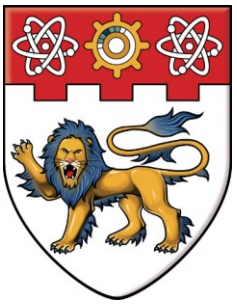

NANYANG
TECHNOLOGICAL
UNIVERSITY

DESIGN OF SECURITY MECHANISM FOR CYBER-PHYSICAL SYSTEMS

CHEN SHUO

School of Electrical & Electronic Engineering

Nanyang Technological University

2014

DESIGN OF SECURITY MECHANISM FOR CYBER-PHYSICAL SYSTEMS

CHEN SHUO

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Master of Engineering

2014

ACKNOWLEDGEMENTS

My sincere gratitude to my supervisor, *Prof. Maode Ma* from Nanyang Technological University (NTU), for his constructive guidance, technical discussions and constant encouragement throughout the course of my Master research works. The profound knowledge in network security, fluent writing skills, distinct analytical skills and high research integrity possessed by *Prof. Ma* have inspired my enthusiasm in research field. *Prof. Ma* is not only a supervisor to my research work but also a mentor of my life, who always provides me with priceless opinions for career and life. It is quite delightful and lucky to have a chance to work under *Prof. Ma*'s supervision.

Great thanks to my other group members (*Mr. Cao Jin, Mr. He Tuo and Miss. Qiu Yue*) and my close friends (*Mr. Yew Kwang sing, Mr. Liu Pan, and Mr. Gu Chenjie*) for their all-round support and friendships.

I dedicate this dissertation to my parents and sister. Their precious love and support are always the motivation of my life.

TABLE OF CONTENTS

Design of Security Mechanism for Cyber-Physical Systems

Chen Shuo
School of Electrical and Electronic Engineering,
Nanyang Technological University

SUMMARY

As the next generation of network, the emerging cyber-physical systems (CPS) are going to connect all of the objects of physical and cyber world. Machine to machine (M2M) communication is a fundamental part of the CPS which utilizes both wireless and wired systems to monitor physical or environmental conditions and exchange the information among different systems without direct human intervention. While being a promising technology which has potentials to become a market-changing force for a wide variety of real-time monitoring applications, M2M communication still faces lots of threats. Even though many solutions have been found to address the security issues of the M2M communication in the literature, there are some security vulnerabilities that yet to be solved. In the thesis, we first introduce the background, architecture security threats and security requirements of M2M communication in CPS. Subsequently, we review the important security solutions for M2M domain proposed in recent literatures from three aspects: detection, authentication and key management, explore the emerging technologies -- IPv6 over Low power Wireless Personal Area Networks (6LoWPAN) and Constrained Application Protocol (CoAP), which would be applied to M2M communication in the future and review the current security solutions for 6LoWPAN in the literature as well. Then we

propose three authentication schemes for M2M communication: a dynamic-encryption authentication scheme for M2M security in cyber-physical system, an authentication scheme with identity-based cryptography (IBC) for M2M security in cyber-physical systems and an authentication scheme for multi-domain M2M security in cyber-physical systems. The proposed dynamic-encryption scheme could avoid directly stealing and modifying of the mobile devices' and the sensors' ID. The dynamic-key generation mechanism in dynamic-encryption scheme could not only provide a reliable one-time-password among M2M service provider (MSP), mobile devices and sensor nodes but also save the computing resource of the sensor nodes. The application of integrated IBC in the authentication scheme with IBC could achieve the message authentication without key escrow problem and reduce the threat of compromise attack to a great extent. The regular updating of secret key could also make the key guessing attack meaningless. In the third scheme, the communication scenario in which the sensor nodes from different domains communicate with each other without human intervention has also been considered. Our analysis indicates that the mutual authentication and the ability of withstanding multiple attacks could be accomplished by the proposed solutions and the balance between system performance and security has been achieved.

Keywords—Security; M2M; CPS; Authentication; IBC

Chapter 1. INTRODUCTION

1.1 Background

The Internet has made the world become smaller and smaller. Now, over the Internet, huge amount of information can be shared among all the people around the world very quickly. However, there are still a serious gap existing between the cyber world and the physical world [1]. The emerging CPS is going to fill this gap and connect all of the objects of physical and cyber world. Over the CPSs, the connected objects and items which are capable to report their locations and states will be able to exchange information among each other automatically without human operation. Since our lives are becoming increasingly interlinked by mobile phones, networked appliances, and other intelligent devices [2], the CPSs could make our lives more convenient and comfortable. A general architecture of a CPS can be shown as follow in Figure 1.

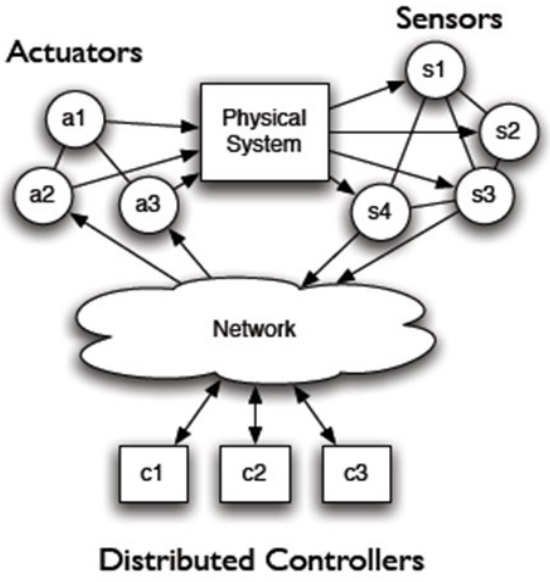

Figure1. General Architecture of a CPS [3]

There will be three major types of the components to form three tiers in a CPS. One type of the components is a group of sensors to form an environmental tier. The second type of the components is the actuator, which can form a service tier. And the last type is the controller forming the control tier. The sensors collect information from physical system and then send the information to the network which is handled by the distributed controllers in the cyber world. After processing the information, the controllers communicate with the actuators to issue appropriate operation commands. Then, the actuators will act to impose the physical world through activating the related operations and generate feedback. Based on the closed process of sensing, decision, execution and feedback, the CPS can achieve self-awareness, self-judgment and self-adjustment [4].

The major function of the environmental tier in a CPS is to collect and transmit the environmental information over the communication network, which connects the three tiers of the CPS, without human intervention. The fundamental feature of the environmental tier in a CPS is the communication without human operations, which is called as M2M communication or Machine-Type-Communication (MTC), where intelligent devices including sensors will communicate to each other end-to-end. The service-providing, decision-making and autonomous control components and technologies consist of the service tier and the control tier. The M2M communication in a CPS integrates wireless sensor networks (WSNs) with other communication systems such as a cellular network or an optical network. By utilizing both wireless and wired technologies, the M2M could monitor the

physical or environmental conditions and exchange the information among the components in different tiers.

A M2M communication system consists of three interlinked domains: 1) A M2M area domain including an M2M area network with M2M gateways, 2) A communication network domain including wired/wireless networks such as xDSL and 3G, and 3) An application services domain [5] consists of the end users and applications required in the CPS. The architecture of a M2M system is shown in Figure 2.

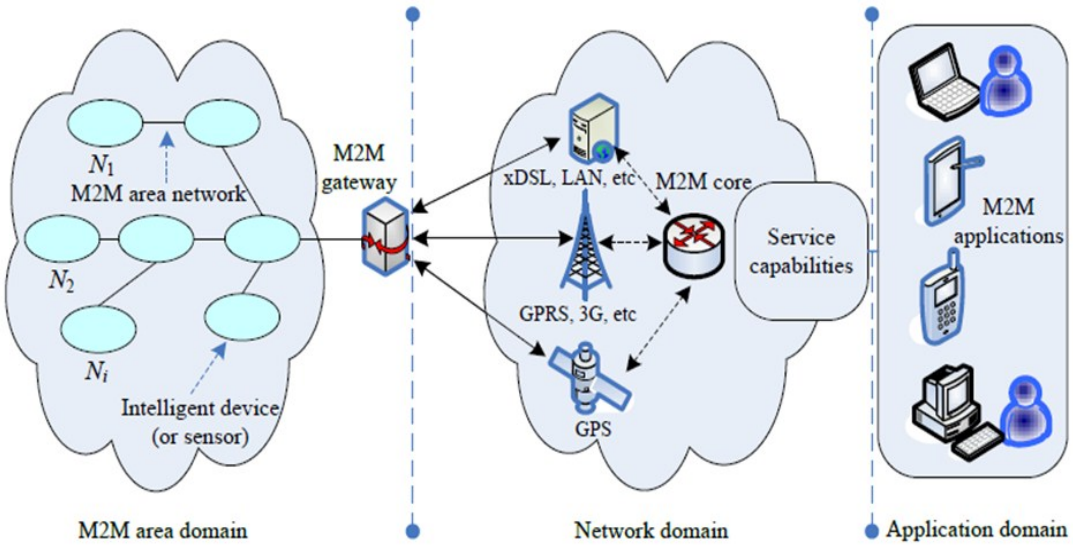

Figure2. The Architecture of a M2M System [5]

The collected information from the environment will be delivered from the M2M area domain to the network domain. The first destination of the information in the delivery is usually the M2M gateway which decides the communication protocols used and transforms the received information into the formats required by the corresponding communication systems. A middleware layer with routing and converting functions could exist in the network domain. The layer may perform network management roles such as auto configurations,

logging, notification etc. The communication systems in the network domain can be of any type of communication systems such as wireless local area networks (WLANs), telephone lines, Ethernets, satellite or cellular networks, which will exchange the information over a long distance. At the end, the information will be integrated into various applications in the application domain such as smart metering and smart grid, etc. [6].

Since CPS is a distributed, complex and hybrid real-time dynamic system with many different types of applications criticality operating at different time and space scales, it is easy to suffer various challenges to lose its functionality partially or entirely. Correspondingly, CPS may face lots of threats due to the challenges from system attackers. In order to ensure the security of CPS, there are many security requirements which should be fulfilled by the designers of CPS. Compared to the decision-making and autonomous control functionalities which are mainly working in the cyber world or the physical world, the M2M communication system, which bridges the physical world and cyber world, is more fragile in the CPS.

The M2M communication network in a CPS, has a few weakness, which makes the M2M unsecured [7]. First of all, in the M2M communication system, the major communication medium is the radio waveform, which is easily eavesdropped. Secondly, the sensor nodes, which are normally unattended, in the M2M communication have limited capabilities in terms of both energy and computing power. Thus, they are easily attacked and complex security schemes will not be feasibly used to protect them. Lastly, the network domain of a M2M system could integrate wireless and wired medium for the communication to the core network with different security schemes, which generates a protocol gap between different

communication protocols and could be a potential threat to the M2M communication system. The above mentioned features of the M2M communication have left the opportunity for the various malicious attacks to impair the system. And to explore the solutions to effectively protect the M2M communication will be a great challenge to the research field of M2M security.

The categories of possible attacks in M2M communication have been explored and specified by the Third Generation Partnership Project (3GPP) Security Workgroup (SA3) as follows [8].

- **Physical Attacks**

  Physical attacks including the insertion of valid authentication tokens into a manipulated device, inserting and/or booting with fraudulent or modified software (reflashing), and environmental/side-channel attacks, both before and after in-field deployment.

- **Compromise of Credentials**

  Compromise of credentials comprising brute force attacks on tokens and (weak) authentication algorithms, physical intrusion, or side-channel attacks, as well as malicious cloning of authentication tokens residing on the machine communication identity module (MCIM)

- **Configuration Attacks**

  Configuration attacks such as fraudulent software update/configuration changes, mis-configuration by the owner, subscriber, or user; and mis-configuration or compromise of the access control lists.

- **Protocol Attacks on the device**

  Protocol attacks directed against the device, which include man-in-the-middle attacks upon first network access, denial-of-service (DoS) attacks, compromising a device by exploiting weaknesses of active network services, and attacks on over-the-air management (OAM) and its traffic.

- **Attacks on the core network**

  Attacks on the core network, the main threats to the mobile network operator (MNO), include impersonation of devices; traffic tunneling between impersonated devices; mis-configuration of the firewall in the modem, router, or gateways; DoS attacks against the core network; also changing the device's authorized physical location in an unauthorized fashion or attacks on the network, using a rogue device.

- **User Data and Identity Privacy Attacks**

  User data and identity privacy attacks include eavesdropping user's or device's data sent over the access network; masquerading as another user/subscriber's device; revealing user's network ID or other confidential data to unauthorized parties.

From a traditional perspective, Rongxing Lu et al [9] have described the security requirements for M2M communication as follows:

- **Confidentiality**

  Confidentiality prevents unauthorized disclosure of sensory data in transmission from passive attackers, which ensures that only authorized entities can read these data in M2M communication systems.

- **Integrity**

  Integrity must be ensured so that illegal alteration of the sensory data (e.g., modifying, deleting, delaying, or replaying data) can be detected. In an M2M communications system, it is critical to meet the integrity requirement since illegal alteration may result in serious consequences, especially in life-critical M2M application contexts such as a remote e-healthcare-system.

- **Authentication**

  Authentication is a prerequisite for secure M2M communications, allowing the Base station (BS) in the application domain to corroborate the sensory data of the M2M nodes in the M2M domain.

- **Non-repudiation**

  Non-repudiation guarantees that M2M nodes, once sending data, cannot deny the transmission.

- **Access control**

  Access control is the ability to limit and control access to the BS in the application domain. Specifically, it allows only authorized M2M application systems to gain access to the BS.

- **Availability**

  Availability ensures that whenever M2M application systems access the BS, the BS is always available.

- **Privacy**

Privacy is also of paramount importance in some privacy-sensitive M2M communications systems (e.g., e-healthcare systems).

Since M2M communication is a developing technology and it possesses some unique weaknesses as mentioned above, the design of M2M communication should also fulfill the following requirements [10].

- Devices of M2M area domain normally have resource constraints and cannot afford the traditional heavy security technology. Therefore, designing a lightweight key management protocol is a key security issue.
- M2M integrates many different communication technologies in network domain. Defense capability of M2M communication varies in different subsystems. Therefore, during the security architecture design, the consistency and compatibility of different proposed security protocols must be considered so as to achieve the seamless connections among different networks.
- The data management in M2M system is automatic processing technology which lacks the detection functionality to detect malicious information. Therefore, an effective trust and repudiation management mechanism is requisite for data management in M2M. Separating the information content and source is another important requirement.
- The data precision demands of different applications in application domain might be significantly different. It will increase the hardness of privacy protecting since different precisions should be provided for different applications. To provide appropriate precision for different applications is a key requirement in the design of application

domain. Therefore, the M2M must make sure that the things and persons which possess the devices involved in M2M communication will never be marked or tracked by unauthorized things. In order to achieve this security requirement, deeply research on identification and privacy protection technology is very critical. Meanwhile, massive data real-time storage and inquiry problems in M2M system also raise a big requirement for distributed database technology.

1.2 Motivation

The CPS in the future would be such a powerful system which could lead another evolution of our life style. Among the CPS, almost every person is connected into the network and our daily life will be closely bound up with the CPS. If the security of CPS could not be guaranteed, there will be a lot of serious problems to cause the failure of its normal operations and our people's life and property would also be threatened. As a fundamental but fragile part of the CPS, there are a lot of attacks may occur in the M2M systems, which could endanger the operation of the M2M communication systems and thus the CPS. So it is very important to construct an effective security mechanism against various attacks to protect the M2M communication systems in CPS. Authentication is one of the basic security requirements in communication systems. Without a secure authentication, illegal users could masquerade as legal users to steal information or insert wrong information into the system. So in the thesis, we focus on the authentication of M2M devices in the M2M domain.

