design of IoT is still growing. So, the best way to represent the security need can be by using a reference model as we discussed earlier. So, it is well understood that any single structural design will be problematic for referring to the system. All the researchers, governments, and industries are dedicated for evolving and regulating identity and security mechanisms, for IoT building blocks. We already know that researchers are forming better cryptographic algorithms and modes for IoT devices. The *ISO/IEC 29192 standards* aim for lightweight cryptography for constrained devices. This standard includes block and stream ciphers and asymmetric mechanisms. Sony's *CLEFIA* is an example of block cipher with 128-bit key supports (www.sony.net/Products/cryptography/clefia/about/index.html). The *eSTREAM project* (www.ecrypt.eu.org/stream) has considered the robustness of stream ciphers, for instance, *Salsa20/12* and *Trivium*. These are very much beneficial for embedded systems. Also, we know that some researches on *lightweight dedicated hash functions* are going on. Everybody in this area is trying to make a new *cryptographic hash algorithm* that is able to transform a variable-length message into a short message digest. The digest can be a portion of either generating digital signatures or message authentication codes or can be many other security applications in the information setup (http://csrc.nist.gov/groups/ST/hash/sha-3/index.html). We have already works which are on forming lightweight hash functions, depend on lightweight block ciphers. We know that *AES-CCM* and *AES-GCM* project data integrity and confidentiality. Another way for optimization can be algorithm management in a *cross-layer architecture*. Here, the reason for optimization is numerous security mechanisms that share *one algorithm*. The Internet Engineering Task Force has an intention to execute Internet standards in the IoT. We have seen that many researchers have tweaked the IPSec protocol, for offering the network layer security between Internet hosts and constrained devices. But, still some issues are hard to resolve. We all know that the IPSec prerequisites a shared password, for doing the encryption and decryption for all incoming and outgoing messages. But, big issue is that if these passwords are static,

then it can be compromised after some 1000 messages. For resolving this issue, the IKE (Internet Key Exchange) and IKEv2 protocols were formed. These protocols promise a protected communication among two devices and are capable to generate new shared passwords, by use of circling derivative tactics. We can use DTLS for protecting UDP packets (even over IPSec). By use of an initial handshake, it sets the passwords. Then, the content of the UDP packet is encrypted (usually with TLS PSK over AES) and a header of 13 bytes is added. This process is done together with the initialization Vectors (IV) (over 8 bytes for AES128), integrity values (8 bytes), and the padding prerequisite by the cipher suite. In general, future researches in the security issues of the IoT would mostly quintessence on the following characteristics: the open security system, individual privacy protection mode, terminal security function, related laws for the security of the IoT, etc. It is unquestionable that the security of the IoT is more than a technical problem, which also prerequisites a series of policies, laws, and regulations, perfect security management system for mutual collocation.

Acknowledgments This work is supported by National Natural Science Foundation of China (No. 61631013) and Key Laboratory of Universal Wireless Communications (Beijing University of Posts and Telecommunications), Ministry of Education, P.R. China (No. KFKT-2014101).

