International Journal of Security and Its Applications
Vol. 10, No. 6 (2016) pp.13-22
http://dx.doi.org/10.14257/ijsia.2016.10.6.02

A First Step Towards Security for Internet of Small Things

Namhi Kang

Duksung Women's University, Seoul, Korea
Email: kang@duksung.ac.kr

Abstract

Industrial and research organizations expect that tens of billions of daily life objects can communicate with each other in the near future for realizing a hyper connected society that is called Internet of Things (IoT). They also agree that security is one of the most important concerns to provide smart and intelligent services successfully in the IoT. The author notes in this paper that the first step towards a secure IoT is the initial configuration of connected objects and networks in a secure fashion. As a solution, this paper proposes a secure configuration (i.e., bootstrapping) scheme for resource constrained devices such as sensors or actuators. The scheme is activated when a new object (or node) initially installs and re-installs to a network that is currently in operation. The method is suitable for a scenario, where resource constrained small things are interconnected with each other and thus formed Internet of Things.

Keywords: *Secure bootstrapping, Reconfiguration, Internet of Things, Resource constrained devices, Pre-shared Key*

1. Introduction

These days, Internet of Things (IoT) has received increasing attention, where a rapidly growing number of small things around us, that is called everyday objects, are intended to connect to the Internet. In effect, a lot of devices have already been interconnected with other devices thanks to both cost efficient access networking technologies such as WiFi, Bluetooth, and ZigBee and the popularity of using various smart services and devices such as smart phone, smart car and smart TV. Further, smart and intelligent IoT world can be realized in the near future, where interconnected various things automatically discover a target service for humans and intelligently support the service by cooperating with nearby various things without human intervention. Thus, IoT can be used in wide industry fields such as industrial control, smart home and building services, healthcare services and several other industries [1][2].

Security and privacy are highly required to provide service successfully in the Internet. Under the same aspect, IoT services must consider not only general security services such as confidentiality, integrity, availability, data access control, and authentication, but also an efficient way to protect against various threats and attacks that can occur. For instance, a healthcare service that transmits sensitive personal information must carefully consider security and privacy [3]. The author notes here that the first step towards a secure IoT is the initial configuration of nodes (*i.e.*, small things) and networks in a secure fashion.

This paper presents an efficient and secure scheme to configure a resource constrained IoT device. Especially, the author focuses on secure key configuration. Most IoT devices are embedded in objects of daily life and operate with minimal resources (*i.e.*, 8 bit processing microcontrollers with limited amounts of memory). The network is also constrained one (*e.g.*, 6LoWPAN having high packet error rates and a typical throughput of 10s of kbit/s). The paper assumed that communications of IoT nodes are based on TCP/IP protocols and the nodes use the constrained application protocol (CoAP) over 6LoWPAN network [4][5].

ISSN: 1738-9976 IJSIA

Pre-shared key (PSK) based secure schemes are well known and widely used for various security services in Internet. In particular, resource constrained things prefer to use PSK based secure scheme since it is computing efficient. All such schemes strictly assume that PSK is only known to the two communication entities involved in current security service. As a result, security of the schemes is compromised if the assumption is broken [6]. However, it is still not clear how PSK of resource constrained things can be initially configured in a secure manner.

As a conceptual solution, this paper presents an initial setup method that might be a part of secure bootstrapping scheme. Like the author's previous work proposed in [6], the basic idea of the proposed scheme is conceptualized from the lock of a suitcase. Simple and default password such as '0000' or '1234' is the initial setup on a lock of suitcase when sold . The owner can change the password after purchasing it. In this paper, similarly, initial key of a node is configured (or imprinted) by installer (or manufacturer) during the enrollment phase (see Figure 3). Thereafter, when the node joins to an existing network, the key (*i.e.*, PSK) can be securely reconfigured.

The proposed scheme does not cover all operations of secure bootstrapping for IoT networks, but it is intended to securely support self-reconfiguration of the pre-installed temporary key of joined nodes. Depending on service, either a controller or an administrator configures the nodes by using the newly configured key. The configurations include application setting and network layer settings such as domain name, default gateway, and proxy. The setting information should be delivered securely for making further service secure.