1.3 Contribution

In the thesis, we propose three authentication schemes for M2M communication. Our contributions made in this thesis can be summarized up as follows. (1) three authentication schemes for M2M communication: a dynamic-encryption authentication scheme for M2M security in cyber-physical system (CPS), an authentication scheme with identity-based cryptography (IBC) for M2M security in cyber-physical systems and an authentication scheme for multi-domain M2M security in cyber-physical systems have been proposed to mutually authenticate mobile devices, sensor nodes and the M2M service provider (MSP). (2) The first scheme proposes a dynamic encryption mechanism and a dynamic key generation mechanism to improve the security of M2M. A lightweight encryption algorithm has been employed to save the computation resource of sensor nodes. (3) The second scheme applies an integrated IBC scheme which is able to authenticate a message when encrypting it and without key escrow problem to a robust and efficient authentication scheme. (4) The third scheme applies hybrid encryption scheme which integrates certificateless encryption scheme and efficient advanced encryption standard (AES) algorithm to provide high level security. The communication scenario where sensor nodes of multi-domain exchange information with each other without human intervention has been considered and the corresponding authentication process is proposed. (5) All of the three schemes have considered the resource limit of sensor nodes and achieved a good balance between the security and system performance.

1.4 Organization

The reminder of this thesis is organized as follows. In Chapter 2, the important security solutions proposed in recent literatures is reviewed and the emerging technologies which would be applied to M2M communication in the future are explored. In Chapter 3, the two authentication schemes for single domain M2M security in CPS are presented in details. In Chapter 4, the authentication scheme for multi-domain M2M security in CPS is presented in details. Finally, the conclusion and future work are presented in Chapter 5.

Chapter 2. LITERATURE REVIEW

In this chapter, we first review the M2M security solutions related to mature technologies, i.e. solutions for M2M domain. Then we introduce the new technologies which would be applied in the network domain to make the M2M communication become a practical technology in the future and list the existed security solutions for the new technologies.

2.1 Solutions for M2M Domain

In the literature, there are many solutions found to address the security issues of the M2M communications. Most of the solutions focused on the mature technologies which are mainly related to M2M domain. The solutions could be classified into three kinds: detection, authentication and key management.

2.1.1 Detection

Rongxing Lu et al [9] designed a detection mechanism to prevent the M2M system from the node compromising attack in M2M domain. Since it usually takes the attacker some time to compromise the M2M nodes, forming the M2M nodes into couples to monitor each other could be a feasible way to detect compromised node. The M2M nodes in couple can utilize beacon messages to monitor each other periodically. The beacon message of the compromised node would be exceptional and detected by the couple nodes.

Zubair et al [11] modeled the malicious events in the smart grid (SG) system as a Gaussian process. A novel early warning system which utilizes a Bayesian data modeling technique called Gaussian process regression was designed to predict malicious events in the SG network based on this model. Samples of the function in different locations are observed

through Gaussian process regression. With a set of observation points and the corresponding real value, it is possible to compute the posterior distribution of a new point and thus possible to make predictions for unseen cases. The warning system enables the SG system to react in advance to defend the malicious activity.

Ioannis Broustis et al [12] designed a framework to detect and prevent M2M device hijacking. They evaluated the possibility of hijacking the devices which are deployed attended, and explained how the authentication process could be manipulated by an adversary. Then a lightweight framework was designed to defend the hijacking attacks based on traffic monitoring. The framework need no modification of clients and can improve the detection of hijacking attacks.

G. Cagalaban et al [13] proposed a mobile phone virtualization mechanism for M2M communication in cloud computing. They investigated a new approach which alters the detection ability to network service to achieve the increased detection coverage, less complex mobile software and reduced resource consumption. Their approach would become more important and valuable with the increasing of the mobile threats' scale and sophistication.

2.1.2 Authentication

A bandwidth efficient cooperative authentication (BECAN) for false reports filtering in M2M communication was designed by Rongxing Lu et al [9]. The objective of the scheme is to prevent the compromise attack which happens in the sleep mode of M2M nodes. BECAN applies the cooperative neighbor × router (CNR)-based filtering mechanism in which an M2M node and its neighboring nodes authenticate the sensory data cooperatively before the

M2M node sends the data to the M2M gateway. Therefore, the false data sent by the compromised node can be filtered as long as there is one uncompromised neighboring node taking a part in authenticating the sensory data.

Tien-Dung Nguyen et al [14] proposed a dynamic ID-Based authentication scheme which is applied in the M2M environment for hospital. Considering the resource limits of sensor nodes, a computationally efficient encryption algorithm is applied to defend the network attacks and the symmetric key between mobile devices and sensor nodes is established by a pair wise key pre-distribution scheme. Each time a sensor node wishes to transmit sensory data to mobile device, the beacon signal from the source is authenticated first. Through their scheme, the probability for uncompromised sensors to construct a secure session with M2M device is higher.

Sachin et al [15] designed an authentication and verification scheme for M2M system. In the solution, the standard Generic Bootstrapping Architecture (GBA) in the 3GPP specifications is extended so as to minimize the additional asset requirements. There are two procedures in the solution: bootstrapping authentication and bootstrapping usage. The coordinator node authenticates itself to operator in bootstrapping authentication and derives the key material which is used to secure the subsequent session between the M2M Server and itself. Both of the subscriber identity module (SIM) card and coordinator state are verified in the scheme to prevent the system from card stealing attacks which are easily to launch since the M2M devices are usually deployed widely.

Yingying He et al [16] propose a new improved Direct Anonymous Attestation (I-DAA)

protocol to realize the remote authentication of M2M device with anonymousness and trustworthy while considering the resource limits of most embedded devices at the same time. The proposed I-DAA scheme achieve two goals: the security of traditional DAA keeps the same and the computational complexity is greatly reduced which means a lot to the resource-limited M2M networks.

A. Bartoli et al [17] propose an authentication/verification method at the physical (PHY) layer for M2M networks. The presented scheme is able to distinguish whether a packet should be kept through the received authentication preamble (AP) at the PHY layer. With this ability, the scheme could save energy and increase the system lifetime so as to guarantee the system's long-term availability. Recognizing the importance of the thus required synchronization window and the possibility of desynchronization because of poor channel conditions, A. Bartoli et al introduce a novel synchronization process in [18]. For said process, they calculate the optimum synchronization window length taking error rates into account. The analysis indicates that the scheme allows minimizing energy expenditure and also quantifies the impact onto memory and central processing unit (CPU). In [19], A. Bartoli et al extend the previous work with a novel synchronization protocol that addresses previous desynchronisation issues. Besides, more appropriate deployment parameters which could maximize the overall energy savings are analyzed and presented. They also describe the details about the key management mechanism: key generation and key updating processes. Moreover, for practical usage, they show how to fit the proposed mechanism into the IEEE 802.15.4e amendment to the IEEE 802.15.4-2006 standard.

Hyundong Lee and Mokdong Chung [20] propose a context-aware authentication system for M2M services in the smart phone environment. The context-awareness, integrated authentication, access control, and an open service gateway initiative (OSGi) service platform are used in the system. In addition, they recommend Fuzzy Logic and Multi-Attribute Utility Theory (MAUT) as the solution for handling diverse contexts properly as well as in determining the appropriate security level. The proposed context-aware security system can provide a flexible, secure and seamless security service by adopting diverse contexts in the smart phone environment.

Liang Hu et al [21] construct an integrity and secure authentication system which consists of four parts: two-way authentication, re-authentication, roaming authentication and inside authentication. Two-way authentication is to make the mobile device and the center system trust each other, and two-way authentication is the foundation of the other three. Re-authentication is to re-establish the active communication after the mobile subscriber changes his point of attachment to the network. Inside authentication is to prevent the problems in which the mobile device is captured by the attacker and the secret stored in the mobile device is stolen. Roaming authentication is to authenticate the mobile subscriber's legitimate identity to the new agency which locates in the place where it roams into, and roaming authentication can be regarded as the integration of the above three. The simulation of the proposed authentication mechanism and analysis of the existed schemes indicate that the authentication mechanism and the encryption mechanism establish a secure integrated framework for Mobile Payment.

Chengzhe Lai et al [22] raise several new security issues in M2M communication including group access authentication, multiparty authentication and data authentication, and propose their solutions through modifying existing authentication protocols and cryptographic algorithms, the first is group authentication and key agreement protocol used to solve group access authentication of M2M, the second is proxy signature for M2M system to tackle authentication issue among multiple entities and the third is aggregate signature used to resolve security of small data transmission for M2M.

Xuebin Sun et al [23] propose a M2M application model that connects a mobile user with the home network under the Time Division-Synchronous Code Division Multiple Access (TD-SCDMA) network environment. Based on the proposed model, a password-based mutual authentication and key establishment protocol is designed. In the protocol, the mobile users, M2M server, and smart home devices are mutual authenticated; the legal home gateway is verified by the M2M server when its first log in; and an encryption key is generated to encrypt the messages transmitted among parties. Each time a mobile user registers with the M2M server, the M2M server gives permission to the user and the encryption key to the corresponding home gateway. Then after a mutual authentication process, the user can access its home network through the secure channel established between the M2M server and home gateway.

Ye Yan et al [24] present a zero correlation zone (ZCZ) code division multiple access (CDMA) based scheme in M2M communications for the advanced metering infrastructure (AMI) in smart grid. They design a mutual authentication process for the initialization phase of

the system. Through the authentication scheme, every entity establishes a ZCZ code at both the communication counterparts for data coding/decoding when joins the smart grid AMI M2M communications. Subsequently, a ZCZ CDMA based scheme which possesses the security and efficiency features is provided as the data collector/ dispersion process.

Inshil Doh et al [25] propose a secure user authentication and key distribution mechanism based on Kerberos. The Kerberos is an authentication protocol which utilizes a trusted third party in an open network environment. The establishment of pairwise keys which secure the content transfer between provider and receiver applies the Kerberos technology. Moreover, they propose group key sharing and ticket redistribution schemes for secure three screen services in home network. The experimental results showed that the proposed method provides more efficiency and flexibility than the traditional mechanisms in an open Internet protocol television (IPTV) environment.

Wujun Zhang et al [26] present an end-to-end security scheme for MTC communication. The scheme first presents the Generic Authentication Architecture and provides an authentication scheme which is suitable to the MTC feature for application layer. The scheme possesses good scalability since it makes use of the existing network infrastructure and needs no extra devices. In the scheme, they come up with the function which adapts the lifetime of the master session key according to the expected number of bootstrapping request.

Nenad Gligoric et al [27] propose a hybrid application-layer security scheme with compression for M2M communication over short message service (SMS). In the scheme, the device international mobile equipment identity number (IMEI), secret key and payload are

used to calculate the signature which is used in message authentication so as to achieve the dynamic signature and thus prevents duplicated/stolen signature and replay attack. The implementation of the security mechanism shows that the SMS service delivery time is the same in the proposed scheme and regular M2M communication. Securing SMS transport for M2M communication could enable the mobile and sensor networks interacting with the SMS Gateway interface and thus fill the gap between cellular system and resource limited sensor system.

Ioannis Broustis et al [28] propose an efficient group authentication framework. In the framework, the service provider is able to categorize the client devices in groups and authenticate them as members of a particular group. With the group authentication, the signal overhead and process of authentication are greatly reduced and simplified compared to the traditional authentication scheme. The application of their framework in M2M system enables the M2M server to authenticate large amounts of M2M devices in the same group at a time. So the complexity and bandwidth requirements for authentication are reduced to a large extent. Furthermore, a unique key is generated to each device to prevent them from the attacks from the other devices.

Jin Cao et al [29] propose a group-based access authentication scheme by the technique of aggregation signature. Their scheme can not only achieve mutual authentication and key agreement between each MTC device (MTCD) in a group and the mobile management entity (MME) at the same time, but also greatly reduce the signaling traffic and thus avoid network congestions. By the scheme, a mass of MTCDs is initialized to form an MTC group and

choose a group leader. When multiple MTCDs in the MTC group request to access to the network simultaneously, the MME authenticates the MTC group by verifying the aggregate signature generated by the group leader on behalf of all the group members and establishes a distinct session key for each MTCD with different key agreement parameters sent from the MTCDs.

Wei Ren et al [30] construct a formally model of authentication in M2M. After the stating, presenting and analyzing of security for authentication, they model four attacking adversaries which could launch channel eavesdropping attack, credential compromise attack, function compromise attack and ghost compromise attack to the M2M system. Next, for each adversary, a model is proposed to handle the corresponding attacks. Finally, they present the security proof for the authentication models.

Jin-Mook Kim et al [31] propose an efficient privacy problem solving using device and user authentication (PSDUA) design. The system can prevent the privacy of the parties from leaking to the communication which they are not involved in. Besides, they also achieve the confidentiality, integrity and ability against man-in-the-middle attack. The PSDUA design separates the certification service into two parts: the user authentication and service authentication. The user authentication is achieved by utilizing user login information and service authentication is solved through timestamp.

2.1.3 Key Management

Yosra et al [32] propose a novel approach for establishing session keys for highly resource-constrained sensor nodes encountered in these M2M environments with an external

server. The proposed system exploits the heterogeneity of M2M systems by delegating cryptographic computational task to the nodes with more resource in a collaborative scheme. They present a novel key establishment protocol in which a highly resource-constrained node obtains assistance from more powerful M2M nodes in order to make use of asymmetric cryptography primitives to establish a shared secret key with a remote server. Furthermore, the highly resource-constrained node can do so through simple exchanges with neighbor nodes, which are considerably less energy consuming than actual use of these cryptographic primitives.

Relying on a similar collaboration scheme of [32], Yosra et al [33] present a new approach which enforce the session key derivation process. The scheme is a two-pass key transport protocol in which the secret values are exchanged between the resource limited node and the powerful server. To transmit its secret value, the resource limited node splits it into multiple parts and requires its neighbors to encrypt the split parts and deliver them to the server. Once receiving the secret value, the server requires the neighbors to transmit its own secret value to the resource limited node. So in the scheme, the resource limited node need no to deal heavy asymmetric computation tasks and save their resource a lot.

B S Adiga et al [34] discuss the suitability of Identity Based (IDB) Cryptosystems to solve privacy and security issues in Machine to Machine (M2M) communications for Internet of Things (IoT) applications. Using the IDB cryptosystem allows any users to communicate securely and verify each other's signatures without distributing certificates in advance. They also discuss various privacy and security issues related to M2M communication and shows

how these issues can be resolved using IDB. However, in the cryptosystem, it assumes the existence of a trusted key generation center (PKG) whose sole purpose is to give each user a smart card when he first joins the network. The center can close down after all the cards are issued. The information embedded in the card enables the user to sign the message he sends and to verify the messages he receives. That means the smart card of users keep the same after obtained from the PKG which implies the potential security threat that the leakage of smart card would enable the attackers to stole or change the transmitted message. Moreover, they didn't give the procedure of smart card issuing.