References

1. Jara A, Kafle V, Skarmeta A (2013) Secure and scalable mobility management scheme for the internet of things integration in the future internet architecture. Int J Ad Hoc Ubiquitous Comput 13(3-4):228–242
2. Li S, Gong P, Yang Q, Li M, Kong J, Li P (2013) A secure handshake scheme for mobile-hierarchy city intelligent transportation system. In: International conference on ubiquitous and future networks. ICUFN, Da Nang, pp 190–191
3. Kang KC, Pang ZB, Wang CC (2013) Security and privacy mechanism for health internet of things. J China Univ Posts Telecommun 20(Suppl 2):64–68
4. Goncalves F, Macedo J, Nicolau M, Santos A (2013) Security architecture for mobile e-health applications in medication control. In: 2013 21st international conference on software, telecommunications and computer networks. SoftCOM, Primosten, pp 1–8
5. An J, Gui X, Zhang W, Jiang J, Yang J (2013) Research on social relations cognitive model of mobile nodes in internet of things. J. Netw Comput Appl 36(2):799–810
6. Kasinathan P, Costamagna G, Khaleel H, Pastrone C, Spirito M (2013) Demo: an ids framework for internet of things empowered by 6lowpan, Berlin, Germany, pp 1337–1339
7. BETaaS Consortium (2014) BETaaS building the environment for the things as a service D2. 2. 2–Specification of the extended capabilities of the platform, pp 1–61
8. IoT-A Consortium (2014) IoT-A unified requirements. http://www.iot-a.eu/public/requirements/. 31 Jan 2014
9. Gao L, Bai X (2014) A unified perspective on the factors influencing consumer acceptance of internet of things technology. Asia Pac J Mark Logist 26(2):211–231
10. Gazis V (2014) Carlos Garcia Cordero, Emmanouil Vasilomanolakis, Panayotis Kikiras, and Alex Wiesmaier. Security perspectives for collaborative data acquisition in the internet of things. In: International conference on safety and security in internet of things. Springer, New York
11. IoT-A Consortium (2014) IoT-A – Internet of things architecture. http://www.iot-a.eu/. 27 Jan 2014

12. Logvinov O, Kraemer B, Adams C, Heiles J, Stuebing G (2014) Mary Lynne Nielsen, and Brenda Mancuso. Standard for an architectural framework for the internet of things (IoT) IEEE P2413 Webinar Panelists, pp 1–12
13. Zanella A, Bui N, Castellani AP, Vangelista L, Zorzi M (2014) Internet of things for smart cities. IEEE Internet Things J 1:22–32
14. Grieco LA, Alaya MB, Monteil T, Drira KK (2014) Architecting information centric ETSI-M2M systems. In: IEEE PerCom
15. Anderson J, Rainie L (2014) The internet of things will thrive by 2025, Pew research internet project. http://www.pewinternet.org/2014/05/14/internet-of-things/
16. Yan Z, Zhang P, Vasilakos AV (2014) A survey on trust management for internet of things. J Netw Comput Appl 42:120–134
17. Piro G, Boggia G, Grieco LA (2014) A standard compliant security framework for IEEE 802.15.4 networks. In: Proceedings of IEEE world forum on internet of things (WF-IoT), Seoul, South Korea, pp 27–30
18. Lee J-Y, Lin W-C, Huang Y-H (2014) A lightweight authentication protocol for internet of things. In: 2014 international symposium on next-generation electronics, ISNE 2014, Kwei-Shan, pp 1–2
19. Turkanovi M, Brumen B, Hlbl M (2014) A novel user authentication and key agreement scheme for heterogeneous ad hoc wireless sensor networks, based on the internet of things notion. Ad Hoc Netw 20:96–112
20. Ye N, Zhu Y, Wang R-CB, Malekian R, Lin Q-M (2014) An efficient authentication and access control scheme for perception layer of internet of things. Appl Math Inf Sci 8(4):1617–1624
21. Cherkaoui A, Bossuet L, Seitz L, Selander G, Borgaonkar R (2014) New paradigms for access control in constrained environments. In: 2014 9th international symposium on reconfigurable and communication-centric systems-on-chip (ReCoSoC), Montpellier, pp 1–4
22. Sicari S, Rizzardi A, Cappiello C, Coen-Porisini A (2014) A NFP model for internet of things applications. In: Proceedings of IEEE WiMob, Larnaca, Cyprus, pp 164–171
23. Wang X, Zhang J, Schooler E, Ion M (2014) Performance evaluation of attribute-based encryption: Toward data privacy in the IoT. In: 2014 IEEE international conference on communications, ICC 2014, Sydney, NSW, pp 725–730
24. Su J, Cao D, Zhao B, Wang X, You I (2014) ePASS: an expressive attribute-based signature scheme with privacy and an unforgeability guarantee for the internet of things. Futur Gener Comput Syst 33:11–18
25. Peng LB, Ru-chuan WB, Xiao-yu S, Long C (2014) Privacy protection based on key-changed mutual authentication protocol in internet of things. Commun Comput Inf Sci 418:345–355
26. Ukil A, Bandyopadhyay S, Pal A (2014) IoT-privacy: to be private or not to be private. In: Proceedings – IEEE INFOCOM, Toronto, ON, pp 123–124
27. Sicari S, Cappiello C, Pellegrini FD, Miorandi D, Coen-Porisini A (2014) A security-and quality-aware system architecture for internet of things. Inf Syst Front 18:1–13
28. Tormo GD, Marmol FG, Perez GM (2014) Dynamic and flexible selection of a reputation mechanism for heterogeneous environments. Futur Gener Comput Syst 49:113–124
29. Gu L, Wang J, Sun BB (2014) Trust management mechanism for internet of things. China Commun 11(2):148–156
30. Liu Y-B, Gong X-H, Feng Y-F (2014) Trust system based on node behavior detection in internet of things. Tongxin Xuebao/J Commun 35(5):8–15
31. Singh J, Bacon J, Eyers D (2014) Policy enforcement within emerging distributed, event-based systems. In: DEBS 2014 – Proceedings of the 8th ACM international conference on distributed event-based systems, pp 246–255
32. Neisse R, Steri G, Baldini G (2014) Enforcement of security policy rules for the internet of things. In: Proceedings of IEEE WiMob, Larnaca, Cyprus, pp 120–127
33. Gòmez-Goiri A, Orduna P, Diego J, de Ipina DL (2014) Otsopack: lightweight semantic framework for interoperable ambient intelligence applications. Comput Hum Behav 30:460–467
34. Colistra G, Pilloni V, Atzori L (2014) The problem of task allocation in the internet of things and the consensus-based approach. Comput Netw 73:98–111