However, unlike the author's previous work (*i.e.*, [6]), this paper proposes a way to configure and reconfigure PSK utilizing out-of-band (OOB) channel of small things. Audio signal, light (LED or displayed visual code) and NFC can be a proper candidate for the OOB channel. For example, a thing, which contains speaker and mouse, can use a wireless communication technology such as Bluetooth or ZigBee in order to connect to Internet. In such a case, the thing can use Bluetooth as its in-band channel, whereas audio channel can be used as an OOB channel.

The remainder of the paper is organized as follows. Section 2 presents related research work, and Section 3 describes problems and limitations considered in this paper. In Section 4, the author proposes a secure bootstrapping scheme. Section 5 analyzes the scheme, and finally, the author concludes the study in Section 6.

2. Related Work

The author has already proposed an initial method to configure (and/or reconfigure) a security key for a resource constrained node when it joins to a network that is currently in operation [6]. In the scheme, as the author described in Section 1, an initial key of a node is configured by either installer or manufacturer during the enrollment phase statically or manually. Thereafter, a new key (*i.e.*, PSK) can be securely reconfigured by either owner or administrator when the node joins to an existing network.

The scheme is very simple thus can be used efficiently in practice. However, there is a serious problem in the fourth flight of the protocol. A reconfigured security key (*i.e.*, new PSK configured by owner) can be revealed if the installer or manufacturer who configured the initial key (*i.e.*, "0000" in the example of suitcase) is nearby the newly joining node in the bootstrapping phase. That is due to the fact that anyone in the coverage of wireless communication can hear the data. If the installer can hear the data, he can decrypt the data by using the initial key. To mitigate the vulnerability, a physical security tool such as Faraday cage can be used as discussed by [7]. However, method is expensive to deploy. This problem motivates the author to enhance the previous scheme by utilizing strong OOB channel.

Several technologies have been proposed in the literature to support secure pairing between devices in machine to machine communication (for example, see [15] for more information). In such technologies, various OOB channels were used in the authentication phase.

Jennings proposed a scheme to configure a new device and register the device to the controller using a QR code [8]. In his scheme, as shown in Figure 1, an introducer, which might be a smart phone, of a network reads QR code printed on or contained in the box of a device. That is, when the device is installed, the introducer derives secret parameters, such as OTP and secret key, of the device by scanning the QR code (message 1 in Figure 1). OTP is a one time password generated by a manufacturer for the device registration and the secret is the secret value generated by a manufacturer for enabling the communication between the device and the controller. Next, the introducer delivers the network information of the network and OTP used by the device to the transfer agent which is handled by the manufacturer (message 2). The introducer transmits the secret to the controller (message 3). When the device is booted up for the first time and the network connection is made, it connects to the transfer agent. The transfer agent transmits the network information of the controller to the device (message 4). Since the device knows the network information of the controller, the device can communicate with the controller directly in the subsequent device operation (message 5).

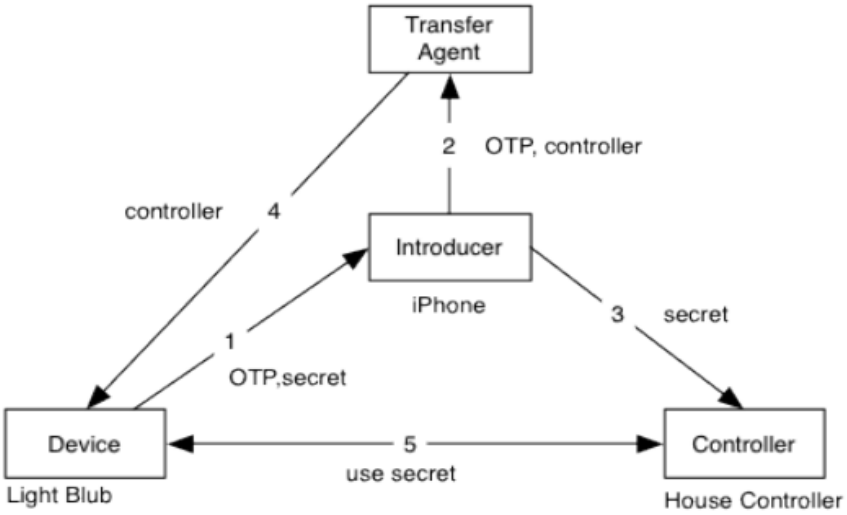

Figure 1. Jennings's Scheme [8]