2.2 Solutions for Network Domain

Although there are already many solutions exist for M2M security, there are two features in M2M communication may bring the problems which could not be solved by the current technologies. The two features are: 1) The devices are deployed in very large amount and the quantity will be larger and larger in the future, 2) Most of the devices are resource-constrained. To overcome the challenges brought by the two features, there are two rising technologies would be applied to M2M communication--the first one is Internet Protocol version 6 (IPv6) over Low power Wireless Personal Area Networks (6LoWPAN) and the second one is Constrained Application Protocol (CoAP). Nowadays, there are already some security solutions for 6LoWPAN which belongs to network domain.

2.2.1 6LoWPAN [35]

Due to the amount of devices, M2M will need a very large address space which could only be provided by IPv6. So applying IPv6 to M2M communication is definitely the future

trend. The Internet Engineering Task Force (IETF) has been developing a new standard named 6LoWPAN to enable the use of IPv6 in Low-power and Lossy Networks (LLNs), such as those based on the IEEE 802.15.4 standard.

IEEE802.15.4 only stipulated the standard of PHY and MAC layer while no touching the criterion of network layer. To make the interconnection and cooperation of different machines come true, the standard of network layer is needed. 6LoWPAN enable the IPv6 to be used on the 802.15.4 compliant devices which are resource limited by compressing the IPv6 packets. The PHY and MAC layer standards of IEEE802.15.4 are adopted as the bottom layer and IPv6 functions as the networking technology in 6LoWPAN technology. To solve the inconformity between the payload length supported by IPv6 and the one provided by 6LoWPAN bottom layer, 6LoWPAN working group introduces an adaptation layer between MAC layer and network layer to achieve the seamless connection. The reference model of 6LoWPAN Protocol Stack is shown in the following Figure 3.

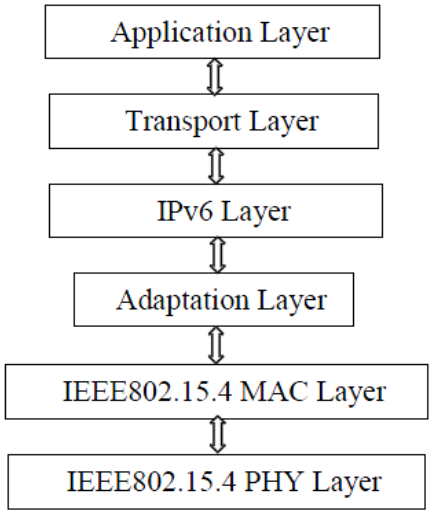

Figure3. Model of 6LoWPAN Protocol Stack [35]

2.2.2 CoAP [36]

Since the M2M devices are battery supplied and stay in the sleeping mode unless there is data traffic, the M2M applications require a multicast and asynchronous communication approach compared to the unicast and synchronous approach of standard Internet applications.

In March 2010, the IETF Constrained RESTful Environments (CoRE) Working Group started the standardization activity on CoAP. CoAP is an application layer protocol proposed to enable the very simple electronics devices to communicate over the Internet. It is particularly designed for resource limited components that need to be attended through Internet remotely. CoAP is based on a Representational State Transfer (REST) architecture in which Universal Resource Identifiers (URIs) identify the resources. Then the resources can be accessed by the same means as those used by HyperText Transfer Protocol (HTTP). CoAP is composed of a subset of HTTP functionalities which is reconstructed with the consideration of the resource constraints feature. Besides, it modifies some existing mechanisms and proposes some new functions to make itself suitable for M2M applications. The HTTP and CoAP protocol stacks are shown in Figure 4.

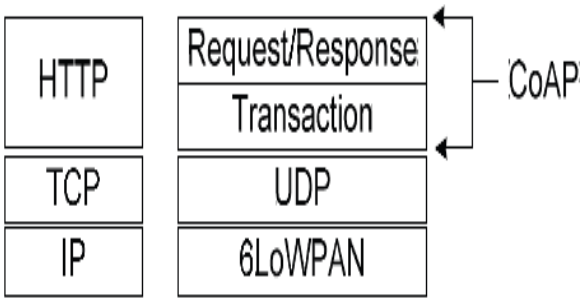

Figure4. The HTTP and CoAP protocol stacks [36]

2.2.3 Security Solutions of 6LoWPAN

Shahid Raza et al [37] provide an End-to-End (E2E) secure communication scheme between IP-based sensor networks and the traditional Internet. In the scheme, the IPsec technology which provides authentication and privacy for IPv6 is firstly transformed into the 6LoWPAN extension. The benefit of using IPsec is that there is no need to modify the existing ports on the Internet in order to communicate with the WSN. What's more, utilizing IPsec eliminates the need for a trustworthy gateway. One drawback of using IPsec is that supporting IPsec's Authentication Header (AH) and Encapsulation Security Payload (ESP) will increase the packet sizes due to the need to include additional headers. However, utilizing IPsec eliminates the need of using existing 802.15.4 link-layer security mechanisms which could in turn free some header space. Through applying IPsec, the authentication, confidentiality and integrity of messages could be achieved by using standardized and established IPv6 mechanisms.

Yuanyuan Zhou et al [38] propose a 6LoWPAN-based security gateway which connects wireless sensor network with IPv6 network. The secure network encryption protocol (SNEP) is adopted in the design. The benefit of SNEP is the low communication overhead feature. With the SNEP protocol, the authentication, freshness, integrity and confidentiality are realized. In the system, their concern is only the communication between an IPv6 network user and a WSN sensor node without broadcast queries. A web server which stores the periodically collected data is built in the gateway. The query of stored historical data allows

event detection in WSN. Moreover, a user access authority table which is in charge of user authentication and access control is maintained in the web server.

Chapter 3. AUTHENTICATION SCHEMES FOR SINGLE DOMAIN M2M SECURITY

Motivated to improve the security functionality of the M2M communication with much more robust authentication schemes, in this chapter, a dynamic-encryption authentication scheme for M2M security in cyber-physical system (CPS) and an authentication scheme with identity-based cryptography (IBC) for M2M security in cyber-physical systems have been designed and formally verified.

3.1 A Dynamic-Encryption Authentication Scheme for M2M Security

3.1.1 M2M System Model

The M2M system model used in our scheme is shown as Figure5. There are four parties—mobile devices, sensor nodes, gateways and the MSP in the M2M system. The mobile device is carried by a user to gather information from sensor nodes. Sensor nodes could relay messages with Bluetooth technology which is the communication medium between each other. The gateways are in charge of sending messages from sensor nodes to the MSP. The MSP works as an authentication center to verify the IDs of mobile devices and generates new IDs. It is assumed that the communication link between gateways and the MSP is secure since the link could consist of GSM, 4G, satellite, etc which have already been equipped with strong security schemes, while the secure communication over the links among the MSP, sensor nodes and mobile devices is the concern in this scheme.

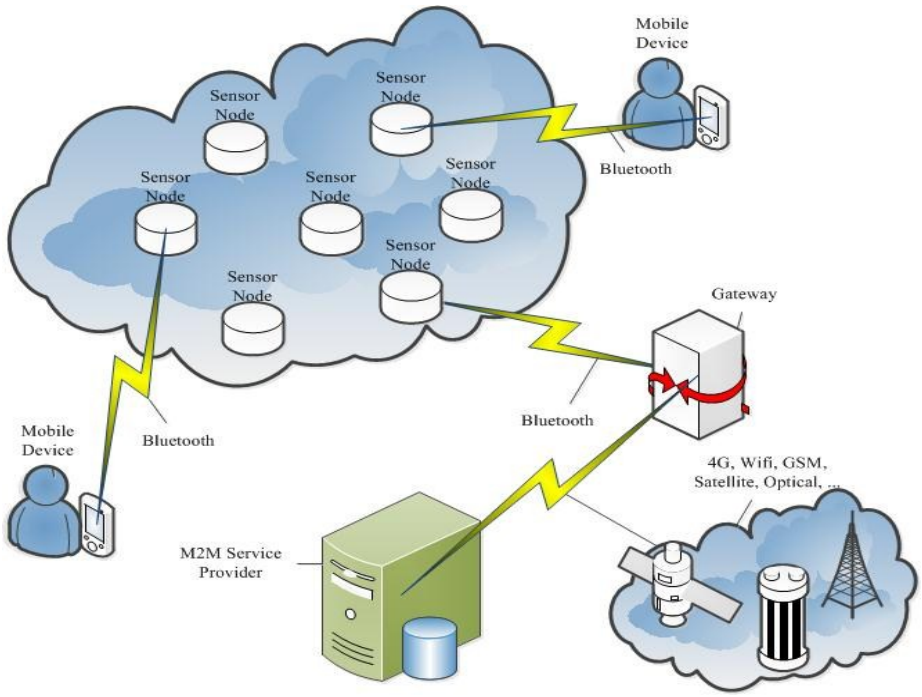

Figure5. M2M System Model

3.1.2 Parameters and Functions

The same collections of complex encryption algorithms *EP[l]* are stored at the mobile devices and the MSP where *l* is the number of algorithms. The mobile devices and MSP could randomly select an encryption algorithm EP_u from the collection by an index *u*.

The same initial key space, which is a matrix $G_{n\times n}$, a function *F(x, y)*, a matrix $T_{m\times r}$, of which T_{ji} indicates the number of *j*th message corresponding to mobile device *i* has been exchanged, and a lightweight encryption algorithm *EK* are stored at the mobile devices, sensor nodes and the MSP. With a shared seed, the initial key space and the function *F(x, y)*, mobile devices, sensor nodes and the MSP could generate an encryption key as a one-time-password.

Each mobile device and each of sensor nodes will be assigned with an ID when the system is deployed, while the IDs of mobile devices could be updated in the authentication.

3.1.3 Proposed Scheme

As the major novel contribution made in this study, the dynamic-encryption mechanism is first introduced which is the soul of the proposed authentication scheme. The mechanism has two functions: the dynamic encryption algorithm used between the mobile devices and the MSP and the dynamic encryption key generation scheme among the MSP, mobile devices and sensor nodes.

Each time a mobile device wants to collect information from a certain sensor node, it needs to be authenticated by the MSP to set up a session. It will encrypt a request message with a randomly generated key and one encryption algorithm selected from the collection of algorithms by an index. After the encryption, the mobile device will send the encrypted request message, algorithm index and the random key to the MSP. At the MSP side, it will select the same algorithm according to the index received to decrypt the received message with the received random key and vice versa. The mobile device and the MSP could select different encryption algorithm each time. So the encryption becomes dynamic, which is used only for the mobile devices and the MSP because the mobile devices and the MSP are powerful enough to equip complex encryption algorithms.

With the $G_{n\times n}$, a seed S_{abc}—$a|b|c$ and a function $F(x, y)$, an encryption key $K_{abc} = F(a, G_{bc})$ could be generated, where a, b, c are random numbers, $0 < b, c \leq n$ and G_{bc} is the element in $G_{n\times n}$ of bth row and cth column. The encryption key will be used by the lightweight encryption algorithm EK at each component to encrypt the messages transmitted over the M2M network. In each session, the S_{abc} would be dynamic since a, b, c

are random numbers. Then K_{abc} corresponding to S_{abc} is a one-time-password which enhances the security. On the other hand, the higher frequency we update the initial key space, the higher security level will be. It implies that the security strength can be controlled by the system administrators.

In the following, we describe our proposed authentication scheme consisting of two phases: Key pre-distribution phase and Authentication phase. The notations used in our scheme are defined in Table I.

TABLE I. DEFINITION OF NOTATION

ID_i	Identity of mobile device i |i
IP_h	Identity of sensor node$h$
S_{abc}	Seed$a$|b|c used to calculate key
K_u	The key corresponding to complex encryption algorithm u
$F(x,y)$	A fuction F:{x,y}→z
K_{abc}	The encryption key corresponding to S_{abc}
T_{ji}	The number of Msgj related to device i has been exchanged
$EK_{abc}(d)$	Encryption of message d with lightweight algorithm and K_{abc}
$EP_u(d)$	Encryption of message $d$withalgorithm$u$ and K_u

- **Key Pre-distribution Phase**

The elements of $T_{m\times r}$ are initialized to zero, where m is the number of messages used in the authentication process and r is the number of mobile devices.An initial key space $G_{n\times n}$, where n is a security parameter of our scheme, will be assigned. The larger the initial key

space, the higher level of security will be and more storage cost will be required. And a function *F(x, y),* designed by the system administrator will be updated.

- **Authentication Phase**

In this phase, the mobile device *i*, the target sensor node *h* and the MSP will be mutually authenticated by the proposed dynamic-encryption authentication scheme. The process is shown as Figure6.

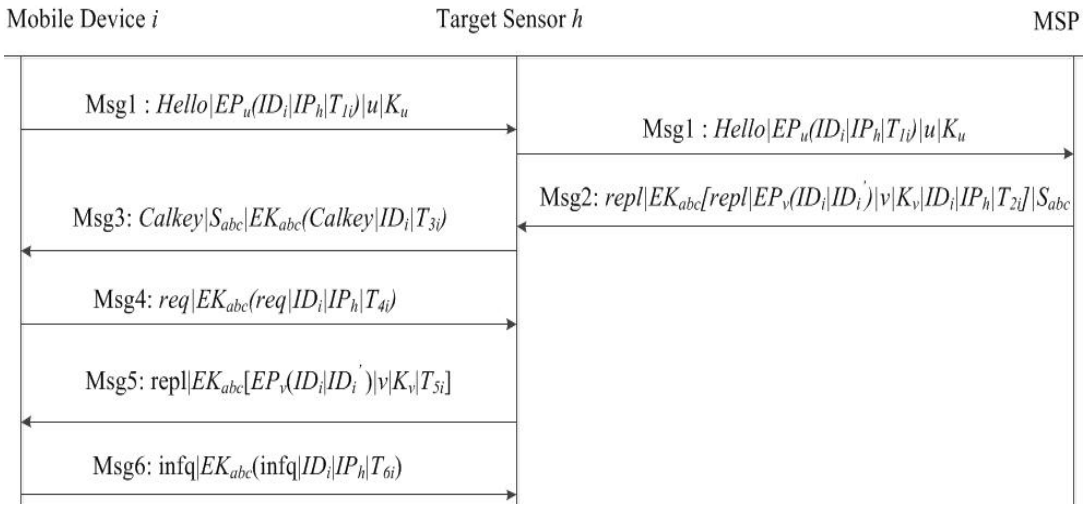

Figure6. The Process of Authentication Phase

The details of authentication process are described as follows.

- Msg1 : $Hello|EP_u(ID_i|IP_h|T_{1i})|u|K_u$

Msg1 is the message to initialize a session for the data collection purpose. When the mobile device *i* wants to build a new connection with sensor node *h* to collect information, it will select an encryption algorithm *u* from its collection of the encryption algorithms and generates a random number as the session key K_u to encrypt its ID, the ID of the target sensor node and T_{1i}. Then, it will send the encrypted message, K_u and index *u* as the Msg1

to any of its neighbor sensor nodes. And the sensor nodes will forward the Msg1 to the MSP. $T_{1i}=T_{1i}+1$.

