JCSE Journal of Computing Science and Engineering

Regular Paper

Journal of Computing Science and Engineering,
Vol. 12, No. 1, March 2018, pp. 12-23

Why Dynamic Security for the Internet of Things?

Seyyed Yasser Hashemi and Fereidoon Shams Aliee*
Faculty of Computer Science and Engineering, Shahid Beheshti University, Tehran, Iran
y_hashemi@sbu.ac.ir, F_shams@sbu.ac.ir

Abstract
The Internet of Things (IoT) ecosystem potentially includes heterogeneous devices with different processing mechanisms as well as very complicated network and communication models. Thus, analysis of data associated with adverse conditions is much more complicated. Moreover, mobile things in the IoT lead to dynamic alteration of environments and developments of a dynamic and ultra-large-scale (ULS) environment. Also, IoT and the services provided by that are mostly based on devices with limited resources or things that may not be capable of hosting conventional controls. Finally, the dynamic and heterogeneous and ULS environment of the IoT will lead to the emergence of new security requirements. The conventional preventive and diagnostic security controls cannot sufficiently protect it against increasing complication of threats. The counteractions provided by these methods are mostly dependent on insufficient static data that cannot sufficiently protect systems against sophisticated and dynamically evolved attacks. Accordingly, this paper investigates the current security approaches employed in the IoT architectures. Moreover, we define the dynamic security based on dynamic event analysis, dynamic engineering of new security requirements, context awareness and adaptability, clarify the need for employment of new security mechanism, and delineate further works that need to be conducted to achieve a secure IoT.

Category: Privacy and Security

Keywords: Dynamic Security; Internet of Things; IoT architectures

I. INTRODUCTION

We can define security as a situation free from any risk or threat, or a situation free from any misgiving, and anxiety [1]. The concept of security was not considered a significant requirement in the early computer systems, and this was mainly because the systems were mostly centralized and concerns were mostly associated with physical security of computers or protection of them against theft or computer hardware sabotage. With the development of distributed systems and emergence of ultra-large-scale (ULS) systems, the need for security became more prominent. There are many views about security goals. According to Bishop, three important aspects of computer security include confidentiality, integrity, and availability [1]. Menezes defined 17 major objectives of information security that include confidentiality, integrity, and identification. According to some other sources, sign and seal are the main objectives of information security. Although we mainly use these concepts as mechanisms and tools for achieving a specific goal, Hafner [1] argues that security goals derive from four concepts: confidentiality, integrity, authentication, and non-repudiation. In another definition, Eckert defined 6 primary goals for security:

Open Access http://dx.doi.org/10.5626/JCSE.2018.12.1.12 http://jcse.kiise.org

Received 23 July 2017; **Revised** 19 December 2017; **Accepted** 28 January 2018
*Corresponding Author

 pISSN: 1976-4677 eISSN: 2093-8020

authenticity, confidentiality, integrity, availability, auditability, and anonymity [1]. However, the goals of security that are commonly stated in different texts include confidentiality, integrity, and availability [1-5].

On the other hand, the Internet of Things (IoT) technology emerged after entering the era of information and new communications. The IoT is generally defined as a set of standards, protocols, devices, and technologies required for communication and transfer of data between smart devices (and human beings) at a global level. Therefore, we can also define IoT as an extensive network that connects all objects across the world, by specific rules. In fact, IoT is a concept according to which the smart objects are equipped with sensors, drivers, micro-processors, communication interfaces, and energy resources, and are capable of performing different processes and communicating with one another. This network is mainly designed to share the data with the objects with which it is associated, and to enable objects to contact anything and anyone that ideally uses any path or network or service at any time or place.

Due to its unique features such as uncontrolled, dynamic and movable environment, physical accessibility of objects, heterogeneity of objects, and constrained resources, IoT has managed to develop a dynamic and highly heterogeneous ecosystem implementation associated with numerous challenges. The World Wide Web that was globalized years ago still has many security weaknesses that have threatened the life and even the properties of many people so far. In such circumstances, the establishment of security in a worldwide web of things with specific features and limitations, that contact each other and human beings, is much more complicated. New environmental conditions as well as dynamic nature and different characteristics of devices are among the factors that have placed the IoT security at the center of attention and instigated us to provide suitable architecture and security mechanisms for that. In the present study, we define the new security requirements of IoT under the light of dynamic security and investigate the need for employment of dynamic security in IoT.

We have designed the remaining parts of this paper as follows: Section II introduces the most important features of IoT that we should take into account in the security mechanisms considered for this internet. In Section III, we investigate the security requirements of IoT and define the dynamic security for this internet. In Section IV, we discuss the security and architectures of this internet as the most important measures taken to guarantee the IoT security. In Section V, we compare the mechanisms provided by the IoT architectures and investigate the use of these mechanisms from the perspective of dynamic security. We also discuss the need for the use of dynamic security as well as the shortcomings of the approaches presented in this regard in this section. We discuss conclusion and suggestions for further research in Section VI.