35. Wang Y, Qiao M, Tang H, Pei H (2014) Middleware development method for internet of things. Liaoning Gongcheng Jishu Daxue Xuebao (Ziran Kexue Ban)/J Liaoning Tech Univ (Nat Sci Ed) 33(5):675–678
36. Ferreira H, De Sousa R Jr, De Deus F, Canedo E (2014) Proposal of a secure, deployable and transparent middleware for internet of things. In: Iberian conference on information systems and technologies. CISTI, Barcelona, pp 1–4
37. Niu B, Zhu X, Chi H, Li H (2014) Privacy and authentication protocol for mobile RFID systems. Wireless Pers Commun 77(3):1713–1731
38. Jeong Y-S, Lee J, Lee J-B, Jung J-J, Park J (2014) An efficient and secure m-IPS scheme of mobile devices for human-centric computing. J Appl Math 2014:1–8
39. Geng J, Xiong X (2014) Research on mobile information access based on internet of things. Appl Mech Mater 539:460–463
40. Kubler S, Frmling K, Buda A (2014) A standardized approach to deal with firewall and mobility policies in the IoT. Pervasive Mobile Comput 20:100–114
41. Daubert J, Wiesmaier A, Kikiras P (2015) A view on privacy & trust in IoT. In: IOT/CPS-Security Workshop, IEEE international conference on communications, ICC 2015, London, GB, June 08–12, 2015, page to appear. IEEE
42. Sadeghi AR, Wachsmann C, Waidner M (2015) Security and privacy challenges in industrial internet of things. In: Annual design automation conference. ACM, New York, p 54
43. Sicari S, Rizzardi A, Grieco LA, Coen-Porisini A (2015) Security, privacy and trust in internet of things: the road ahead. Comput Netw 76:146–164
44. Zhang Z-k, Cheng M, Cho Y, Shieh S (2015) Emerging security threats and countermeasures in IoT. In: ACM symposium on information, computer and communications security. ACM, New York, pp 1–6
45. Bhattacharjya A, Zhong X, Wang J (2016) Strong, efficient and reliable personal messaging peer to peer architecture based on Hybrid RSA. In: Proceedings of the international conference on internet of things and cloud computing (ICC 2016) ISBN 978-1-4503-4063-2/16/03. The Møller Centre-Churchill College, Cambridge. https://doi.org/10.1145/2896387.2896431
46. BUTLER Project. http://www.iot-butler.eu
47. EU-Japan Project. http://www.eurojapan-ict.org/
48. European FP7 IoT@Work project. http://iot-at-work.eu
49. HYDRA Project. http://www.hydramiddleware.eu/
50. Usable Trust in the Internet of Things. http://www.utrustit.eu/
51. iCORE Project. http://www.iot-icore.eu
52. HACMS Project. http://www.defenseone.com/technology
53. National Science Foundation Project. http://www.nsf.gov
54. FIRE EU-China Project. http://www.euchina-fire.eu/
55. FIRE EU-Korea Project. http://eukorea-fire.eu/
56. Bhattacharjya A, Zhong X, Wang J (2018) An end to end users two way authenticated double encrypted messaging scheme based on hybrid RSA for the future internet architectures. Int J Inf Comput Secur 10(1):63–79
57. Bhattacharjya A, Zhong X, Wang J, Xing L (2018) On mapping of address and port using translation (MAP-T). Int J Inf Comput Secur. http://www.inderscience.com/info/ingeneral/forthcoming.php?jcode=ijics. https://doi.org/10.1504/IJICS.2018.10008372
58. Bhattacharjya A, Zhong X, Wang J (2018) HYBRID RSA based highly efficient, reliable and strong personal full mesh networked messaging scheme. Int J Inf Comput Secur. http://www.inderscience.com/info/ingeneral/forthcoming.php?jcode=ijics. https://doi.org/10.1504/IJICS.2018.10010256
59. Bhattacharjya A, Zhong X, Wang J, Xing L (2018) Secure IoT structural design for smart cites. In: Smart cities cybersecurity and privacy. Elsevier, New York. ISBN: 9780128150320. https://www.elsevier.com/books/smart-cities-cybersecurity-and-privacy/rawat/978-0-12-815032-0#