- Msg2: $repl|EK_{abc}[repl|EP_v(ID_i|ID_i^{'})|v|K_v|ID_i|IP_h|T_{2i}]|S_{abc}$

When the MSP receives the Msg1, it will choose the corresponding encryption algorithm according to the received index u and decrypt Msg1 with K_u to get T_{1i} and ID_i for verification. The MSP will firstly verify ID_i. After verifying ID_i, the MSP continues to compare T_{1i} with the value of T_{1i} stored in local. If the value of T_{1i} is wrong, the MSP will ignore the Msg1. Otherwise, it generates a new ID--$ID_i^{'}$ and a seed--S_{abc}. It first encrypts ID_i and $ID_i^{'}$ by an encryption algorithm v selected from the collection of encryption algorithms with a new session key K_v and then combines the encrypted message with v, K_v, ID_i, IP_h and T_{2i}. Then it calculates K_{abc} with S_{abc} and $G_{n\times n}$ by the function of $K_{abc} = F(a,G_{bc})$ and encrypts the combined message by the lightweight encryption algorithm with K_{abc}. Finally, the MSP combinesthe double-encrypted message with S_{abc} as Msg2 and sends it to sensor nodes. $T_{1i}=T_{1i}+1$.$T_{2i}=T_{2i}+1$.

- Msg3: $Calkey|S_{abc}|EK_{abc}(Calkey|ID_i|T_{3i})$

After a sensor node receives the Msg2, it calculates K_{abc} with S_{abc} and $G_{n\times n}$ and decrypts the Msg2 by the lightweight encryption algorithm with the K_{abc}. Then, the sensor node gets IP_h, ID_i, T_{2i}, v, K_v and the message encrypted by the encryption algorithm v. If T_{2i} is not equal to the value of T_{2i} stored in local, the Msg2 will be ignored. Otherwise, the sensor node will continue to compare IP_h with its ID. If IP_h is not its ID, it will forward the received Msg2 to other sensor nodes.

Once the target sensor node decrypts the Msg2 and the value of T_{2i} is correct, it will find the IP_h is the same as its ID. It also gets ID_i, v, K_v and the message encrypted by the algorithm v. The target sensor node will encrypt ID_i and T_{3i} by the lightweight encryption algorithm with K_{abc} and send the encrypted message and S_{abc} as Msg3 to the mobile device i. $T_{2i}= T_{2i}+1$. $T_{3i}= T_{3i}+1$.

- Msg4: $req|EK_{abc}(req|ID_i|IP_h|T_{4i})$

After receiving the Msg3, mobile device i calculates K_{abc} with S_{abc} and $G_{n\times n}$ and decrypts Msg3 by the lightweight encryption algorithm with K_{abc}. It will firstly compare the received device ID with ID_i. If the received ID is not ID_i, the mobile device i will ignore the Msg3. Otherwise, the mobile device i will compare the value of T_{3i}. If $T_{3i}$is correct, the mobile device will encrypt ID_i, IP_h and T_{4i} by the lightweight encryption algorithm with K_{abc} as Msg4 and send it to target sensor node h to indicate the connection has been correctly built. $T_{3i}=T_{3i}+1$. $T_{4i}=T_{4i}+1$.

- Msg5: $repl|EK_{abc}[EP_v(ID_i|ID_i^{'})|v|K_v|T_{5i}]$

After decrypting the Msg4 and comparing T_{4i} and IP_h, the target sensor node h combines the message encrypted by the encryption algorithm v in Msg2 with index v, K_v and T_{5i} and encrypts them by the lightweight encryption algorithm with K_{abc} to generate Msg5 to send to the mobile device$i$.$T_{4i}=T_{4i}+1$.$T_{5i}=T_{5i}+1$.

- Msg6: $infq|EK_{abc}(infq|ID_i|IP_h|T_{6i})$

After receiving the Msg5, mobile device i will decrypt it twice by the lightweight encryption algorithm with K_{abc} and by the encryption algorithm v with K_v, correspondingly.

Besides comparing ID_i and T_{5i}, it also obtains its new $ID_i^{'}$ which generated by the MSP. It will further encrypt ID_i, IP_h and $T_{6i}$by the lightweight encryption algorithm with K_{abc} as Msg6 and send the Msg6 to the target sensor node h to request information.T_{5i}=T_{5i}+1.T_{6i}=T_{6i}+1.

By now, the mobile device i has built a secure session with the sensor node h with K_{abc}, IP_h and ID_i. The information collected by sensor node h could be delivered to the mobile device i with high security. After the session is over, the ID of mobile device i will be replaced by $ID_i^{'}$.

3.1.4 Security Analysis

In this section, we will analyze the security function of our authentication scheme.

- Mutual Authentication

A mutual authentication among mobile devices, sensor nodes and the MSP can be accomplished with the dynamic-encryption mechanism. Firstly, the ID of mobile device is verified by the MSP with the dynamic encryption algorithm. Only the legitimate mobile devices and the MSP possess the collection of encryption algorithms. The target sensor node judges whether the mobile device is legitimate by checking if the mobile device could encrypt and decrypt the exchanged message with a correct key. Only the legitimate mobile devices and sensor nodes could generate the same key with a seed and possess the same lightweight encryption algorithm. Moreover, the mobile devices and sensor nodes will check the value of $T_{m\times r}$ once they receive a message, which could prevent the message flooding in the system since the repeated messages will be ignored due to the unchanged T_{ji}.

- Ability against Man-in-the-middle Attacks

Although an attacker could obtain the transmitted message to get the seed or the index of the encryption algorithm used, it has no access to the initial key space $G_{n\times n}$, lightweight encryption algorithm and the collection of encryption algorithms. So the attacker could not insert any illegal information into the connection through modifying the encrypted data.

- Ability against Reply Attacks

An attacker could get a message and reply it to the destination to pretend that the legitimate source sends the message again. However, by our scheme, T_{ji} has been also encrypted in Msgj or stored in $T_{m\times r}$ at destination. Each time a legitimate party sends or receives Msgj, it will change the corresponding value of T_{ji}. Since the attacker is not able to decrypt the Msgj, the value of T_{ji} cannot be changed. So even an attacker ceaselessly replies the messages, the replied messages will all be ignored as the value of T_{ji} stays the same. The attacks will not be successful.

- Ability against DoS Attacks

An attacker may reply Msg1 to the MSP continually. After receiving many Msg1s containing the same ID_i, the MSP will reject to serve for the ID_i since it considers the device which possesses the ID_i has been compromised. If an attacker just replies Msg1 to the MSP to launch a DoS attack, the MSP will decrypt Msg1 with the encryption algorithm u. Then the MSP will find the value of T_{1i} is wrong, it will ignore the Msg1. If an attacker changes the index u to u' in Msg1, the MSP will get an invalid ID after it decrypts the Msg1 with

algorithm u', if the algorithm u' exists. The MSP will still ignore the Msg1. So the DoS attacks could not success.

- Ability against Impersonation Attacks

An attacker may store valid messages in this session and reply them in the next session to launch an impersonation attack. By our scheme, the ID will be replaced by ID$'$ when the session is over. So if the attacker wants to use the messages of previous session to issue an impersonation attack, the previous ID contained in the messages will be no longer valid. These messages will be ignored to make the attacks failure.

3.1.5 Formal Verification

Simple Promela Interpreter (SPIN) is a very popular and one of the most powerful tools for catching software defects in concurrent system designs. The tool functions by thoroughly checking system models which reflect their essential elements. If a defect is found, SPIN can produce a sample execution of the model to indicate it. SPIN has a wide range of applications from the verification of complex call processing software used in telephone exchanges, to the validation of control software for spacecraft.

The M2M communication system under the study is a distributed system consisting of three types of participants: mobile devices, sensor nodes and the MSP. To prove the logical correctness of our authentication scheme in the M2M system, we use SPIN to simulate and verify the model of the proposed scheme. The results are shown as Figure7 and Figure8.

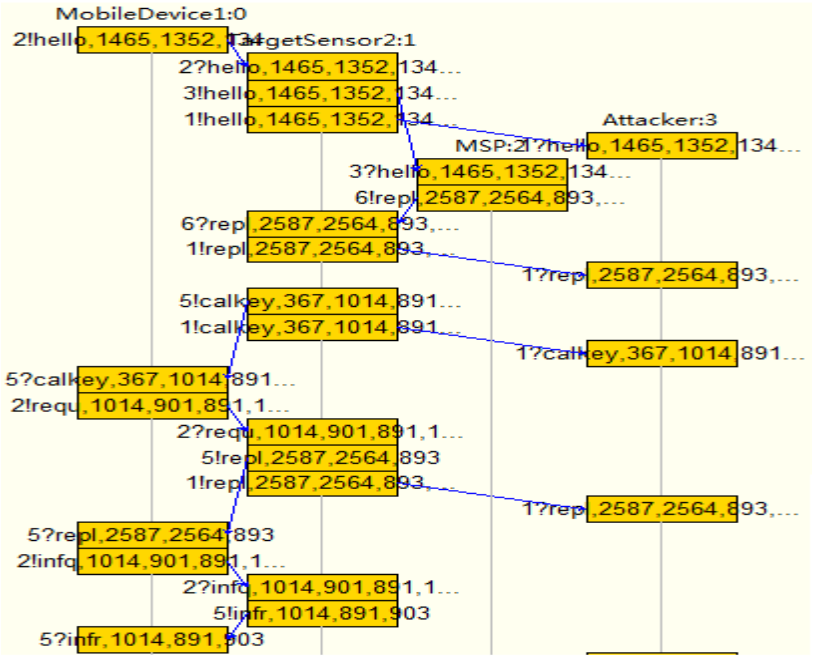

Figure7. The Process of Authentication

The Figure7 indicates the process of authentication among mobile device *1*, target sensor *2* and the MSP. In real world, there are many sensor nodes to relay messages besides target sensor *2*, the MSP and mobile device *1*. Since the relay sensors just transmit messages without modifying them, we regard the whole relay sensors as a simple link. In Figure7, we can see Msgj transmitted among the three parties and caught by an attacker which is in listening mode. However, we could not see all the testing information due to the limitation of space. In Msg1, the encrypted ID_1 is 1465, IP_2 is 1352 and T_{11} is 134. After ID_1 is received by the MSP, it will be decrypted and verified. In Msg2, the encrypted ID_1 is 2587, ID_1' is 2564 and index *v* is 893. The ID_1' is stored in target sensor *2* temporarily instead of sending to mobile device *1* directly. In Msg3, the seed S_{abc} is 367, the encrypted ID_1 is 1014 and T_{31} is 891. Using seed 367, the mobile device calculates the key. In Msg4, the ID_1 is 1014, IP_2 is 901 and T_{41} is 891. In Msg5, the ID_1 is 2587, ID_1' is 2564 and index *v* is

893.The mobile device could obtain $ID_1^{'}$ by decrypting 2564 with K_{367}, lightweight algorithm, K_v and the algorithm v. At the end of Figure7, it is clear that the authentication is successful and mobile device*1*is able to request and receive information from the target sensor. In Msg6, the encrypted ID_1 is 1014, IP_2 is 901 and T_{61} is 891. In the subsequent reply message, the encrypted ID_1 is 1014, T_{data} is 891 and the encrypted data is 903. T_{data} could be recorded by reusing an element of $T_{m\times r}$ after authentication is successful.

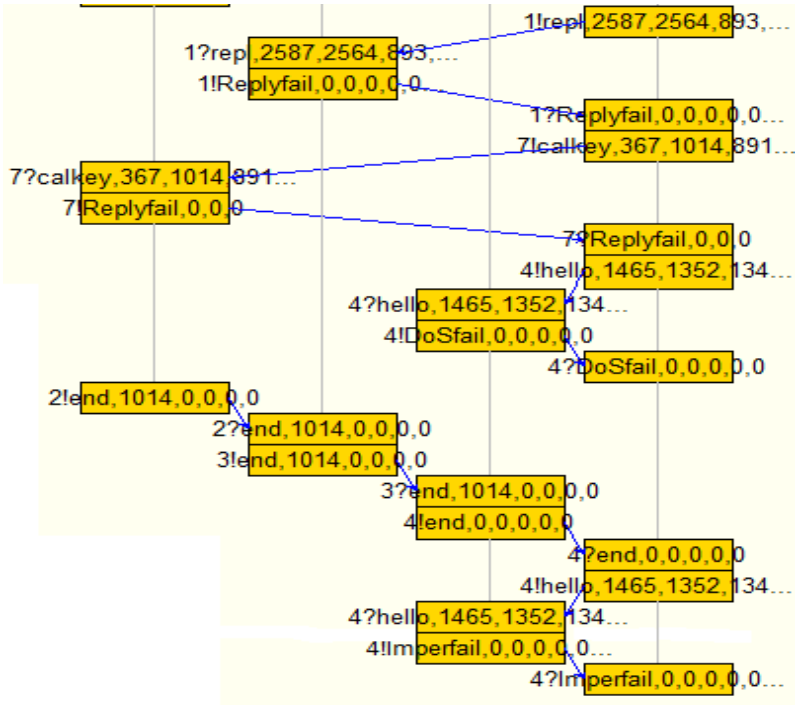

Figure8. Demonstration of Withstanding of Attacks

In Figure8, the attacker changes to be active. It launches multiple attacks by sending the previously stored messages to the related party. When it sends Msg2 to target sensor *2* and Msg3 to mobile device *1*, the reply attacks cannot succeed because the value of T_{21} and T_{31}have not been updated. When it sends Msg1 to the MSP in a session, the DoS attacks cannot succeed because the value of T_{11} stays the same. When it sends Msg1 to the MSP in the next session, the impersonation attacks cannot succeed because the ID is already invalid.

The Figure8 demonstrates the security functions of the proposed authentication scheme against various attacks.

3.1.6 Efficiency Analysis

We evaluate the computation cost and storage cost of our scheme. The notations are shown in Table II.

The computation cost of random key and seed generation is assumed to be small enough to ignore. The computation cost is $2C_p+4C_k+C_f$ for mobile devices, $4C_k+C_f$ for sensor nodes and $2C_p+C_k+C_f$ for the MSP. It's clear that both C_k and C_f are much

TABLE II. DEFINITION OF NOTATION

C_p	Computation cost of complex encryption algorithm
C_k	Computation cost of lightweight encryption algorithm
C_f	Computation cost of function $F(x, y)$
S_I	Storage cost of ID_i or IP_h
S_p	Storage cost of complex encryption algorithm
S_k	Storage cost of lightweight encryption algorithm
S_g	Storage cost of an element in key space
S_f	Storage cost of function $F(x, y)$
S_t	Storage cost of T_{ji}

less than C_p. The total computation cost of a sensor node is much smaller than that of a mobile device or the MSP. Assume that the size of initial key space is $n \times n$, the number of

mobile devices is r, the number of sensor nodes is o and the number of complex encryption algorithms is l. In the authentication process, a sensor node handles 5 messages, while the MSP processes 2 messages. And mobile device i deals with 5 messages and it only records the T_{ji} since it ignores the Msgj related to other devices after comparing ID. The storage cost is $l \times S_p + n^2 \times S_g + S_f + 5S_t + S_k + (o+1)S_I$ for a mobile device, $l \times S_p + n^2 \times S_g + S_f + 2r \times S_t + S_k + (r+o)S_I$ for the MSP, and $n^2 \times S_g + S_f + 5r \times S_t + S_k + S_I$ for a sensor node. The storage cost of a sensor node is much less than that of a mobile device or the MSP. Moreover, the cost of sensor nodes increases with the size of initial key space and the number of mobile devices. The initial key space size and the number of mobile devices have an upper bound due to the limitation of the storage resource of a sensor node. It is clear that by our scheme, the majority of the computation and storage burden has been left to mobile devices and the MSP while saving the resource of sensor nodes and guaranteeing the security performance at the same time.