II. IOT CHARACTERISTICS

The most important characteristics of IoT that distinguish it from other environments include:

- **Uncontrolled environment**: The environment of this internet is open and uncontrolled. Objects move across the vast environment of this internet without being monitored by a central system and are also physically accessible. In such an environment, stable network connection and continuous presence are unexpected. Moreover, in this environment, the previous trust relationships will not be effective, and we will require new approaches for the new trust level between objects, services and users. The uncontrolled environment of this internet calls for a revision of the previous security approaches, or introduction of new methods. The main pillar of the uncontrolled environment includes mobility, physical accessibility, and trust.
- **Heterogeneity**: Different objects with different features that are produced by different producers are merged in the IoT ecosystem, and are supposed to provide new services in cooperation with one another. In this highly heterogeneous ecosystem, compatibility and interoperability are of utmost importance.
- **Scalability**: A large number of objects have come together in IoT and have created a ULS ecosystem. A large number of objects and a significant part of the ULS environment in this internet require highly scalable protocols so that they can have the required efficiency in this environment
- **Constrained resources**: In IoT, objects don't have the same potentials. Many objects in this internet usually have limitations regarding energy and processing power and thus require appropriate security mechanisms.

Considering the above-mentioned characteristics, IoT has created a dynamic, heterogeneous and ULS ecosystem consisting of different objects and constrained resources that have led to the emergence of new security requirements and challenges. The security requirements of IoT will be discussed in the next section.

III. IOT SECURITY REQUIREMENT

As mentioned before, IoT has created a dynamic, heterogeneous and ULS ecosystem consisting of different objects and constrained resources that have led to the emergence of new security requirements and challenges. The security requirements of the IoT are discussed from two points of view: the general security requirements and the dynamic security requirements. It is believed that the dynamic security requirements are applied, as a vertical layer, on the general security requirements and affect them. The approaches provided for the general security

requirements are supposed to have dynamic security features as nature of the IoT requires, so that they can successfully improve the security conditions in this ULS, heterogeneous, and dynamic ecosystem.

A. IoT Common Security Requirements

In its primary form, the IoT is a self-configurable dynamic ultrastructure of the World Wide Web based on standard, compatible, and functional communication protocols, in which the virtual and physical objects, as well as the physical features and virtual characters, are provided with a specific identity and integrated by means of smart interfaces in the information network [6]. Therefore, in this dynamic ultrastructure of the World Wide Web, the security challenges and requirements are much more severe, and new requirements may also emerge over time. A general classification of the IoT common requirements includes:

- **Network security**: The network security requirements can be divided into confidentiality, authenticity, integrity, and availability. These requirements need to be applied to the architecture of the IoT.
- **Identity management**: Considering the number of devices, the sophisticated relationship between devices, services, owners and users, identity management in the IoT is associated with specific challenges. Thus, it is necessary to pay special attention to authentication, authorization, revocation, accountability, and non-repudiation in this regard.
- **Privacy**: include data privacy, anonymity, pseudonymity, and unlinkability.
- **Trust**: can be divided into device trust, entity trust and data trust.
  - **Device trust**: In IoT, the previous relationships associated with device trust don't hold true all the time due to reasons such as high dynamism and interaction between domains. Thus, methods such as trust computing and trusted computing are necessary for building trust between devices. Also, each entity may examine trust in different ways, depending on the device type. Therefore, the IoT architecture needs to deal with non-singular trust views.
  - **Entity trust**: Refers to the required behavior of participants including individuals or services.
  - **Data trust**: In IoT, the data trust should be built into two folds. (1) An organization of data obtained from generally non-trustable devices. Thus, methods such as data aggregation and machine learning need to be used to extract trustable data from non-trustable resources. (2) The IoT services derive new data from the integration of different data types. For a newly created data, a new trust evaluation (for example through trusted calculation) is required.

The relationship between security requirements and the IoT characteristics are provided in Table 1. In Table 1, H, M, and L show the high, medium, and low influence of characteristics on security requirements, respectively.

B. Dynamic Security for IoT

The heterogeneous, dynamic, and ULS of the IoT have made it impossible for the static security mechanisms to fulfill the security requirements of the IoT. Therefore, considering the future changes in the security requirements and the dynamic security requirements of the IoT, security mechanisms that are applicable in the dynamic, ULS, and heterogeneous environment of the IoT are required. Thus, from the perspective of this study, the main pillars of dynamic security are:

- **Dynamic requirement analysis (DRA)**: The dynamic and heterogeneous environment of the IoT, as well as its ULS, will lead to the emergence of new security requirements. Therefore, the engineering of security requirements and dynamic review of them play a pivotal role in the security of this internet.
- **Context awareness (CA)**: The information associated with context is regarded as a key element in the IoT. The IoT has a dynamic, ULS, and heterogeneous environment; therefore, data are extracted from different objects and are integrated into macro data.

Table 1. Relationship between security requirements and the IoT attributes

	Network security	Identity management	Privacy	Trust	Resilience
Uncontrolled environment					
Mobility	L	L	L	H	L
Physical accessibility	M	L	M	H	M
Trust	M	M	L	H	L
Heterogeneity	L	M	L	M	L
Scalability	L	L	M	L	H
Constrained resources	M	L	M	L	L

H: high, L: low, M: medium.