This system can be used for a device which has limited memory and processing capability. The network or the power is not required when installing the device, and if OTP used for authentication has been used already, the installer can detect such fact. The user's involvement has been minimized to allow the user to register a device by scanning a QR code, but the authentication information of the QR code is provided without encryption process so it may be vulnerable to security and the applicable environments are limited.

MVSec was proposed to support a secure key agreement between a vehicle and a mobile device by utilizing light and audio sound as an OOB channel [9]. The primary objective of MVSec is to allow the user to pair his/her vehicle and smartphone securely while responding to the man-in-the-middle attack from the attacker. When light is used as the OOB channel, the pairing with the vehicle is carried out by inserting the smartphone into the glove box inside the vehicle and transmitting a signal through the flicker of light. Also, when audio sound is used as the OOB channel, the pairing is carried out by

exchanging a warning sound between the vehicle and the smartphone. This scheme is secure and efficient but is highly tailored not for general IoT environments but for vehicular environments.

3. Problem Statement

In secure bootstrapping and authentication system in Internet, PSK based primitive is widely used. This is mainly due to the fact that PSK based secure scheme is much more efficient than public key based schemes. As we describe in Section 1, however, it is questionable how PSK of resource constrained thing can be securely configured.

Typically, things used for IoT may be manufactured and installed by different subjects (*i.e.*, simple persons) [10]. That is, in general situation, a system administrator may make orders to several different installers. After that, each of the installers purchases one or more different set of things from one or more different manufacturers. It is also unlikely that a single subject installs all nodes used for a large application domain (*e.g.*, all nodes in huge building). That is, many installers and manufacturers may be involved depending on deployed application service. In such case, it is a matter for consideration whether all installers and manufacturers can be trusted or not (in other words, a matter of trust relationship). Figure 2 shows the trust relationship among participants.

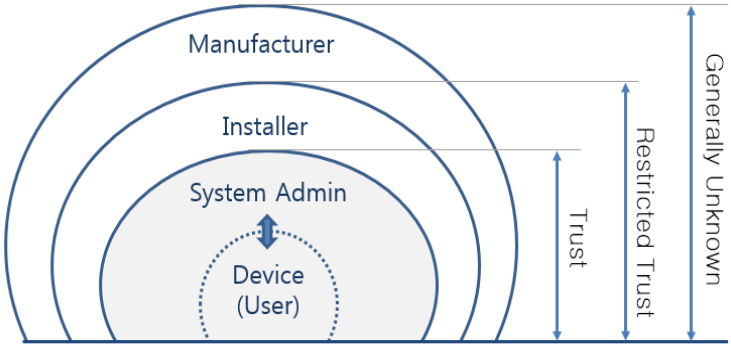

Figure 2. Trust of Participants

This paper considers a scenario, where nodes are initially configured by an installer (or a manufacturer in some cases) during enrolment phase (or manufacturing/factory configuration phase) as shown in Figure 3. If secure credential including PSK is required to be configured in this phase, the trust between installer (or manufacturer) and system administrator is extremely important. However, this is not an easy process because manufacturer, installer and service provider do not share a tight and trust relationships in general cases. Even if the case is properly settled, there might be several secure threats and vulnerabilities to be handled.