3.2 An Authentication Scheme with Identity-Based Cryptography for M2M Security

3.2.1 Identity-Based Cryptography (IBC)

IBC enables any party to compute a public key based on a known identity value. There is a private key generator (PKG) which is responsible for generating the corresponding private keys. During the system initialization, the PKG first publishes the public parameters and retains the master secret key. Any party could compute other party's public key using the corresponding identity value and public parameters. And for each party, the PKG computes

the corresponding private key with the master secret key. Obtaining the public key and private key, a party could encrypt, sign and decrypt the messages transmitted in the system.

From the weaknesses mentioned in Chapter 1, we know that a reliable cryptosystem is required because the information transmitted in M2M system may be easily eavesdropped. In the M2M communication systems, the number of M2M devices would be quite large. If the symmetric cryptography is adopted in a M2M communication system, each communication pair would need a unique secret key. Then the quantity of required secret keys for the overall system would be extremely huge. It would lead to the difficulties of key management. If the asymmetric cryptography is adopted in the M2M communication system, each communication entity only need one public key and one private key to communicate with other entities and the amount of required keys for the whole system would be much less. So the asymmetric cryptography is a better choice for the M2M communication system due to its advantage on key management issue. However, the asymmetric cryptography costs more resource while the sensor nodes in M2M have limited resource and the management of certificates is also a burden to the M2M system. Based on the two reasons, it occurs to us that the Identity-based cryptography (IBC) technology should be the best choice to apply to the M2M system.

We recommend the IBC to be applied to the M2M system because it provides similar level of security as the traditional asymmetric cryptography while possesses several advantages: the IBC utilizes the users' identities instead of digital certificates to generate the public keys which reduces the complexity of a cryptography system by eliminating

the need for generating and managing user certificates [39]; the IBC requires much lower resource regarding process power, storage space and communication bandwidth and provides pairwise keys without any interaction between nodes [40].

In literature, various kinds of IBC have been proposed. The IBC was firstly proposed by Shamir in [41] through implementing an email-address based public-key infrastructure (PKI). However, IBC kept being an open problem for lots of years due to the lack of concrete solutions until Boneh came up with the first practical implementation in [42]. In the same year, C.Cock [43] also proposed an IBC implementation which is considered as impractical due to the length of ciphertext. Later on, Ben Lynn [44] proposed an authenticated identity-based encryption scheme which integrated authentication with encryption in the Boneh-Franklin IBE system. Through his scheme, the ciphertext is also the message authentication code which eliminates the need of extra signature. Another variant of IBC is the hierarchical IBC which appeared in [45] [46]. R. Sakai and M. Kasahara [47] focused their work on the reduction of computation in IBC. Due to the merit of IBC, variants of IBC have been applied to many different communication systems to solve the security problems, such as authentication, key agreement and privacy etc, in [48] [49] [50] [51] [34] [52] [53] [54].

However, the previously mentioned IBC variants and applications still possess an inherent problem: key escrow. The key escrow problem means master secret key is stored in the private key generator (PKG). So the leakage of master secret key or a compromised PKG could lead to the compromise of all public-private key pair. In recent years, many efforts have

also been made to solve the key escrow problem in IBC system [55] [56] [57] [58]. The first three ones have solved the key escrow problem through introducing new entities into the system, such like key privacy authorities and identity-certifying authority. The drawback of their scheme is that they produce a more complex system and make the private key distributing issue more complicated. The last one proposed by Yan Zhu et al solved the key escrow problem without bringing any new entities into the system and only adding a simple algorithm to the private key generation procedure. Their solution is similar as the certificateless cryptography which is a variant of IBC. The difference is that the solution proposed by Yan Zhu et al is constructed based on ElGamal while the certificateless cryptography is based on elliptic-curve pairings. Yan Zhu et al indicate that with proper parameter choice, their solution can be more computation efficient than elliptic-curve pairings based solution which means a lot to the devices with limited energy. However their work only focused on the building of encryption system without considering the signature problem.

From the above discussion, in order to construct an effective authentication scheme for M2M, applying the IBC to the authentication scheme is one of the best choices. To satisfy the resource limitation, the authenticated IBC [44] should be applied since it integrates the signature function into the encryption which could save the resource used to compute and verify signature. To solve the key escrow problem, the scheme in [58] should be applied since it keeps the system as simple as the traditional IBC and may cost even less energy than the certificateless cryptography. Motivated by the characteristic of the two IBCs, we integrate the

two IBC schemes to a new scheme which possesses the advantages desired by the M2M system and propose an authentication scheme based on the integrated IBC scheme to ensure the authentication process is secure and resistant to key leakage.

3.2.2 M2M System Model

The M2M system model used in the scheme is the same as that used in the previous scheme which is shown as Figure5. In our scheme, the MSP plays the role of PKG. At first, the MSP generates master secret keys, public parameters, different identities and private keys. Then, MSP publishes the public parameters, distributes identities and private keys to corresponding mobile devices and sensor nodes and retains the master secret keys. Obtaining the identities' value, mobile devices or sensor nodes can compute public keys.

3.2.3 Decisional Diffie-Hellman Assumption

Definition (DDH Assumption) Let G be a (multiplicative) cyclic group and g be a generator of G. We say the Decisional Diffie-Hellman Assumption (DDH) assumption holds in G, if for any probability polynomial time (PPT) algorithm Al, we have

$$|Pr[Al(g, g^x, g^y, g^{xy}) =1] - Pr[Al(g, g^x, g^y, g^z) =1]| < \epsilon,$$

where ϵ is negligible and the probability is taken over the random choice of x, y, z and the random bits used by Al.

3.2.4 Parameters and Functions

When MSP setups the system, it will generate the following parameters with a parameter generator and security parameter k:

- A large prime number p.

- A p-order multiplicative cyclic group G such that the DDH assumption holds.
- g: a group generator of G.
- A master secret key $msk = (a, b)$ where the integers a, b are randomly selected from Z_p^* $= \{z|1 \leq z \leq p\text{-}1\}$.
- $w=2^S$ for $s \leq k$.
- $A=g^a \in G$ and $B=g^b \in G$.
- Cryptographic hash function H_1: $\{0, 1\}^* \rightarrow Z$ and H_2: $G \rightarrow \{0, 1\}^n$ where $\{0, 1\}^*$ means a group of bit strings whose length is indeterminate and $\{0, 1\}^n$ means a group of bit strings of which length is n.
- The message space is $M=\{0, 1\}^n$ and the ciphertext space is $C \subseteq G \times G \times \{0, 1\}^n$.
- T: The timestamp which indicates whether a message is valid.

3.2.5 The Integrated IBC Scheme

There are four algorithms in the integrated IBE scheme: extract, publish, encrypt and decrypt.

- **Extract:** Given an ID $\in \{0, 1\}^*$.

1. Compute the public key as $Q_{ID} = H_1(ID) \in Z$ and chooses a uniformly random ε_i.

2. Compute the partial private key as $d_{ID} = \frac{b}{a+Q_{ID}} + w\varepsilon_i mod\ p \in Z$.

3. Compute $E_{ID} = g^{-w\varepsilon_i(a+Q_{ID})} \in G$.

4. The Q_{ID} and E_{ID} are published while d_{ID} is sent to the corresponding entity in private.

- **Publish:** This procedure is not only for the MSP but also for the mobile devices and sensor nodes. Given the entity's ID ID_i, choose a uniformly random u_i from Z_p^*.

  1. Compute $U_{IDi} = B^{u_i} = g^{bu_i} \in G$, $A_{IDi} = A^{u_i} = g^{au_i} \in G$ and $g_{IDi} = g^{u_i} \in G$. Let $(pk_{IDi}, sk_{IDi}) = (\{U_{IDi}, A_{IDi}, g_{IDi}\}, u_i)$.
  2. After the (pk_{IDi}, sk_{IDi}) is produced, the entity should send the pk_{IDi} to the MSP and keep the sk_{IDi}.

- **Encrypt:** Given the message $M \in \{0, 1\}^n$, the sender's ID ID_i, the receiver's ID ID_j and the receiver's $pk_{IDj} = \{U_{IDj}, A_{IDj}, g_{IDj}\}$.

  1. Compute $Q_{IDj} = H_1(IDj)$ and choose a uniformly random $r \in Z_p^*$.
  2. Compute $c_1 = (A_{IDj} * g_{IDj}^{Q_{IDj}})^{r+sk_{IDi}} \in G$.
  3. Compute $c_2 = (E_{IDj})^{r+sk_{IDi}} \in G$.
  4. Compute $c_3 = M \oplus H_2((U_{IDj})^r) \in \{0, 1\}^n$. The '$\oplus$' denotes the bitwise exclusive-or (XOR) of two bit strings or two octet strings.

  Finally, the ciphertext is $c_{IDj} = (c_1, c_2, c_3)$.

- **Decrypt:** Given the sender's ID ID_i. To decrypt c_{IDj}, using the partial private key d_{IDj} and sk_{IDj} :

  1. Compute $P = (c_2)^{sk_{IDj}} * (c_1)^{d_{IDj}} * (U_{IDi})^{-sk_{IDj}} \in G$.
  2. Compute $M' = c_3 \oplus H_2(P) \in \{0,1\}^n$.

  The decryption result is M'.

The validity of the above process is described as follows:

$$P = (c_2)^{sk_{IDj}} * (c_1)^{d_{IDj}} * (U_{IDi})^{-sk_{IDj}}$$

$$=((E_{IDj})^{r+sk_{IDi}})^{sk_{IDj}}*((A_{IDj}*\ g_{IDj}^{Q_{IDj}})^{r+sk_{IDi}})^{d_{IDj}}*(U_{IDi})^{-sk_{IDj}}$$

$$=\ g^{-w\varepsilon_i u_j(a+Q_{IDj})(r+u_i)}*g^{(au_j+u_jQ_{IDj})(r+u_i)\,(\frac{b}{a+Q_{IDj}}+w\varepsilon_j)}*g^{-bu_iu_j}$$

$$=\ g^{-w\varepsilon_i u_j(a+Q_{IDj})(r+u_i)}*g^{u_jb(r+u_i)+(a+Q_{IDj})u_j(r+u_i)\,w\varepsilon_j}*\ g^{-bu_iu_j}$$

$$=g^{u_jb(r+u_i)}*\ g^{-bu_iu_j}$$

$$=g^{bu_jr}$$

$$=U_{IDj}{}^{r}$$

3.2.6 The Proposed Scheme

In this section, we describe our proposed authentication scheme consisting of three phases: System initialization phase, Authentication phase and Key updating phase. The notations used in our scheme are defined in Table III.

TABLE III. DEFINITION OF NOTATION

ID_i	Identity of mobile device *i*
IP_j	Identity of sensor node *j*
ID_{MSP}	Identity of the MSP
(*a, b*)	Master secret key
g	a group generator of *G*
A	$g^a \in G$
B	$g^b \in G$
Q_{ID}	The public key corresponding to ID. $Q_{ID} = H_1(ID)$
d_{ID}	The partial private key corresponding to ID. $d_{ID} = \frac{b}{a+Q_{ID}} + w\varepsilon_i mod\ p \in Z$

E_{ID}	$g^{-w\varepsilon_i(a+Q_{ID})} \in G$
r	A number randomly selected by the MSP
sk_{IDi}	A number u randomly selected by the sensor or mobile device
U_{IDi}	$B^{u_i} = g^{bu_i} \in G$
A_{IDi}	$A^{u_i} = g^{au_i} \in G$
g_{IDi}	$g^{u_i} \in G$
T	The timestamp which indicates whether a message is valid
$E_{IDij}(m)$	Encryption of message m with Q_{IDj}, pk_{IDj}, E_{IDj} and sk_{IDi}

- **System Initialization Phase**

During the system initialization, the processes taken place in the MSP, mobile devices and sensor nodes are shown as follow:

- MSP:
  1. MSP publishes <$p, A, B, g, w, G, H_1, H_2$> as public parameters to all mobile devices and sensor nodes.
  2. MSP distributes a unique identity value to each mobile device and sensor node.
  3. For each entity, taking ID_i as the example ID, the MSP calculates Q_{IDi}, d_{IDi} and E_{IDi}. MSP distributes Q_{IDi} and d_{IDi} to corresponding entity and publishes E_{IDi}.
  4. MSP randomly chooses a sk_{MSP}: u_{MSP} and calculates pk_{MSP}: $\{U_{MSP}, A_{MSP}, g_{MSP}\}$. MSP publishes the pk_{MSP} and retains the sk_{MSP}. MSP also generates its own ID: ID_{MSP} and calculates Q_{MSP}, d_{MSP} and E_{MSP}. MSP publishes Q_{MSP} and E_{MSP}.
- Mobile devices and sensor nodes:

1. The mobile device or sensor node randomly chooses a sk_{ID}: u_{ID} and calculates the corresponding pk_{ID}: $\{U_{ID}, A_{ID}, g_{ID}\}$.
2. The mobile device or sensor node sends the pk_{ID} to the MSP and keeps the sk_{ID} as secret.

- **Authentication Phase**

In this phase, the mobile device *i*, the target sensor node *j* and the MSP will be mutually authenticated by the proposed IBC-based authentication scheme. The process is shown as Figure9. The details of authentication process are described as follows.

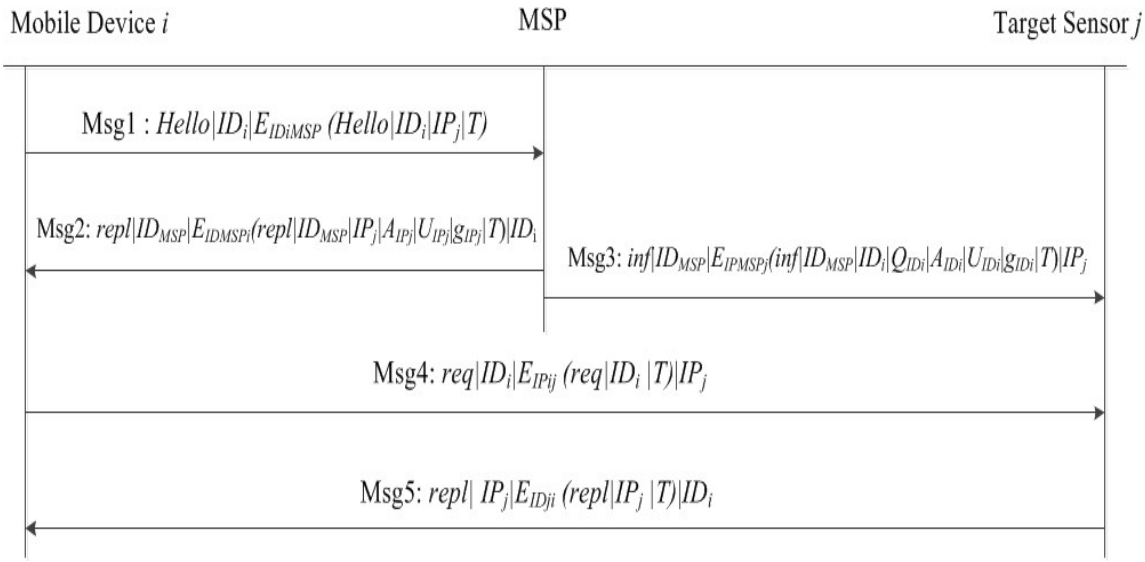

Figure9. The Process of Authentication Phase

- Msg1 : $Hello|ID_i|E_{IDiMSP}\ (Hello|ID_i|IP_j|T)$

Msg1 is the message to initialize a session for the data collection purpose. When the mobile device *i* wants to build a new connection with sensor node *j* to collect information, it will utilize Q_{MSP}, E_{MSP}, pk_{MSP} *and* sk_{IDi} to encrypt its ID_i, the ID of the target sensor node IP_j

and timestamp T. Then, the mobile device i will send the encrypted message as the Msg1 to its neighbor sensor nodes. And the sensor nodes will forward the Msg1 to the MSP.