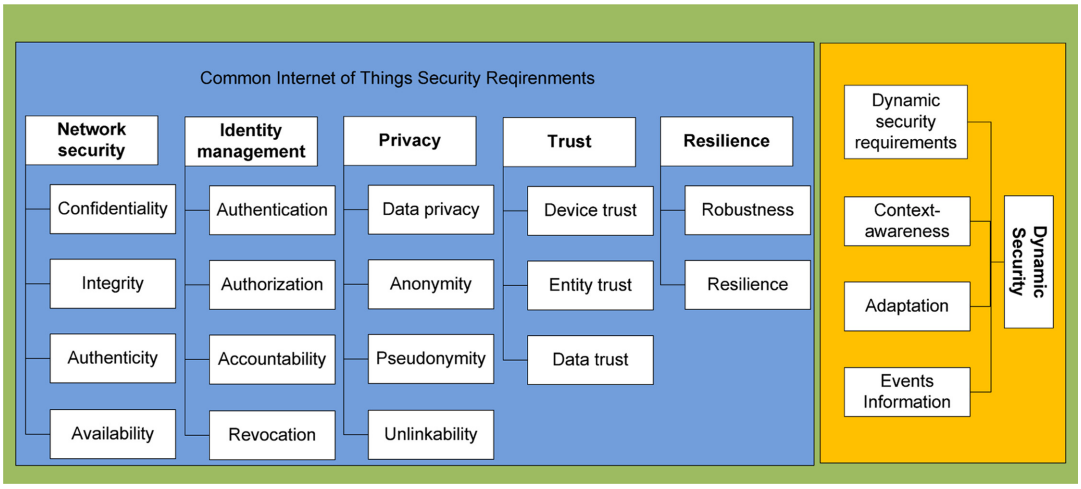

Fig. 1. Dynamic security requirements are applied, as a vertical layer, on the general security requirements and affect them.

Table 2. A general classification of the common security requirements of the IoT as well as its dynamic security parameters that affect the approaches used to meet the security requirements of the IoT

	Network security	Identity management	Privacy	Trust	Resilience
Effective dynamic security parameters	AD-CA	AD-CA-EI-DRA	DRA-AD	AD-CA-EI-DRA	DRA-AD
Uncontrolled environment					
Mobility	H	H	H	H	H
Physical accessibility	L	L	L	L	L
Trust	H	H	M	H	H
Heterogeneity	H	H	H	H	M
Scalability	H	H	H	H	H
Constrained resources	H	H	H	H	H

AD: adaption, CA: context awareness, EI: event information, DRA: dynamic requirement analysis, H: high, L: low, M: medium.

The collected data may be worthless, thus they need to be checked, refined and converted into knowledge. This process is conducted under the veil of context awareness and useful context-related data including the location and potential of things are used in different security sections.

- **Event information (EI)**: The use of events information in the dynamic environment of the IoT can include useful data about security and the level of trust in devices and entities. This process can be useful in dynamic selection of security approaches. Therefore, the event information constitutes one of the main pillars of dynamic security in the IoT.
- **Adaptation (AD)**: In most cases, active operation of the IoT is dynamically conducted in the unprotected real-time environments where the response to risks and threats should be dynamically taken into account. Adaptability is a system feature for self-regulation of behaviors in accordance with the current situation and autonomous reconfiguration of settings. Accordingly, the security approaches should be adaptable and adaptability is considered one of the main pillars of dynamic security in the IoT.

Finally, according to the above information, we define dynamic security for the IoT as follows:

"*Mechanisms for development of adaptable security in the IoT according to dynamic analysis of security requirements, context awareness and review of events in the environment of the IoT*".

Fig. 1 shows that the dynamic security requirements are applied, as a vertical layer, on the general security requirements and affect them. Table 2 shows a general classification of the common security requirements of this internet as well as its dynamic security elements that affect the approaches used to meet the security requirements of the IoT.

IV. SECURITY IN THE IOT ARCHITECTURES

In the previous section, we discussed the IoT characteristics that make new security mechanisms necessary as well as the security requirements of the IoT and introduced the concept of dynamic security in the IoT. In the present section, we introduce the security approaches of the most important IoT architectures to investigate the relevant works and mechanisms introduced for the security of the IoT as well as the degree to which dynamic security is used in them.

The potential of these architectures for covering the dynamic and general security requirements of the IoT is compared and the need for the introduction of new approaches is discussed in this section.

A. Architectural Reference Model

This architecture is the output of the IoT-A project conducted in the international research center of the European Union [7]. Adaptability serves as the origin of the need for a comprehensive architecture for the IoT. In the contemporary world, different smart networks are designed and implemented by their specific architecture. These networks are not adaptable to other networks. These smart networks are in fact a large but separate subset of the IoT that bring the IoT, rather than their goals, to fruition. Achievement of the main goal requires a comprehensive architecture for implementation of the IoT, so that the above-mentioned diverse networks can be executed based on the same platform (just like a tree with a large number of leaves, but one single bark), become adaptable to one another, support one another, and finally have interaction with one another at the same time. That's why IoT-A was launched to provide the IoT with a reference architecture (ARM).