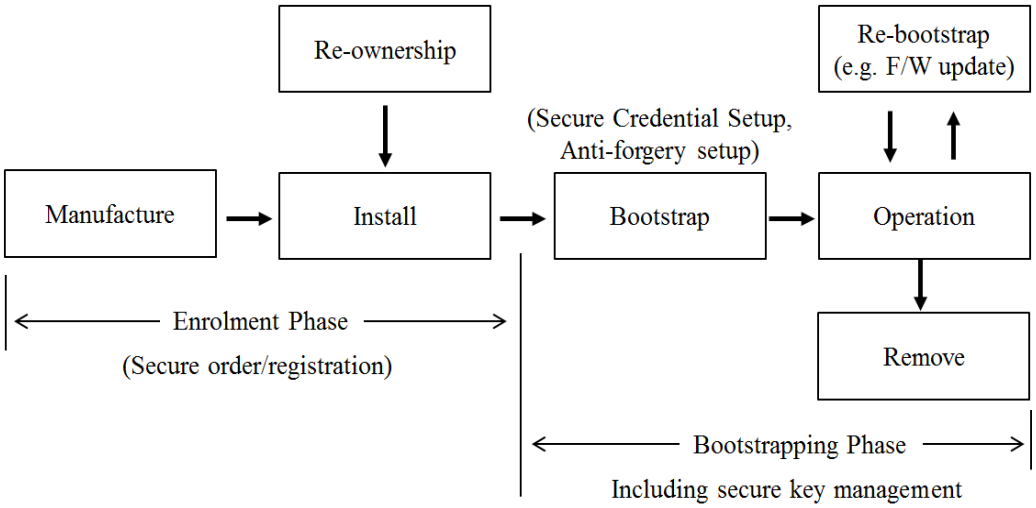

Figure 3. Two Phases in Secure Bootstrapping

Secure bootstrapping is regarded as a difficult problem in IoT environments. This is mainly because a lot of daily-life things that tend to connect to the internet are resource constrained. IETF classified constrained devices with consideration of data size and code size as shown in the following Table 1 [11].

Table 1. Classes of Constrained Devices

Classification	data size (i.e. RAM)	code size (i.e., Flash ROM)
Class 0	<< 10 KiB	<< 100 KiB
Class 1	~ 10 KiB	~ 100 KiB
Class 2	~ 50 KiB	~ 250 KiB

Devices in Class 0 are highly constrained (*e.g.*, small motes). They cannot communicate with the Internet directly, so Class 0 devices have to get help from infra devices such as a proxy or gateway [16]. In addition, to support energy efficient communication, small things in the IoT may adopt low power wireless communication specifications like IEEE 802.15.4 [12].

In order to make a practical and efficient method, the proposed method requires only a single cryptographic primitive that is AES with 128bits length of key [13]. All cryptographic primitives cannot be installed on resource restricted devices, mainly because of limited size of flash or RAM. For this reason, CoAP also does not consider all modes of cryptographic operations in DTLS which is a recommended secure protocol for CoAP applications [4]. In case of establishing a CoAP session using a pre-shared key mod of DTLS, implementation of cipher suite TLS_PSK_WITH_AES_128_CCM_8 specified in [17] is mandatory.

In addition, user-device interfaces of resource constrained device are not enough for doing configurations manually by person (*i.e.*, inadequate or even no input/output equipment such as display or keyboard).

To solve the problem, the author proposes a scheme to securely re-configure a symmetric key (*i.e.*, pre-installed key in manufacturing phase) automatically upon joining to an existing network. After the secure configuration phase, an installer (or manufacturer)

cannot read, modify, and insert any communication data even though the installer did initial pre-setup of secure credential of the communicating nodes. The author does not assume that a system administrator trusts an installer (or manufacturer) even though the administrator gives orders for the installer. This is because the trust and responsibility of the installer, who buys and install devices, are different from those of system administrator.

The following transactions are done prior to the secure key reconfiguration (*i.e.*, procedures in the enrolment Phase).

1. System administrator makes orders and requests initial setup of devices to an installer. Pre-setup information is a set of values that include ID and network socket ID (*NID*) of controller for each of the devices, and a temporary key used as an initial key (*IK_N*). All devices handled by a single installer may share the same *IK_N*.