- Msg2: $repl|ID_{MSP}|E_{IDMSPi}(repl|ID_{MSP}|IP_j|A_{IPj}|U_{IPj}|g_{IPj}|T)|ID_{\text{i}}$

When the MSP receives the Msg1, it will decrypt Msg1 with d_{MSP}, sk_{MSP} and U_{IDi} according to ID_i. Then the MSP gets the ID_i, IP_j and T encrypted in Msg1. The MSP will firstly verify the value of T. If T indicates the Msg1 is invalid, the MSP will ignore the Msg1. Otherwise the MSP continues to verify the ID_i and check whether the encrypted ID_i is the same as the ID_i which is transmitted as plaintext. If ID_i is invalid or the two ID_i is different, it means the Msg1 is forged or tampered and should be discarded. Otherwise the MSP encrypts ID_{MSP}, IP_j A_{IPj}, U_{IPj}, g_{IPj} and T with Q_{IDi}, E_{IDi}, pk_{IDi} *and* sk_{MSP}. Finally, the MSP sends the encrypted message as Msg2 to mobile device i.

- Msg3: $inf|ID_{MSP}|E_{IPMSPj}(inf|ID_{MSP}|ID_i|Q_{IDi}|A_{IDi}|U_{IDi}|g_{IDi}|T)|IP_j$

After the MSP sends Msg2 to mobile device i, it also encrypts ID_{MSP}, ID_i, Q_{IDi}, A_{IDi}, U_{IDi}, g_{IDi} and T with Q_{IPj}, E_{IPj}, pk_{IPj} *and* sk_{MSP}. Then the MSP sends the encrypted message as Msg3 to the target sensor node j to inform the sensor that the mobile device i wishes to build a connection with it. After the target sensor j receives the Msg3, it will decrypt the message with d_{IPj}, sk_{IPj} and U_{MSP} and get the ID_i, Q_{IDi}, A_{IDi}, U_{IDi} and g_{IDi}.

- Msg4: $req|ID_i|E_{IPij}$ $(req|ID_i\ |T)|IP_j$

After receiving the Msg2, the mobile device i will decrypt the message with d_{IDi}, sk_{IDi} and U_{MSP}. Then the mobile device i gets the ID_{MSP}, IP_j, A_{IPj}, U_{IPj}, g_{IPj} and T encrypted in Msg2. The device i will firstly verify the value of T. If T indicates the Msg2 is invalid, the

device *i* will ignore the Msg2. Otherwise the device *i* continues to check whether the encrypted ID_{MSP} is the same as the ID_{MSP} which is transmitted as plaintext. If the two ID_{MSP} is different, it means the Msg2 is forged or tampered and should be discarded. Otherwise the device *i* computes Q_{IPj} and encrypts ID_i and *T* with Q_{IPj}, E_{IPj}, pk_{IPj} *and* sk_{IDi}. Finally, the device *i* sends the encrypted message as Msg4 to the target sensor *j* to request the communication.

- Msg5: $repl| IP_j|E_{IDji}(repl|IP_j|T)|ID_i$

After receiving the Msg4, the target sensor *j* will decrypt the message with d_{IPj}, sk_{IPj} and U_{IDi}. Then the target sensor *j* gets the ID_i and *T* encrypted in Msg4. The sensor *j* will firstly verify the value of *T*. If *T* indicates the Msg4 is invalid, the sensor *j* will ignore the Msg4. Otherwise the sensor *j* continues to check whether the encrypted ID_i is the same as the ID_i which is transmitted as plaintext. If the two ID_i is different, it means the Msg4 is forged or tampered and should be discarded. Otherwise the sensor *j* encrypts IP_j and *T* with Q_{IDi}, E_{IDi}, pk_{IDi} *and* sk_{IPj}. Finally, the sensor *j* sends the encrypted message as Msg5 to the mobile device *i* to indicate the session is constructed successfully.

By now, the mobile device *i* has built a secure session with the sensor node *j* through utilizing Q_{IDi}, d_{IDi}, E_{IDi}, sk_{IDi}, pk_{IDi}, Q_{IPj}, d_{IPj}, E_{IPj}, sk_{IPj} and pk_{IPj}. The information collected by sensor node *j* could be delivered to the mobile device *i* with high security.

- **Key Updating Phase**

The phase is for the practical applications which have the need to update keys to prevent the system from the password guessing attack and reduce the threat of key leakage.

In the phase, the public key Q_{ID} and the partial private key d_{ID} stay the same and the pk_{ID} and sk_{ID} are updated. The updating process is: the entity which wants to update the key chooses a new random number u_{ID}' as new sk_{ID}' and computes the new pk_{ID}'. Then the entity encrypts the new pk_{ID}' with Q_{MSP}, E_{MSP}, pk_{MSP} and sk_{ID} and sends the encrypted message to the MSP. After the MSP receives and decrypts the message, it replaces the pk_{ID} with the new pk_{ID}' in its database. Then the key updating process finishes.

3.2.7 Security Analysis

In this section, we will analyze the security function of our authentication scheme. Firstly, we analyze the security design of the integrated IBC scheme. Secondly, we analyze the security from the perspective of mutual authentication. Thirdly, we analyze the scheme's ability of standing against various attacks. Finally, we give the protocol analysis with Burrows–Abadi–Needham logic (BAN Logic).

1. The Security Design of the Integrated IBC Scheme

The design of the integrated IBC scheme makes the scheme gain two characteristics: 1) the message is authenticated when it is encrypted; 2) the scheme is without key escrow problem.

1) When sender *IDi* communicates with receiver *IPj*, the sender needs to use sk_{IDi} to encrypt the message and the receiver needs to use U_{IDi} to decrypt the message. Only the correct (sk_{IDi}, U_{IDi}) pair could ensure the message is encrypted and decrypted correctly. That means only if the message is encrypted by legitimate sender ID_i, the receiver could decrypt

it by corresponding U_{IDi}. So the message is authenticated with the encryption and no more signatures are needed.

2) When a receiver IP_j wants to decrypt a message, it needs to use d_{IPj} and sk_{IPj}. The d_{IPj} is known to the receiver and the MSP and the sk_{IPj} is only known to the receiver. So even the MSP is compromised or the partial private key d_{IPj} is leaked, the message could still only be decrypted by the receiver due to the sk_{IPj}. So the existence of sk_{IPj} solves the key escrow problem. What's more, the updating of sk_{IPj} improves the security of the IBC scheme.

2. Mutual Authentication

A mutual authentication among mobile devices, sensor nodes and the MSP can be accomplished with the integrated IBC. The ID of mobile device is verified by the MSP via the IBC mechanism. Only the message encrypted by legitimate mobile devices could be decrypted by the MSP with the corresponding U_{ID}. Moreover, the application of the secret key sk_{ID} ensures that only the legitimate mobile device could make the message authenticated with encryption and only the target sensor could decrypt that message and vice versa. So the integrated IBC scheme guarantees the mutual authentication among the MSP, mobile devices and sensor nodes.

3. Ability against Multiple Attacks

- **Against Man-in-the-Middle Attacks**

Man-in-the-Middle attack means an attacker between the sender and receiver receives a message from the sender, tampers the message, inserts or steals the information in the message and forwards the message to the receiver without its attention.

In our scheme, although an attacker could obtain the transmitted message to get the ID of sender and receiver, it has no knowledge of the corresponding d_{ID} and sk_{ID}. So the attacker could not decrypt the message and insert any illegal information or steal private information in the connection. On the other hand, even the attacker changes the ID which are transmitted as plaintext in a message, the altered message will be ignored since the encrypted message is authenticated.

- **Against Reply Attacks**

Reply attack means an attacker gets a message and replies it to the receiver to pretend that the legitimate sender sends the message again.

However, in our scheme, timestamp T has been also encrypted in a message. Each time a receiver decrypts a message, it will check the value of T. If T indicates the message is invalid, the message will be ignored. Since the attacker is not able to decrypt a message, the value of T cannot be changed. So even an attacker ceaselessly replies the messages, the replied messages will all be discarded as the T stays the same.

- **Against DoS Attacks**

DoS attack means an attacker could send a “hello” message containing a certain device’s ID to the MSP continually. After receiving many messages which contain the same ID, the MSP will reject to serve for the ID since it considers the corresponding device has been

compromised. The “hello” message could be a legitimate message transmitted previously or a forged message.

In our system, if an attacker sends a previously legitimate message to the MSP, the message will be ignored due to the value of T. If an attacker forges a “hello” message, the message will also be rejected since the attacker does not have the correct sk_{ID} and the forged message could not be decrypted correctly. So the attacker could never launch a DoS attack.

- **Against Impersonation Attacks**

Impersonation attack means an attacker disguises as a legal entity to communicate with the MSP or legitimate entities.

In our authentication scheme, to impersonate an entity needs the corresponding secret key sk_{ID}. The sk_{ID} is chosen by a legitimate entity randomly, only stored in the entity and never show up in any messages. It’s very hard for an attacker to impersonate a legitimate entity since it doesn’t have the access to the corresponding sk_{ID}.

- **Against Compromised Attacks**

Compromised attack means the legitimate entity or the MSP is compromised and the partial private key d_{ID} and secret key sk_{ID} are leaked. Then the attacker could utilize the d_{ID} and sk_{ID} to damage the system.

In our scheme, if the MSP is compromised, the attacker could not threat the system since the entities’ sk_{ID}s are not stored in the MSP. The attacker still could not decrypt the transmitted messages or impersonate a legitimate entity. If a mobile device or sensor node is compromised, the attacker could only impersonate the compromised entity and decrypt the

messages sent to the compromised entity. The d_{ID} and sk_{ID} of uncompromised entity are still secret to the attacker and the majority of the system keeps safe. So the damages caused by the compromised attack are quite limited.

4. Protocol Analysis with BAN Logic

BAN logic is the logic of belief for defining and analyzing communication protocols. The goal of using BAN Logic is to analyze authentication protocols by deriving the beliefs that the correct execution of protocol by legitimate principals could generate a trustworthy result. The advantage and disadvantage of the BAN logic are shown as follow.

Advantage:

(1) BAN logic introduces a set of simple and powerful notations which make its proof relatively short.

(2) The BAN logic is easy to use.

(3) The logic postulates could be straightforwardly applied to derive BAN beliefs.

(4) The notion of freshness used in BAN logic avoids the use of timestamps and simplifies the proofs.

(5) BAN logic helps to clarify the protocol's assumptions by stating them formally and uncover the implicit assumptions.

Disadvantage:

(1) The BAN logic is not suitable to model some protocol execution and attacks behaviors accepted nowadays. The idealization step in the modeling process may make the model imperfect.

(2) The BAN logic is only useful to analyze authentication protocols.

(3) The BAN logic assumes that all participant principals in the protocol are honest, which means every principal believes each message it sends is true. However, the assumption of honesty is not logical.

Although there are some drawbacks exist in the BAN logic, the drawbacks have little influence on our protocol analysis since the protocol models in our schemes are authentication models which the BAN logic is helpful and we only utilize the BAN logic to derive the logic correctness of our schemes without analyzing the attacks in which the principals may not be honest.

There are four steps in the protocol analysis using BAN Logic.

1. Idealize the protocol.

2. Write initial state assumptions.

3. Protocol annotation.

4. Beliefs derivation with logic.

The notations being used are shown as follows:

D: mobile device *i*. *S*: target sensor node *j*. *M*: MSP. *PK(P,k)*: *k* is a public key of *P*.

$P \overset{k}{\leftrightarrow} Q$: *k* is the shared key between *P* and *Q*.

- **Idealization**

Message2: $M \rightarrow D$: $\{ PK(S, pk_{IPj}), fresh(pk_{IPj}), n_T \}_{\mathrm{KMD}}$ from *M*

Message3: $M \rightarrow S$: $\{ PK(D, pk_{IDi}), fresh(pk_{IDi}), n_T \}_{\mathrm{KMS}}$ from *M*

In idealization form, we regard the *T* as nonce n_T. We also regard the Q_{ID}, d_{ID}, E_{ID}, pk_{ID}

and sk_{ID} which are believed by two entities as the shared key K between them. Note that Q_{ID}, d_{ID}, E_{ID} and sk_{ID} are already trusted since the initialization phase, so making the pk_{ID} trusted by each other is the main concern. Since the Msg1, Msg4 and Msg5 are without any public key or private key, we omit the Msg1, Msg4 and Msg5.

- **Initial State Assumption**

P1. *D* believes $D \overset{KMD}{\longleftrightarrow} M$

P2. *S* believes $S \overset{KMS}{\longleftrightarrow} M$

P3. *D* believes *M* controls *PK(S, k)*

P4. *S* believes *M* controls *PK(D, k)*

P5. *D* believes *M* controls *fresh(PK(S, k))*

P6. *S* believes *M* controls *fresh(PK(D, k))*

P7. *D* believes *fresh(n_T)*

P8. *S* believes *fresh(n_T)*

- **Protocol Annotation**

The annotation states assumptions based on the idealization protocol. It is shown below:

P9. *D* received $\{ PK(S, pk_{IPj}), fresh(pk_{IPj}), n_T \}_{\text{KMD}}$ from *M*

P10. *S* received $\{ PK(D, pk_{IDi}), fresh(pk_{IDi}), n_T \}_{\text{KMS}}$ from *M*

- **Beliefs derivation**

In the derivations below, every line is followed by the rule by which it was derived.

1. $D \models M \sim \{PK(S, pk_{IPj}), \#(pk_{IPj}), n_T \}$:

D believes *M* said $\{PK(S, pk_{IPj}), fresh(pk_{IPj}), n_T \}$. By Message Meaning using P1, P9.

2. $D \models \#\{PK(S, pk_{IPj}), \#(pk_{IPj}), n_T\}$:

   D believes $fresh\{PK(S, pk_{IPj}), fresh(pk_{IPj}), n_T\}$. By Freshness Conjuncatenation using 1, P7.

3. $D \models M \models \{PK(S, pk_{IPj}), \#(pk_{IPj}), n_T\}$:

   D believes M believes $\{PK(S, pk_{IPj}), fresh(pk_{IPj}), n_T\}$. By Nonce Verification using 2, 1.

4. $D \models M \models PK(S, pk_{IPj})$:

   D believes M believes $PK(S, pk_{IPj})$. By Belief Conjuncatenation using 3.