The IoT covers many orthogonal items that don't support interactivity and adaptability. Many attempts have been made for the development of these features, but the presented solutions are not scalable and cannot, therefore, for the IoT in the future. Moreover, subjects such as privacy and security are not taken into account in these solutions.

In IoT-A, security is implemented based on four aspects, security, trust, privacy, and availability of five components [7]:

- Key management and exchange (KEM)
- Authorization (authz)
- Authentication (AuthN)
- Identity management (IM)
- Trust and reputation (TRA)

B. Embedded Device Gateways Cloud (MGC)

The MGC architecture is, in fact, an architecture proposed for IoT based on cloud computing that is going to be derived from the SITP project [8]. The MGC architecture was proposed to meet the processing requirements of IoT. This architecture includes four building blocks 'Compact devices,' 'Gateways,' 'Cloud,' and 'Application.'

This project has two main goals:

- **Data security**: Investigating and defining new computational encryption models for secure data analysis and application of them on large real-time data sequences in compact systems.
- **System security**: Realization and implementation of a safe and open software/hardware framework that facilitates the employment of these new computational models in the development of the IoT applications.

The system framework develops a code structure for the stages mentioned above. The designer can modify this code to meet his/her requirements. In the meantime, framework mentioned above continuously checks the modifications to make sure that they don't violate the architecture proposed for data processing. The data type, data leakage, and encryptions are also checked out by this framework. Also, the proposed framework is also supposed to automatically solve problems associated with low power protocols, communication layer complexities, and protocol mismatch through the new network algorithms. The security and privacy implementation approach is inclined towards employment of software-defined hardware, and observance of simplicity at all levels.

C. BETaaS Architecture

This architecture is proposed based on the TaaS reference model and through augmentation of the IoT-A architecture [9]. BETaaS is an IoT-based architecture that proposes a machine to machine (M2M) relationship that the applications could be run on the local cloud of gates. Each BETaaS architecture developed its gate clouds that integrate the M2M heterogeneous systems into an integrated approach. As mentioned before, BETaaS depends on the reference model of things known as TaaS.

This architecture includes four layers; the first layer is known as a physical layer that covers M2M systems connected to the platform. The second layer is known as the adaptation layer that provides connections to the physical layer. This abstract layer belongs to the personal M2M systems. The third layer is TaaS and is located on the abstract layer and provides extensive access to the network to the M2M layer devices. The last layer is known as the service layer that manages the performance of the BETaas application services.

From the architectural point of view, BETaaS claims that the security requirements of all internet layers, except for the physical layer, can be met through some unique mechanisms.

In network security, key management includes entities, authentication, user session management, and provision of encrypted communications. Since the BETaaS architectures

include several gateways, they usually use a PKI with a certificate authority to manage keys and to ensure privacy, authenticity and accuracy through secure communication channels. The BETaaS architecture can cover cases that include several complicated organizations, e.g., the external entities that are controlled by internal CA. moreover, BETaaS applied effective computational encryption models such as ECC on devices with constrained resources.

The BETaaS architecture provides the specific architecture component of authentication for management of identity. Authentication is divided into two types: authentication at the gateway level, for example when a gateway joins a BETaaS, and authentication at the service or application level, for example when a user is using an application. In the first case, the authentication module uses key management, while in the second case, the OAuth can be developed for authentication and authorization purposes. Authorization is perfectly covered by this factor, but the accountability requirements in BETaaS are still unclear. Although privacy is considered a key factor of security mechanisms in BETaaS, no information is available about the way this requirement is met in BETaaS. Identity management is responsible for management of the sensors and gateway's identity. However, nothing has been said about the data anonymity and aliases.

In BETaaS, trust is provided by the Trust and Reputation component. Aspects of individual trust include security mechanisms (such as information related to encryption algorithms, licences and so on), launch of QoS, efficiency, dependability, battery load, and stability in data supply.

These aspects of trust come together so that the final value of trust can be calculated. Flexibility is employed through four main principles: failure prevention, failure elimination, failure tolerance, and failure prediction. Failure analysis is responsible for the detection of potential fault reasons and presentation of solutions for their management.

D. WSO2 Architecture

The WSO2 Company in the United States presented a reference architecture that covers both server and cloud devices as well as the required architecture for the mutual effect of devices as well as their management [10]. The main goal of this project was to provide designers and architects with an effective starting point that covers many IoT requirements in the development systems and projects. However, this architecture is not limited to a specific set of technologies and doesn't emphasize on the details of a client-server, hardware or cloud architecture, but provide an architecture that is independent of specific providers.

In the WSO2 architecture, security is provided in the form of a vertical layer and the access and authentication management layer performs different security operations at different architecture layers and levels. In the security provided by this architecture, token-based model rather than username/password is used for authentication.

E. OpenIoT

The European Union FP7 OpenIoT project (2011–2014) introduced an IoT architecture. OpenIoT architecture is based on reference architecture model of IoT-A (ARM) and supports the main concepts of the ARM. This architecture provides a cloud-based middleware infrastructure for demand-based access to IoT and IoT services [11]. The OpenIoT architecture provides an open-source implementation based on structural principles of IoT applications with cloud-based attributes and acts as a demand-based service or pay-as-you-go service. From an architectural point of view, OpenIoT deals with a combination of cloud and IoT.