2. System administrator also stores the same initial information for each of nodes in authentication server. A controller can perform operations of an authentication server in case of a small network.

3. Installer purchases devices and then configures the information requested by the administrator in doing installation phase. Some of the information for a node may be pre-configured by manufacturer.

4. When a node joins to network, it knows the *NID* of its associated controller with which it can communicate. Also, authentication server has lists including node ID and pre-installed key for new nodes.

5. PSK reconfiguration phase can be then started.

4. Secure Bootstrapping

There are three message exchanges between a new node SBI (i) and network node (s) (*i.e.*, SBR (c) and SBS (s)). A controller SBR (c) may include functions of both SBR (c) and SBS (s) depending on the size of application domain or the ability of SBR (*i.e.*, computing power and memory). Mutual authentication and PSK reconfiguration procedures are shown in Figure 4.

When a new node SBI (i) joins an existing network, it generates a random number *RNi* and sends it with its identifier *IDi* to his controller SBR (c) over OOB channel. QR code, NFC, light, audio can be used as the OOB CH that is relatively more secure than wireless channels with wide communication coverage such as WiFi, Bluetooth, *etc*. This is due to the fact that user can make sure that there is no attacker (*i.e.*, man in the middle attacker) nearby him.

In Figure 4, the author uses the notation *AE* (Authenticated Encryption), but the two nodes transmit plaintext depending on the secure strength of the OOB CH. The *NID* of SBR (c) (*i.e.*, IP address and port number) has been pre-configured by installer of the SBI (i) in the enrolment phase.

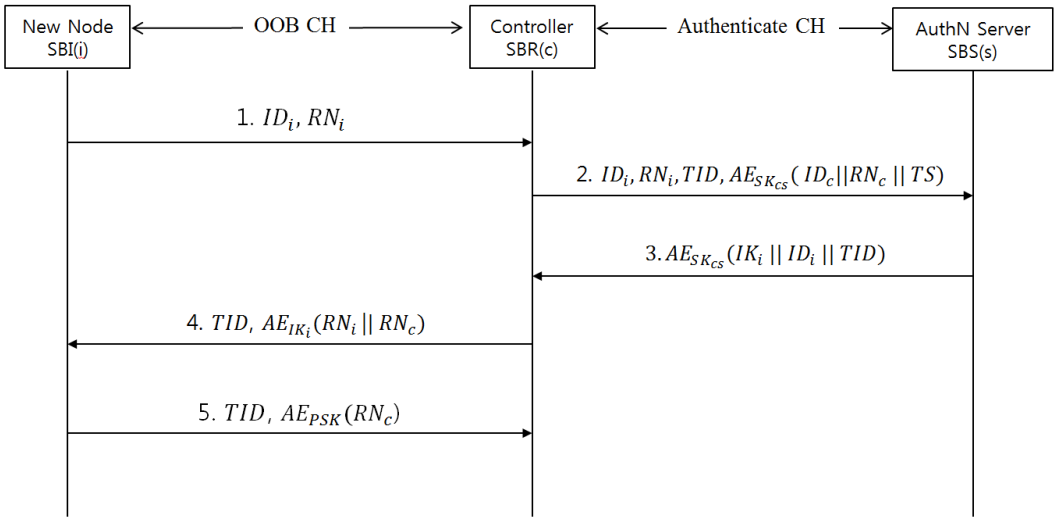

Figure 4. Secure Bootstrapping Procedures

Upon receiving the message, SBR (c) generates a random number *RNc* and a sequence number used as a transaction ID (*i.e.*, *TID*). Then it sends the values with his *IDc*, time stamp (*TS*) and the message received from SBI (i) to the authentication server SBS (s). The encrypted TS allows SBR (s) to derive the valid time of key and verify the freshness of the message that arrived. To protect against DoS attack lunched by sending lots of bogus packets, IDc is also encrypted.