5. $D \models M \models \#(PK(S, pk_{IPj}))$:

   D believes M believes $(fresh\ PK(S, pk_{IPj}))$. By Belief Conjuncatenation using 3.

6. $D \models PK(S, pk_{IPj})$:

   D believes $PK(S, pk_{IPj})$. By Jurisdiction using 4, P3.

7. $D \models \# PK(S, pk_{IPj})$:

   D believes $(fresh\ PK(S, pk_{IPj}))$. By Jurisdiction using 5, P5.

We have derived mobile device i's belief in the goodness and freshness of pk_{IPj}. We now turn to target sensor j.

8. $S \models M \mid\sim \{PK(D, pk_{IDi}), \#(pk_{IDi}), n_T\}$:

   S believes M said $\{PK(D, pk_{IDi}), fresh(pk_{IDi}), n_T\}$. By Message Meaning using P2, P10.

9. $S \models \#\{PK(D, pk_{IDi}), \#(pk_{IDi}), n_T\}$:

   S believes $fresh\{PK(D, pk_{IDi}), fresh(pk_{IDi}), n_T\}$. By Freshness Conjuncatenation using 8, P8

10. $S \models M \models \{PK(D, pk_{IDi}), \#(pk_{IDi}), n_T\}$:

S believes *M* believes $\{PK(D, pk_{IDi}), \#(pk_{IDi}), n_T\}$. By Nonce Verification using 9, 8.

11. $S \models M \models (PK(D, pk_{IDi}))$:

S believes *M* believes $(PK(D, pk_{IDi}))$. By Belief Conjuncatenation using 10.

12. $S \models M \models \#(PK(D, pk_{IDi}))$:

S believes *M* believes *fresh* $(PK(D, pk_{IDi}))$. By Belief Conjuncatenation using 10.

13. $S \models PK(D, pk_{IDi})$:

S believes $(PK(D, pk_{IDi}))$. By Jurisdiction using 11, P4

14. $S \models \#(PK(D, pk_{IDi}))$:

S believes $(fresh\ PK(D, pk_{IDi}))$. By Jurisdiction using 12, P6

Now we have also derived target sensor *j*'s belief in the goodness and freshness of pk_{IDi}.

By now, we have already derived the target sensor node's belief in pk_{IDi} and the mobile device's belief in pk_{IPj}. With their beliefs' in Q_{ID}, d_{ID}, E_{ID}, sk_{ID} and pk_{ID}, the mobile device *i* and target sensor node *j* could set up a session with the integrated IBC scheme securely. The protocol analysis with BAN Logic indicates the correctness of the protocol design.

3.2.8 Efficiency Analysis

In the section, we evaluate the computation cost of our scheme. The notations are shown in Table IV.

TABLE IV. DEFINITION OF NOTATION

C_{H1}	Computation cost of hash function H_1
C_{H2}	Computation cost of hash function H_2
C_e	Computation cost of modular exponentiation
C_m	Computation cost of multiplication
C_o	Computation cost of bitwise exclusive-or

According to the integrated IBC scheme, the computation cost of publish algorithm is $3C_e$. The computation cost of encryption algorithm is $C_{H1}+C_{H2}+4C_e+C_m+C_o$ for mobile device or MSP and $C_{H2}+4C_e+C_m+C_o$ for sensor node. The computation cost for decryption algorithm is $C_{H2}+3C_e+2C_m+C_o$.

According to the authentication scheme, the total computation cost is shown as follow:

1. For mobile device, there are one time of publish, two times of encryption and two times of decryption. Note that the Q_{IP} only need to compute for once. So the total computation cost is $C_{H1}+4C_{H2}+17C_e+6C_m+4C_o$.
2. For MSP, one time of publish, two times of encryption and one time of decryption. Note that the MSP compute Q_{ID} for sensor node in Msg3. So the total computation cost is $3C_{H1}+3C_{H2}+14C_e+4C_m+3C_o$.
3. For sensor nodes, there are one time of publish, one time of encryption and two times of decryption. So the total computation cost is $3C_{H2}+13C_e+5C_m+3C_o$.

It's clear that the computation cost of sensor nodes is less than that of mobile devices and the MSP. The sensor nodes receive Q_{ID} from the MSP instead of computing by themselves. The allocation of computing task could help to save the computation resource of sensor nodes. What's more, the number of C_e is the most. So choosing proper parameters to make the modular exponentiation more efficient is very important to the system performance.

3.3 Summary

The two proposed authentication schemes for single domain M2M security in this chapter can ensure a safe session between mobile devices and sensor nodes. The proposed dynamic-encryption scheme could avoid direct stealing and modifying of the mobile devices' and the sensors' ID. The dynamic-key generation mechanism in dynamic-encryption scheme could not only provide a reliable one-time-password among MSP, mobile devices and sensor nodes but also save the computing resource of the sensor nodes. The application of integrated IBC in the authentication scheme with IBC could achieve the message authentication without key escrow problem. The regular updating of secret key could also make the key guessing attack meaningless. Our security analysis indicates that the mutual authentication and the ability of withstanding multiple attacks could be accomplished by the proposed solutions.

Chapter 4. AUTHENTICATION SCHEME FOR MULTI-DOMAIN M2M SECURITY

In previous chapter, we propose two authentication schemes for single domain M2M security. However, in real scenario, the amount of the sensor nodes in M2M communication system in CPS would be huge. The single domain assumption in the M2M model is impractical in large system. So in this chapter, we propose an authentication scheme for multi-domain M2M security. In the scheme, we apply an authenticated certificateless encryption scheme [59] with authenticated and key escrow free features. The security analysis of the encryption scheme could be found in [59]. Different from the scheme which is based on ElGamal in previous chapter, the authenticated certificateless encryption scheme in this chapter is based on elliptic curve pairing and it could be replaced easily by the one based on ElGamal. The reason we apply such a scheme is to make the user could choose any of the two encryption schemes depending on the specific applications. The authentication scheme is introduced as follow.

4.1 M2M System Model

The M2M system model used in our scheme is shown as Figure10. There are five parties—mobile devices, sensor nodes, gateways, access points (APs) and the MSP in the M2M system. The communication medium among the mobile devices, sensor nodes, gateways and APs is WiFi technology and the medium between the MSP and APs is optical fiber. The mobile device is carried by a user to gather information from sensor nodes. Sensor nodes could relay messages for mobile devices or exchange information between

each other. The gateways are in charge of the communication among different domains and forwarding messages from sensor nodes to the MSP or mobile devices. The MSP works as an authentication center to verify the ID of mobile devices, sensor nodes or gateways and generates public keys and partial private keys. The APs transmit messages for the MSP, mobile devices and gateways with optical fiber because the physical distance is too far for WiFi. The secure communication over the communication links consist of WiFi is the concern of this study while the links consist of optical fiber are considered as secure. It is assumed that the MSP, gateways and mobile devices are powerful enough to run complex computation which means they could equip the certificateless mechanism, while the sensor nodes could only afford efficient AES function for encryption. In the model, each gateway corresponds to one domain.

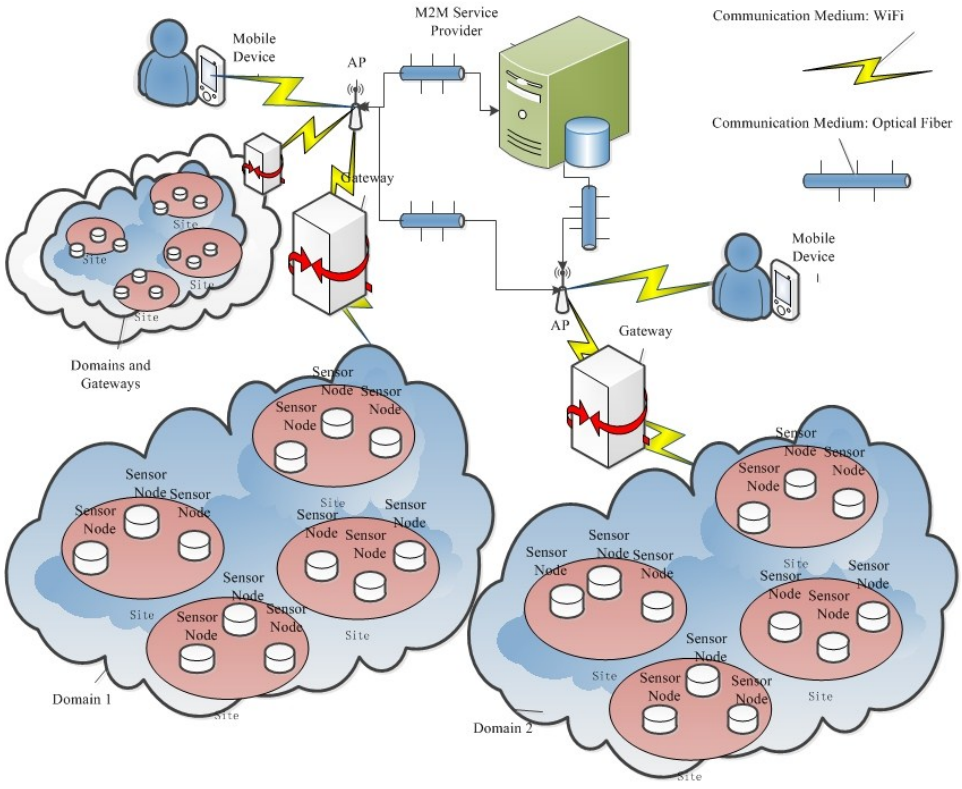

Figure10. M2M System Model

4.2 Discrete Logarithm Problem

Definition (DLP) Let G be a multiplicative group and g be a generator of G. Let $<g>$ be the cyclic subgroup generated by g. The discrete logarithm problem for G is stated as:

Given $g \in G$ and $a \in <g>$, find an integer x such that $g^x = a$.

4.3 Bilinear Diffie-Hellman Problem

Definition (BDHP) Let G_1 and G_2 be two cyclic groups of prime order q and P be a generator of G_1. Let e: $G_1 \times G_1 \rightarrow G_2$ be a bilinear map. The BDHP in (G_1, G_2, e) is as follow:

Given (P, aP, bP, cP) for some a, b, $c \in Z_q$, compute $v \in G_2$ such that $v = e(P, P)^{abc}$.

4.4 Parameters and Functions

(1) For authenticated certificateless encryption scheme

- A security parameter k.
- A large prime q.
- A q-order additive group G_1 of points of an elliptic curve $y^2 = x^3 + ax + b$ over a finite field. A q-order multiplicative group G_2 of a finite field. The Discrete Logarithm Problem (DLP) is computationally hard in both G_1 and G_2.
- P: (x_p, y_p) a generator of G_1.
- A master secret key s randomly selected from a group $Z_q = \{a | 1 \leq a \leq q-1\}$.
- Sub-secret key sd randomly selected from a group Z_q. Each entity generates a sd.
- $P_{pub} = sP$.
- Cryptographic hash function H_1: $\{0, 1\}^* \rightarrow G_1$, H_2: $G_1 \rightarrow \{0, 1\}^n$, H_3: $\{0, 1\}^n \times G_2 \rightarrow \{0, 1\}^n$, H_4: $\{0, 1\}^n \times \{0, 1\}^n \rightarrow Z_q$ and H_5: $\{0, 1\}^n \rightarrow \{0, 1\}^n$ where $\{0, 1\}^*$ means a group

of bit strings whose length is indeterminate and $\{0, 1\}^n$ means a group of bit strings of which length is n.

- A bilinear map e: $G_1 \times G_1 \rightarrow G_2$, which can be constructed from a Weil or a Tate pairing on an elliptic curve over a finite field. The map e has the following properties: (1) Bilinearity: $\forall P, Q, R, S \in G_1$, $e(P+Q, R+S) = e(P, R)\, e(P, S)\, e(Q, R)\, e(Q, S)$;i.e.$\forall P, Q \in G_1$ and $a, b \in Z$, we have $e(aP, bQ) = e(aP, Q)^b = e(P, bQ)^a = e(P, Q)^{ab}$. (2) Nondegenerate: G_1, G_2 are groups of prime order which implies that if P is a generator of G_1 then $e(P, P)$ is a generator of G_2. (3) Computable: there exists an efficient algorithm to compute $e(P, Q)$ for any $P, Q \in G_1$. The examples of such bilinear maps for which the Bilinear Diffie-Hellman Problem (BDHP) is believed to be computationally hard are provided in [42], [60]and [61]. [42], [60] and [61] could be regarded as the more comprehensive instructions for how to select the pairing parameters in practice.
- The message space is $\{0, 1\}^n$ and the ciphertext space is $G_1 \times \{0, 1\}^n \times \{0, 1\}^n$.
- T_{stamp}: The timestamp which indicates whether a message is valid.

(2) For efficient encryption scheme

- The same efficient AES encryption function is stored at the gateways, sensor nodes and the MSP.
- The same hash function H_6 which is used to map a large and variable-sized data into a small and fixed-sized data is stored at the gateways, sensor nodes and MSP.

For each sensor node, n ordered relative node's IDs are distributed by MSP. The IDs of relative node are stored in a relative node list, which is an array $RL[n]$, in the fixed order. Each sensor node has a unique list. The whole lists are stored in the MSP.

4.5 The Authenticated Certificateless Encryption Scheme

There are five algorithms in the authenticated certificateless encryption scheme: setup, extract, publish, encrypt and decrypt.

- **Setup**

  Given a security parameter k:

  1. Run a BDH parameter generator on input k to generate a prime q, two groups G_1, G_2 of order q, and a bilinear map e: $G_1 \times G_1 \rightarrow G_2$. Choose an arbitrary generator $P \in G_1$.
  2. Pick a random $s \in Z_q$ and set $P_{pub} = sP$.
  3. Choose cryptographic hash functions H_1: $\{0, 1\}^* \rightarrow G_1$, H_2: $G_1 \rightarrow \{0, 1\}^n$, H_3: $\{0, 1\}^n \times G_2 \rightarrow \{0, 1\}^n$, H_4: $\{0, 1\}^n \times \{0, 1\}^n \rightarrow Z_q$ and H_5: $\{0, 1\}^n \rightarrow \{0, 1\}^n$.
  4. Output system parameters $<G_1, G_2, q, e, P, P_{pub}, H_1, H_2, H_3, H_4, H_5>$ and master secret key s.

- **Extract:** Given an ID $\in \{0, 1\}^*$.

  1. Computes the public key as $Q_{ID} = H_1(ID) \in G_1$.
  2. Computes the partial private key as $d_{ID} = sQ_{ID} \in G_1$.
  3. The Q_{ID} is published while d_{ID} is sent to the corresponding entity in private.

- **Publish:** This procedure is not only for the MSP but also for the mobile devices and gateways. Given the entity's ID ID_i, choose a uniformly random sd_i from Z_q.

  1. Computes the sub-public key $P_{s\text{-}IDi} = sd_i P$ and $P_{y\text{-}IDi} = sd_i Q_{IDi}$.
  2. Computes the sub-private key $d_{s\text{-}IDi} = sd_i d_{IDi} = sd_i sQ_{IDi}$.
  3. The mobile device or gateway will send the $P_{s\text{-}IDi}$ and $P_{y\text{-}IDi}$ to the MSP and keep the $d_{s\text{-}IDi}$.

- **Encrypt:** Given the message $M \in \{0, 1\}^n$, the sender's ID ID_i, the receiver's ID ID_j.