This architecture includes two security modules [11]: security and privacy module, and trust module. In the security module, a sub-module deals with security messages, authentication, and authorization. Currently, the privacy attributes are not available in the general codes. The trust module evaluates the trust in sensor data.

Although this module is known as security and privacy, it seems that the privacy requirements have not been discussed in this module.

In OpenIoT, the trust module is an independent module that deals with the device trust and data trust requirements. OpenIoT draws on the spatial autocorrelation of sensors to obtain the device trust.

F. BUTLER Architecture

BUTLER is the European Union project (FP7) that is mainly focused on IoT studies [12]. BUTLER covers domain-based smart mechanisms and is defined with the aim of developing a horizontal mechanism to activate the secure and smart life applications. BUTLER mainly aims to develop an experimental and technical framework used to support IoT development procedure.

This project is the first European Union's project that focuses on the acquisition, context awareness and security of IoT.

The four main layers of this architecture cover all the necessary attributes and realization of requirements such as communication, information and context management, services, as well as system and device management. Additionally, the BUTLER system is established at three institutes known as BUTLER SmartObject, BUTLER SmartServer, and BUTLER SmartMobile.

BUTLER covers domain-based smart mechanisms and is defined with the aim of developing a horizontal mechanism to activate the development of secure and smart life applications. The BUTLER project mainly aims to develop an experimental and technical framework used to support IoT development procedure.

This framework is supposed to support smart domains by providing communication capabilities, context awareness and guarantee of security and confidentiality.

G. IoT@Work Architecture

The IoT includes new challenges such as access control that can hardly be solved by security mechanisms [13]. In fact, IoT is mainly scalable and manageable that includes an infinite number of potential objects (resources and individuals). Therefore, the IoT@Work project is introduced to defined access control.

IoT@Work is an EU FP7 project that was developed with the aim of creating IoT architecture in the field of industrial automation in 2013, under the support of AMIC during a 36-month project launched by the European Union. IoT@Work was introduced for network slices, virtual combination, resource management and security concepts. A network slice is an abstract layer between the physical vision such as network and device technologies and the application vision.

H. Microsoft Azure

In general, the Microsoft Azure architecture is focused on providing guidelines for the development of a safe device-centered and scalable solution for device connection, analysis, and integration using backend systems on the public cloud (applicable on the private cloud as well).

In addition to providing the IoT with suitable communication attributes, Azure also provides users with security and privacy. Azure is, in fact, a multi-user platform that draws on a shared infrastructure to support millions of users who intend to connect to over 83 databases at the same time. Since the infrastructure of millions of active virtual machines is shared in Azure, security and traffic confidentiality is of great importance in the network. The virtual networks of Azure draw on a combination of firewalls, access control, authentication and encryption to maintain the security of the data that are transferred by users. The Azure data center implements the policies and the security processes associated with coherent data using standardized industrial control frameworks (such as SOC2, SOC1, and ISO27001). Moreover, unbiased agents regularly certify the compliance of Microsoft with these standards (for physical and virtual aspects) [14].

V. COMPARISON OF SECURITY APPROACHES ASSOCIATED WITH THE IOT ARCHITECTURES

In this section, the capabilities and mechanisms provided by the architectures introduced in the previous section are compared and investigated to meet the security requirements of the IoT. The applications of each mechanism are also compared and investigated from the perspective of dynamic security. They are classified based on the security requirements of IoT to facilitate the comparisons, and the potentials and capabilities of each architecture are compared and investigated in each sub-section.

A. Network Security

As for network security, the eight mentioned architecture above to some extent deal with confidentiality as well as accuracy coupled with authentication as the major security requirements. It seems that accessibility is the only security requirement that is not adequately dealt with in these architectures. The IoT-A and BETaaS architectures mainly focus on key and license management concepts in the so called public key infrastructures and key exchange to guarantee confidentiality and authenticity. MGC provides a homomorphic encryption method to guarantee end-to-end security. BETaaS specifies the internal and external entities that are useful for open ecosystems. WSO2 supports openID communications. IoT-A and OpenIoT focus on a combination of existing security mechanisms such as IPsec and TLS. OpenIoT draws on secure Zigbee communication standards to address constrained resource devices. BUTLER uses key management and encoding to guarantee confidentiality and security. IoT@Work differs from other architectures and is mainly focused on authenticity, even for low-level network access, as well as availability regarding network virtualization and link failover. Anyway, IoT@Work, as compared to other architectures, attaches more importance to confidentiality. Virtual Azure networks draw on a combination of firewalls, access control, authentication, and encryption to maintain the security of data transferred by users. Context awareness and adaptability, from among other aspects of dynamic security, can have the greatest impact on these requirements. These aspects of dynamic security are also taken into account in In BeTaaS and BUTLER architectures and are applied, on a limited basis, on the mechanisms of context awareness and adaptability.