The authentication server SBS (s) first decrypts the arrived message then checks the *IDc*. The SBS (s) then discovers the *IKi* for node *IDi* in its secure repository. In the enrollment phase, all *IKs* for nodes installed are preconfigured. SBS (s) now can derive a new PSK for the node SBI (i) and replace the *IKi* with the *PSKi*, where the PSK for SBI (i) is derived as follows.

$$PSKi = E(RNi \oplus RNc, IKi)$$

After the reconfiguration of PSK for node SBI (i), it encrypts the concatenation value of IKi, TS and TID with the symmetric key SKcs which is a shared key between SBS (s) and SBR (c). Then it sends the encrypted data to SBR (c). This is because SBR (c) does not have the key *IKi* at this moment.

On receiving the encrypted value from SBS (s), SBR (c) can authenticate that the data is correctly generated by SBS (s) by verifying *TS and TID* in the encrypted data. Now, SBR (c) can know the key *IKi*. Then, SBR (c) encrypts the concatenation value of *RNi* and *RNc* with the key *IKi*. He sends both the encrypted value and *TID* to SBI (i). Note that, SBR (c) must not transmit the derived PSK over the public network.

SBI (i) can verify the authenticity of SBR (c) by using the decrypted *RNi* value from the received message. Finally, SBI (i) can configure his PSK thereafter sending the encryption value of *RNc* with the new key PSK to SBR (c) for the authenticity validation. SBI (i) derives a session key *SKi* from the PSK and then reconfigures its secure credential.

5. Security Analysis

In order to analyze the security of the proposed scheme, the security analysis on replay attack, impersonation attack and man-in-the-middle attack are given in this section. This paper assumes that data in the flight 1, 4 and 5 is transmitted over OOB channel which is

regarded as a strong OOB discussed in [9]. Therefore, an attacker can only eavesdrop, intercept, and modify data in the flight 2 and 3.

Even if an attacker intercepts a data transmitted in the flight 2 and 3, and tends to make a replay attack after a certain period of time, it cannot expect and modify the *RNc*, which is a part of calculating PSK. Also, there is a time stamp in the encrypted part of the data. Therefore, the proposed scheme protects against replay attack.

In the proposed scheme, impersonation is difficult. As stated in the introduction, it is difficult to make an attack on strong OOB channel without a special device. For example, if we use NFC as the OOB, it has a short communication radius within 10 cm. Two devices are located closely so that these devices can check the access of an attacker. Also, even if an attacker impersonates as a controller, the attacker cannot know the symmetric key which the controller and the authentication server have shared in advance, so the attacker cannot transmit a correct value encrypted in the flight 2 and 3.

An attacker may exist between the controller and the authentication server. However, the controller and the authentication server carry out encryption and decryption using the shared symmetric key in advance in both flights 2 and 3, so it is possible to protect against a man-in-the-middle attack.

The proposed scheme can be implemented by using only a single cryptographic primitive AES [13] which is used for secure bootstrapping in the PSK reconfiguration phase. Single cryptographic primitive implementation is rationally suited for the scenario where applications or services require a secure session (confidentiality and integrity of data) in IoT. Because small devices limited with low computing power and little storage are major entities in IoT. According to a full bootstrapping policy, the PSK can be used for mechanisms of session key derivation and/or entity authentication.

As discussed in ESP-PSK [14], it goes without saying that a single cryptographic primitive may not support extensible security services such as identity protection, perfect forward secrecy and others. However, small devices consisting of Internet of Things might not support all of security services inherently. Service developer should therefore define a scope of his service strictly and consider trade-off between capability and security.

6. Conclusion

The PSK based method is appropriate for the IoT environments consisting of lightweight devices since this method uses less computing time and energy than the method to set the session key based on the public key algorithm. An essential prerequisite for the PSK based method is that PSK should have been configured between the main agents of communication safely in advance. To achieve the prerequisite, this paper proposes a secure key setup method that is a part of secure bootstrapping scheme. Currently, the author is designing and implementing the proposed scheme by using various types of OOB channel. In particular, the author believes that NFC and audio are the best solution for resource constrained IoT devices. That is mainly because the two interfaces are widely used for a lot of daily objects and thus the cost is very low.