Aniruddha Bhattacharjya is with Department of Electronic Engineering, Tsinghua University, Beijing, China, as a PhD scholar (under Chinese Government Scholarship (CGS)). His research interests are cryptography, Network security, RFID-based architectures and middleware, security in fixed and wireless Networks, applications of cryptography, and IoT security. He has received the ICDCN 2010, PhD Forum Fellowship. He achieved the best paper award in ACM ICC 2016, in Cambridge University, UK. Since 2012, he has been working as an IEEE mentor and ACM faculty sponsor. He is a member of 34 IEEE societies and various IEEE technical committees. He has published 33 International Journal papers, International Conference papers and International Book chapters as well as one Chinese innovation patent is filed presently.

Xiaofeng Zhong received his PhD in Information and Communication Systems from Tsinghua University in 2005. He is an Associate Professor in the Department of Electronic Engineering at Tsinghua University. He performs research in the field of mobile networks, including users' behaviors and traffic model analyses, MAC and network protocol design, and resource management optimization. He has published more than 30 papers and holds seven patents.

Jing Wang received his BS and MS degree in Electronic Engineering from Tsinghua University, Beijing, China in 1983 and 1986, respectively. He has worked as a Faculty member in Tsinghua University since 1986. He is currently a Professor at the School of Information Science and Technology. He also serves as the Vice Director of the Tsinghua National Laboratory for Information Science and Technology. His research interests are in the area of wireless communications, including transmission and networking technologies of 4G/5G. He has published more than 150 conference and journal papers.

Xing Li is a Professor in the Department of Electronic Engineering, Tsinghua University, Beijing, China and deputy Director of CERNET Center. He obtained his PhD degree from the Department of Electrical and Computer Engineering at the Drexel University, Philadelphia, USA in 1989. He has published numerous papers and is the author of several RFCs. He is also WWW10, PC Chair in Searching Area and Co-Chair of the Coordination Committee of the Intercontinental Research Network (CCIRN).