B. Identity Management

Identity management constitutes a vital part of IoT. Therefore, this concept is regarded as a security requirement in all the architectures mentioned above and various mechanisms are provided for its realization. IoT-A mainly focuses on mechanisms that provide users and service with authentication and accountability and consider authorization for services. In the field of identity management, BeTaaS manly focuses on authentication and identification but doesn't deal with other aspects of identity management such as accountability. WSO2 supports identity authorization such as SPML and rule-based accessibility and uses token-based model rather than username and password-based authentication. OpenIoT mainly focuses on CAS (central authentication server). BUTLER uses an access control factor for authorization. IoT@Work architecture provides authenticity and accountability via persistent storage for verifying the credentials, and an interface to the credential management service, for latest access and updates, and authorization. Azure uses the access control and authen-

tication procedures commonly used by Microsoft. All aspects of dynamic security affect these security requirements. IoT-A mainly focuses on the investigation of event information and dynamic review of requirements. BeTaaS and BUTLER focus on context awareness and adaptability of the approaches presented for identity management. WSO2 has to some extent dealt with context awareness as well as dynamic rule changes.

C. Privacy

Although privacy is one of the main security requirements, some IoT architectures have neither taken it into account nor provided any specific mechanism to guarantee it. The IoT-A architecture has to some extent protected privacy through pseudonimity. Pseudonimity is associated with privacy policies such as access control policies. In MGC, the privacy implementation approach is also inclined towards employment of software-defined hardware.

BETaaS applies the privacy and access control mechanisms by limiting the illegal accesses. The identity management component is responsible for managing the methodology of sensors and gateways that interact with BETaaS. However, little information has been provided in this regard. WSO2 guarantees privacy based on the identity management component. Except for establishment of privacy through central control access, data anonymity and aliases have not been accounted for in OpenIoT. In BUTLER, privacy is one of the main objectives of system security. IoT@Work indirectly guarantees privacy by providing access to entities and the so called data unlinkability, and by providing some anonymity capabilities. Azure supports privacy through Microsoft privacy standards and Microsoft security development lifecycle. Dynamic review of security requirements and adaptability are supposed to affect approaches provided for the study of requirements in this regard. IoT-A, from among other architectures, has conducted a general dynamic review of security requirements and adaptability in the approaches that deal with the requirements in this field. However, the review has not been documented.

D. Trust

IoT-A only focuses on trust at the application level. The TRA component is responsible for the establishment of trust in objects and measurement of reputation based on the recommendations and views of objects and services. The trust requirements are not dealt with in MGC. BeTaaS deals with trust requirement through a specific component that is suitable for trust. WSO2 uses WS-Trust to trust relationships. The centralized nature of this Architecture minimizes the device trust in OpenIoT. Data trust is guaranteed by computation of data reliability using spatial correlation algorithm. BUTLER uses trust and reputation component to calculate the level of trust in services base on the ratings of users. IoT@Work doesn't provide any mechanism for dealing with trust. Microsoft Azure does not provide any specific method for dealing with trust and has rather imposed some restrictions on communication acceptance by devices. All aspects of dynamic security affect these security requirements. IoT-A has to some extent dealt with dynamic review of requirements. BeTaaS and BUTLER focus on context awareness and adaptability of the approaches presented for identity management.

E. Resilience

Resilience has been regarded as an important requirement in all the architectures mentioned above. This component has been dealt with in all eight architectures. IoT-A includes a fault management model that covers all the resilience cycle phases such as fault prediction. Resilience has not been adequately dealt with in MGC. BeTaaS uses analysis techniques to identify important components and increase resilience. In WSO2, some methods have been provided for resilience. IoT@Work has paid the least attention to resilience. IoT@Work guarantees resilience through the concepts of network chip and virtualization. No specific resilience mechanism has been determined in BUTLER. OpenIoT guarantees resilience through dynamic regulation of information flows. Dynamic review of requirements and adaptability are among the factors that affect the approaches provided for dealing with requirements in this field. IoT-A, from among other architectures presented above, conducts a general dynamic review of security requirements and adaptability in the approaches that deal with the requirements in this field.

Table 3 represents the architectures provided for the IoT as well as their ability to meet the security requirements. Table 3 shows the capability level of approaches that deal with security requirement of the IoT from different aspects of dynamic security. As mentioned above and as Table 4 shows, despite the need for paying a special attention to dynamic security aspects in the IoT approaches, this issue has not been studied adequately, and achieving a trustable IoT that can be used without any specific security, privacy, trust, and accessibility concerns, requires consideration of different dynamic security aspects in the approaches that are presented to dealt with security requirements of IoT, and consideration of dynamic security-based mechanisms for dealing with the security challenges of IoT.