References

[1] J. Gubbi, R. Buyya, S. Marusic and M. Palaniswami, "Internet of things (IoT): a vision, architectural elements, and future directions", Future Generation Computer Systems, vol. 29, no. 7, **(2013)**, pp. 1645–1660.
[2] C. Perera, C. H. Liu, S. Jayawardena and M. Chen, "A Survey on Internet of Things From Industrial Market Perspective", IEEE Access, vol. 2, **(2013)**, pp. 1660-1679.
[3] J. Park, H. Kwon and N. Kang, "IoT-Cloud Collaboration to Establish a Secure Connection for Lightweight Devices", Springer Wireless Net-works, **(2016)**, pp. 1-12(on-line published).
[4] Z. Shelby, K. Hartke and C. Bormann, "The Constrained Application Protocol (CoAP)", IETF Standard, RFC 7252, **(2014)**.

[5] S. L. Keoh, S. S. Kumar and H. Tschofenig, "Securing the Internet of Things: A Standardization Perspective", IEEE Internet of Things Journal, vol. 1, no. 3, **(2014)**, pp. 265-275.
[6] N. Kang, S. Oh and S. Yoon, "Secure initial-key reconfiguration for resource constrained devices", IETF Standard, Internet Draft, **(2014)**, February.
[7] O. BERGMANN, S. GERDES and C. BORMANN, "Simple keys for simple smart objects", In Workshop on Smart Object Security, **(2012)**.
[8] C. Jennings, "Transitive Trust Enrollment for Constrained Devices", IETF Standard Internet Draft, **(2012)**.
[9] J. Han, Y. H. Lin, A. Perrig and F. Bai, "MVSec: Secure and Easy-to-Use Pairing of Mobile Devices with Vehicles (CMU-CyLab-14-006)", **(2014)** May.
[10] O. Garcia-Morchon, S. Kumar, S. Keoh, R. Hummen and R. Struik, "Security Considerations in the IP-based Internet of Things", IETF Standard, Internet Draft, **(2013)**, September.
[11] C. Bormann, M. Ersue and A. Keranen, "Terminology for Constrained-Node Networks", IETF Standard, RFC 7228, **(2015)** June.
[12] N. Kang, J. Park, H. Kwon and S. Jung, "ESSE: Efficient Secure Session Establishment for Internet-Integrated Wireless Sensor Networks", Journal of Distributed Sensor Networks, vol.501, 393754, **(2015)**, pp. 1-11.
[13] National Institute of Standards and Technology, "Specification for the Advanced Encryption Standard (AES)", Federal Information Processing Standards (FIPS) 197, **(2001)** November.
[14] F. Bersani and H. Tschofenig, "The EAP-PSK Protocol: A Pre-Shared Key Extensible Authentication Protocol (EAP) Method", IETF Standard, RFC 4764, **(2007)** January.
[15] S. Mirzadeh, H. Cruickshank and R. Tafazolli, "Secure Device Pairing: A Survey", IEEE Communication Surveys and Tutorials, vol. 16, no. 1, **(2014)**, pp. 17-40.
[16] H. Kwon, J. Park and N. Kang, "Challenges in Deploying CoAP over DTLS in Resource Constraint Environments", Lecture Notes in Computer Science, vol. 9503, **(2016)**, pp. 269-280.
[17] D. McGrew and D. Bailey, "AES-CCM Cipher Suites for Transport Layer Security (TLS)", IETF Standard RFC 6655, **(2012)** July.

Authors

Namhi Kang, he received B.E. and M.S. degrees in electrical and communications engineering from Soongsil University, Korea in 1999 and 2001, respectively. He received a Ph.D. degree in information and communications engineering from Siegen University, Germany, in December 2004. In 2005 he joined Ubiquitous Network Research Center, DASAN Networks, where he was a senior engineer. Since 2009, he has been a professor in the Department of Digital Media, Duksung Women's University in Seoul Korea. His research interests include network technology and security in wired and wireless networks, and the design of communication protocols for future oriented networks.