VI. CONCLUSION AND FUTURE WORKS

IoT has many security shortcomings that have threatened the lives and properties of human beings. Under these

Table 3. Architectures provided for IoT and comparison of their ability to meet the security requirements

Requirements	Architecture							
	IoT-A [7]	MGC [8]	BeTaaS [9]	WSO2 [10]	OpenIoT [11]	BUTLER [12]	IoT@Work [13]	Microsoft Azure [14]
Network security								
Confidentiality	Y	Y	Y	Y	Y	Y	Y	Y
Integrity	Y	P	Y	P	Y	Y	N	Y
Authenticity	Y	Y	Y	N	Y	Y	Y	Y
Availability	P	N	N	N	N	P	P	P
Identity management								
Authentication	Y	Y	Y	Y	Y	Y	Y	Y
Authorization	Y	Y	Y	P	Y	Y	Y	Y
Accountability	N	N	N	N	N	N	Y	N
Revocation	Y	N	N	N	N	N	Y	N
Privacy								
Data privacy	P	N	N	N	N	Y	P	Y
Anonymity	N	N	N	N	N	N	Y	N
Pseudonymity	Y	N	N	N	N	N	Y	N
Unlinkability	Y	N	N	N	N	P	N	N
Trust								
Device trust	Y	N	Y	P	Y	N	N	P
Entity trust	Y	N	N	N	Y	P	N	P
Data trust	N	N	Y	Y	N	P	N	P
Resilience								
Robustness	Y	P	Y	P	N	Y	P	Y
Resilience	Y	P	Y	P	Y	Y	P	Y

Y: yes, N: no, P: partly.

circumstances, the establishment of security in a global network of objects with specific limitations and attributes that communicate with people and one another in specific ways is naturally much more complicated. New environmental conditions as well as dynamic nature and different characteristics of devices are among the factors that have placed the IoT security at the center of attention, and instigated us to provide suitable architecture and security mechanisms for that. In the present study, the new security requirements of IoT are defined under the light of dynamic security and the need for employment of dynamic security in the IoT is investigated.

The uncontrolled, dynamic, and heterogeneous environment of IoT, its ULS, and the objects with constrained resources, have drawn the attention of many researchers towards the IoT since misuse of this internet can threaten the human lives and properties. In the present study, the main attributes and the security requirements of IoT are introduced, and the necessity of approaches presented for IoT are discussed from the perceptive of dynamic security. Therefore, dynamic security can be defined as security approaches for meeting the security requirements of IoT, which deal with information associated with events and dynamic review of requirements and provide adaptable security and context-related data for IoT. Investigation of the security approaches of the most important architectures provided for the IoT showed that despite the significant importance of security in the IoT and considering the need for employment of dynamic security approaches, these issues have not been studied adequately and that the IoT is currently in desperate need for dynamic security approaches.

In the further studies, attempts will be made to provide approaches based on dynamic security to meet the security requirements of IoT. Provision of a dynamic security approach for management of trust is a priority in the further studies.

Table 4. The applicability level of approaches provided for dealing with the security requirements of IoT from different aspects of dynamic security

	Architecture							
	IoT-A [7]	**MGC [8]**	**BeTaaS [9]**	**WSO2 [10]**	**OpenIoT [11]**	**BUTLER [12]**	**IoT@Work [13]**	**Microsoft Azure [14]**
Network security								
Context-awareness								
Confidentiality	N	N	Y	N	N	Y	N	N
Integrity	N	N	P	N	N	P	N	N
Authenticity	N	N	Y	N	N	P	N	N
Availability	N	N	N	N	N	N	N	N
Adaptation								
Confidentiality	Y	N	Y	N	N	Y	N	N
Integrity	Y	N	P	N	N	P	N	N
Authenticity	Y	N	Y	N	N	P	N	N
Availability	P	P	N	N	N	N	N	N
Identity management								
Event analysis								
Authentication	Y	N	N	Y	N	N	N	N
Authorization	P	N	N	P	N	N	N	N
Accountability	N	N	N	N	N	N	N	N
Revocation	P	N	N	N	N	N	N	N
Context-awareness								
Authentication	N	N	P	Y	N	P	N	N
Authorization	N	N	P	P	N	P	N	N
Accountability	N	N	N	N	N	N	N	N
Revocation	N	N	N	N	N	N	N	N
Dynamic requirement analysis								
Authentication	N	N	N	Y	N	N	N	N
Authorization	N	N	N	P	N	N	N	N
Accountability	N	N	N	N	N	N	N	N
Revocation	P	N	N	N	N	N	N	N
Adaptation								
Authentication	N	N	Y	Y	N	P	N	N
Authorization	N	N	P	P	N	P	N	N
Accountability	N	N	N	N	N	N	N	N
Revocation	N	N	N	N	N	N	N	N
Privacy								
Dynamic requirement analysis								
Data privacy	P	N	N	N	N	N	N	N
Anonymity	N	N	N	N	N	N	N	N
Pseudonymity	P	N	N	N	N	N	N	N
Unlinkability	P	N	N	N	N	N	N	N
Adaptation								
Data privacy	P	N	N	N	N	N	N	N
Anonymity	N	N	N	N	N	N	N	N
Pseudonymity	P	N	N	N	N	N	N	N
Unlinkability	P	N	N	N	N	N	N	N

Table 4. Continued

	Architecture							
	IoT-A [7]	**MGC [8]**	**BeTaaS [9]**	**WSO2 [10]**	**OpenIoT [11]**	**BUTLER [12]**	**IoT@Work [13]**	**Microsoft Azure [14]**
Trust								
Event analysis								
Device trust	P	N	N	N	N	P	N	N
Entity trust	N	N	N	N	N	N	N	N
Data trust	N	N	N	N	N	N	N	N
Context-awareness								
Device trust	N	N	P	N	N	P	N	N
Entity trust	N	N	P	N	N	Y	N	N
Data trust	N	N	P	N	N	P	N	N
Dynamic requirement analysis								
Device trust	P	N	N	N	N	N	N	N
Entity trust	P	N	N	N	N	P	P	P
Data trust	P	N	N	N	N	N	N	N
Adaptation								
Device trust	N	N	P	N	N	P	N	N
Entity trust	N	N	Y	N	N	P	N	N
Data trust	N	N	P	N	N	P	N	N
Resilience								
Dynamic requirement analysis								
Robustness	P	N	P	N	N	P	N	P
Resilience	P	N	P	N	N	P	N	P
Adaptation								
Robustness	P	N	P	N	N	P	N	P
Resilience	P	N	P	N	N	P	N	P

Y: yes, N: no, P: partly.

REFERENCES

1. M. Hafner and R. Breu, *Security Engineering for Service-Oriented Architectures*. Heidelberg: Springer, 2009.
2. J. Ramachandran, *Designing Security Architecture Solutions*. Hoboken, NJ: John Wiley & Sons, 2002.
3. R. Kanneganti and P. Chodavarapu, *SOA Security*. Greenwich, CT: Manning Publications, 2008.
4. N. C. Damianou, "A policy framework for management of distributed systems," PhD dissertation, University of London, UK, 2002.
5. A. K. Bandara, E. C. Lupu, J. Moffett, and A. Russo, "A goal-based approach to policy refinement," in *Proceedings of the 5th IEEE International Workshop on Policies for Distributed Systems and Networks*, Yorktown Heights, NY, 2004, pp. 229-239.
6. D. Miorandi, S. Sicari, F. De Pellegrini, and I. Chlamtac, "Internet of Things: vision, applications and research challenges," *Ad Hoc Networks*, vol. 10, no. 7, pp. 1497-1516, 2012.
7. F. Carrez, M. Bauer, M. Boussard, and N. Bui, "Final architectural reference model for the IoT v3.0," 2013, http://www.meet-iot.eu/deliverables-IOTA/D1_5.pdf.
8. Secure Internet of Things Project, http://iot.stanford.edu/.
9. Seven Framework Programme, "BETaaS Architecture: building the environment for the things as a service," 2014, http://www.betaas.eu/docs/deliverables/BETaaS%20-%20D3.1.2% 20BETaaS%20Architecture%20v1.0.pdf.
10. P. Fremantle, "A reference architecture for The Internet of Things," WSO2, 2015, https://wso2.com/wso2_resources/wso2_whitepaper_a-reference-architecture-for-the-internet-of-things.pdf.
11. National University of Ireland Galway, "OpenIoT Project: Open source blueprint for large scale self-organizing cloud

environments for IoT applications," 2015, http://cordis.europa.eu/project/rcn/101534_en.html.
12. B. Copigneaux, F. Clari, J. Galinowski, A. Ramakrishnan, D. Preuveneers, C. Gotze, S. Poilinca, F. Sottile, F. Rizzo, A. Andrushevich, et al., "D5.2 BUTLER final platforms and quality assessment," 2011, https://cordis.europa.eu/docs/projects/cnect/1/287901/080/deliverables/001-287901BUTLERD521.pdf.
13. Siemens Aktiengesellschaft, "IoT@Work: Internet of Things at Works," 2013, http://cordis.europa.eu/project/rcn/95348_en.html.
14. Microsoft, "Microsoft Azure IoT Reference Architecture," 2016, http://download.microsoft.com/download/A/4/D/A4DAD253-BC21-41D3-B9D9-87D2AE6F0719/Microsoft_Azure_IoT_Reference_Architecture.pdf.

Seyyed Yasser Hashemi

Seyyed Yasser Hashemi is a Ph.D student of Computer science and Engineering Department, Shahid Beheshti University of Iran. He is working on his Ph.D thesis named "Dynamic Security Solutions for Internet of Things with Emphasis on Dynamic Trust Management" and has published several conference and journal papers. His research interests are in Internet of Things, trust management and dynamic security. He is currently a member of faculty and lecturer of Miandoab Branch, Islamic Azad University of Iran.

Fereidoon Shams Aliee

Fereidoon Shams Aliee has received his Ph.D in software engineering from Department of Computer Science, Manchester University, UK, in 1996 and his M.S. from Sharif University of Technology, Tehran, Iran. His major interests are in Software Architecture, Enterprise Architecture, Service Driven Architecture, Agile Methodologies, Ultra-Large-Scale (ULS) Systems, Ontological Engineering and has published more than 100 journal and conference papers. He is currently Associate Professor of Computer Science and Engineering Department, Shahid Beheshti University of Iran. Also, he is heading two research groups, namely ASER (Automated Software Engineering Research) and ISA (Information Systems Architecture) at Shahid Beheshti University.