  1. Computes $Q_{IDj} = H_1 (ID_j) \in G_1$.
  2. Chooses a random $\sigma \in \{0, 1\}^n$. Sets $r = H_4 (\sigma, M)$.
  3. Compute $T = sd_i P_{s\text{-}IDj}$.
  4. The cipher text is $C = \langle rQ_{IDi}, \sigma \oplus H_3(H_2(T), e(d_{IDi}, P_{y\text{-}IDj})^r), M \oplus H_5 (\sigma) \rangle$ in which $P_{y\text{-}IDj} = sd_j Q_{IDj}$ is one of the sub-public keys of the receiver and $\oplus$ denotes the bitwise exclusive-or (XOR) of two bit strings or two octet strings.

- **Decrypt:** Let cipher text $C = \langle U, V, W \rangle$. To decrypt C, using the sub-private key $d_{s\text{-}IDj}$ and sub-public key $P_{s\text{-}IDi}$ to compute:

  1. Compute $T = sd_j P_{s\text{-}IDi}$.
  2. Computes $\sigma = V \oplus H_3(H_2(T), e(U, d_{s\text{-}IDj}))$ in which $d_{s\text{-}IDj} = sd_j sQ_{IDj}$.
  3. Computes $M = W \oplus H_5 (\sigma)$.
  4. Sets $r = H_4 (\sigma, M)$. Test that $U = rQ_{IDi}$. If not, rejects the ciphertext.
  5. Outputs M as the decryption of C.

The validity of the above process is described as follows:

$$e(d_{IDi}, P_{y\text{-}IDj})^r = e(sQ_{IDi}, sd_jQ_{IDj})^r$$

$$= e(rQ_{IDi}, sd_jQ_{IDj})^s$$

$$= e(rQ_{IDi}, sd_j\, sQ_{IDj})$$

$$= e(U, d_{s\text{-}IDj})$$

The receiver could ensure the origin of the encrypted message by checking whether the $U = rQ_{IDi}$ holds.

4.6 The Proposed Authentication Scheme

Before describing the proposed scheme, the symmetric key generation mechanism applied in the sensor domain is first introduced.

The mechanism is applied to generate the symmetric key for AES encryption function between gateways and sensor nodes. The hash function H_6 and relative node list $RL[n]$ are utilized in the mechanism. Assume the ID of source sensor node r is IP_r and the ID of the target sensor node is IP_k. Before the source sensor is verified by MSP, its symmetric key is calculated as: $key = [H_6\ (IP_r|RL_r[1]|RL_r[2]|\ldots|RL_r[n])]^{128}$, where $[m]^n$ means the most significant n bits of string m. Note that different relative node list results in different key. After the source sensor is verified, the symmetric key changes to $[H_6\ (IP_r|IP_k|RL_r[1]|RL_r[2]|\ldots|RL_r[n])]^{128}$. That ensures the same source node's key varies with the different target sensor node. What's more, the MSP could update the relative node list during the authentication process to improve the security.

In the following, we describe our proposed authentication scheme consisting of three phases: System initialization phase, Authentication phase and Key updating phase. The notations used in our scheme are defined in Table V.

TABLE V. DEFINITION OF NOTATION

ID_i	Identity of mobile device or gateway i
IP_k	Identity of sensor node k
ID_{MSP}	Identity of the MSP
s	Master secret key of mobile device or gateway
sd	Sub-secret key of mobile device or gateway
pw	Symmetric key using in AES encryption
P	The generator of G_1
P_{pub}	$P_{pub} = sP$
$P_{s\text{-}IDi}$	One sub-public key corresponding to ID_i. $P_{s\text{-}IDi} = sd_iP$.
Q_{ID}	The public key corresponding to ID. $Q_{ID} = H_1(ID)$.
$P_{y\text{-}IDi}$	The other sub-public key corresponding to ID_i. $P_{y\text{-}IDi} = sd_iQ_{IDi}$.
d_{ID}	The partial private key corresponding to ID. $d_{ID} = sQ_{ID}$.
$d_{s\text{-}IDi}$	The sub-private key corresponding to ID_i. $d_{s\text{-}IDi} = sd_i sQ_{IDi}$.
T_{stamp}	The time stamp which indicates whether the related message is valid
$E_{IDij}(M)$	Encryption of message M using the certificateless scheme with sender ID_i and receiver ID_j.
$E_{AES}(a,b)$	Encryption of message a using the symmetric key b in AES function

- **System Initialization Phase**

During the system initialization, the processes taken place in the MSP, mobile devices and gateways are shown as follow:

- MSP:

1. MSP publishes <G_1, G_2, q, e, P, P_{pub}, H_1, H_2, H_3, H_4, H_5> as public parameters to all mobile devices and gateways, publishes AES function to all gateways and sensor nodes and publishes <H_6> to all sensor nodes.
2. MSP distributes a unique identity value to each mobile device and gateways.
3. MSP distributes a unique relative node list *RL[n]* to each sensor node.
4. For each entity, taking ID_i as the example ID, the MSP calculates Q_{IDi} and d_{IDi}. MSP distributes Q_{IDi} and d_{IDi} to corresponding entity.
5. MSP generates its own ID: ID_{MSP} and calculates Q_{MSP} and d_{MSP}. It also randomly chooses a sub-secret key sd_{MSP} and calculates sub-public key $P_{s\text{-}MSP}$: $sd_{MSP}P$ and $P_{y\text{-}MSP}$: $sd_{MSP}Q_{MSP}$ and the sub-private key $d_{s\text{-}MSP}$: = $sd_{MSP}\, d_{MSP}$. MSP publishes the Q_{MSP}, $P_{s\text{-}MSP}$ and $P_{y\text{-}MSP}$ and retains the sd_{MSP}, d_{MSP} and $d_{s\text{-}MSP}$.

- Mobile devices and gateways:

1. The mobile device or gateway ID_i randomly chooses a sub-secret key sd_i and calculates sub-public key $P_{s\text{-}IDi}$: sd_iP and $P_{y\text{-}IDi}$: sd_iQ_{IDi} and sub-private key $d_{s\text{-}IDi}$: $sd_i\, d_{IDi}$.
2. The mobile device or gateway sends the $P_{s\text{-}IDi}$ and $P_{y\text{-}IDi}$ to the MSP and keeps the sd_i and $d_{s\text{-}IDi}$ as secret.

- **Authentication Phase**

In this phase, the mobile device *i*, the target sensor node *h*, the gateway *k* and the MSP will be mutually authenticated by the proposed authentication scheme. In our study, there are two kinds of communication. The first one is the mobile device carried by man collects data from the sensor node and the second kind is two sensor nodes exchange information without human intervention. The details of the two kinds of authentication process are described respectively.

(1)Mobile device collects data from sensor node

The process is shown as Figure11.

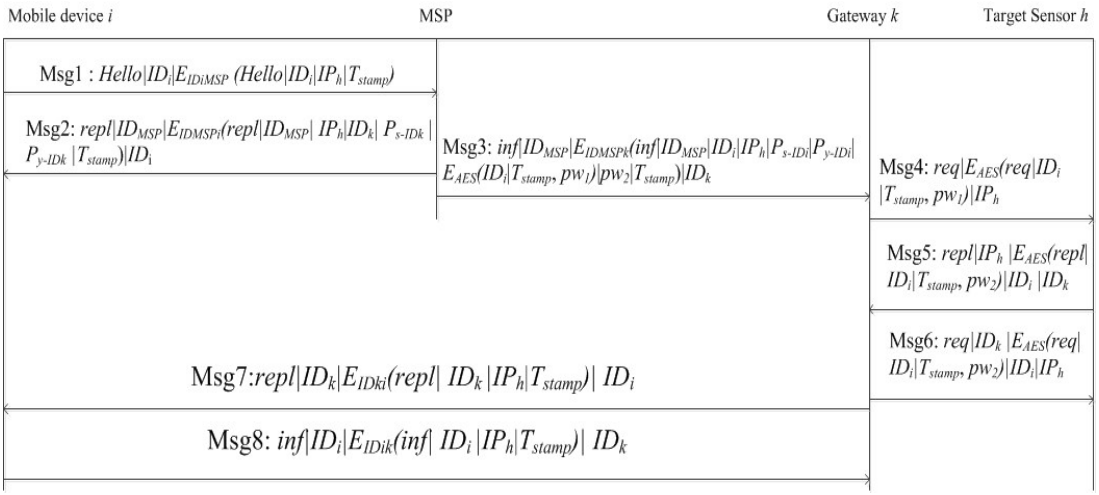

Figure11. The Process1 of Authentication Phase

- Msg1 : *Hello*|ID_i|E_{IDiMSP} *(Hello*|ID_i|IP_h|T_{stamp}*)*

Msg1 is the message to initialize a session for the data collection purpose. When the mobile device *i* wants to build a new connection with sensor node *h* to collect information, it will contact with the MSP for authentication first. Utilizing the sd_i, Q_{IDi}, d_{IDi}, $P_{s\text{-}MSP}$ and $P_{y\text{-}MSP}$, mobile device *i* encrypts its ID_i, the ID of the target sensor node IP_h and T_{stamp}. Then, the mobile device *i* will send the encrypted message as the Msg1 to the MSP.

- Msg2: $repl|ID_{MSP}|E_{IDMSPi}(repl|ID_{MSP}|\ IP_h|ID_k|\ P_{s\text{-}IDk}\ |\ P_{y\text{-}IDk}\ |T_{stamp})|ID_i$

When the MSP receives the Msg1, it will decrypt Msg1 with $d_{s\text{-}MSP}$, sd_{MSP} and $P_{s\text{-}IDi}$ according to ID_i. Then the MSP gets the ID_i, IP_j and T_{stamp} encrypted in Msg1. The MSP will firstly verify the value of T_{stamp}. If T_{stamp} indicates the Msg1 is invalid, the MSP will ignore the Msg1. Otherwise the MSP continues to verify the ID_i and checks whether the encrypted ID_i is the same as the ID_i which is transmitted as plaintext. If ID_i is invalid or the two ID_i is different, it means the Msg1 is forged or tampered and should be discarded. Otherwise the MSP continues to locate the target gateway ID-ID_k according to the target sensor ID-IP_h. Next, the MSP encrypts ID_{MSP}, IP_h, ID_k, $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$ and T_{stamp} with sd_{MSP}, Q_{MSP}, d_{MSP}, $P_{s\text{-}IDi}$ and $P_{y\text{-}IDi}$. Finally, the MSP sends the encrypted message as Msg2 to mobile device i. After the mobile device i receives the Msg2, it decrypts the Msg2 with $d_{s\text{-}IDi}$, sd_i and $P_{s\text{-}MSP}$ and gets the ID_k, $P_{s\text{-}IDk}$ and $P_{y\text{-}IDk}$.

- Msg3: $inf|ID_{MSP}|E_{IDMSPk}(inf|ID_{MSP}|ID_i|IP_h|P_{s\text{-}IDi}|P_{y\text{-}IDi}|E_{AES}(ID_i|T_{stamp}, pw_1)|pw_2|T_{stamp})|ID_k$

After the MSP sends Msg2 to mobile device i, it computes two symmetric keys for ID_k and IP_h: $pw_1 = [H_6\ (IP_h|RL_h[1]|RL_h[2]|\ldots|RL_h[n])]^{128}$ and $pw_2=[H_6\ (ID_i|IP_h|RL_h[1]|RL_h[2]|\ldots|RL_h[n])]^{128}$. Then the MSP encrypts $ID_i|T_{stamp}$ with pw_1 and AES function. Then the MSP combines the $E_{AES}(ID_i|T_{stamp}, pw_1)$ with ID_{MSP}, ID_i, IP_h, $P_{s\text{-}IDi}$, $P_{y\text{-}IDi}$, pw_2 and T_{stamp} and encrypts the combined message with sd_{MSP}, Q_{MSP}, d_{MSP}, $P_{s\text{-}IDk}$ and $P_{y\text{-}IDk}$. Then the MSP sends the encrypted message as Msg3 to the target gateway k to inform the gateway that the mobile device i wishes to build a connection with the sensor node h in its domain.

- Msg4: $req|E_{AES}(req|ID_i|T_{stamp}, pw_1)|IP_h$

After the target gateway ID_k receives the Msg3, it will decrypt the message with sd_k, $d_{s\text{-}IDk}$ and $P_{s\text{-}MSP}$ and get the ID_{MSP}, ID_i, IP_h, $P_{s\text{-}IDi}$, $P_{y\text{-}IDi}$, $E_{AES}(ID_i|T_{stamp}, pw_1)$, pw_2 and T_{stamp}. The gateway k will firstly verify the value of T_{stamp}. If T_{stamp} indicates the Msg3 is invalid, the gateway k will ignore the Msg3. Otherwise the gateway k continues to check whether the encrypted ID_{MSP} is the same as the ID_{MSP} which is transmitted as plaintext. If the two ID_{MSP} is different, it means the Msg3 is forged or tampered and should be discarded. Otherwise the gateway k sends the $E_{AES}(ID_i|T_{stamp}, pw_1)$ as Msg4 to the target sensor h.

- Msg5: $repl|IP_h\ |E_{AES}(repl|\ ID_i|T_{stamp}, pw_2)|ID_i\ |ID_k$

After the target sensor receives the Msg4, it will compute the pw_1 based on its ID and relative node list and decrypt the message to get ID_i. After verifying the value of T_{stamp}, the target sensor continues to compute the pw_2 based on the ID_i, IP_h and relative node list. Lastly, target sensor encrypts $ID_i|T_{stamp}$ with pw_2 and AES function and sends it to the gateway k to indicate that it's ready to transmit data.

- Msg6: $req|ID_k\ |E_{AES}(req|\ ID_i|T_{stamp}, pw_2)|ID_i|IP_h$

After the gateway k receives the Msg5, it will decrypt it with the pw_2 corresponding to ID_i and IP_h and checks the T_{stamp} and ID_i. Then it will encrypt the $ID_i|T_{stamp}$ with pw_2 and AES function and send the encrypted message to target sensor h to indicate the connection between them is constructed successfully.

- Msg7:$repl|ID_k|E_{IDki}(repl|\ ID_k\ |IP_h|T_{stamp})|\ ID_i$

After sending the Msg6, the gateway k will encrypt the $ID_k|IP_h|T_{stamp}$ with the sd_k, Q_{IDk}, d_{IDk}, $P_{s\text{-}IDi}$ and $P_{y\text{-}IDi}$. Finally, the gateway k sends the encrypted message as Msg7 to the mobile device i.

- Msg8: $inf|ID_i|E_{IDik}(inf|\ ID_i\ |IP_h|T_{stamp})|\ ID_k$

After receiving the Msg7, the mobile device i decrypts it with sd_i, $d_{s\text{-}IDi}$ and $P_{s\text{-}IDk}$. After checking the T_{stamp} and ID_k, the mobile device i will encrypt the $ID_i|IP_h|T_{stamp}$ with the sd_i, Q_{IDi}, d_{IDi}, $P_{s\text{-}IDk}$ and $P_{y\text{-}IDk}$. Lastly, the mobile device i sends the encrypted message as Msg8 to the target gateway k to indicate the session is constructed successfully.

By now, the mobile device i has built a secure session with the sensor node h through utilizing sd_i, Q_{IDi}, d_{IDi}, $d_{s\text{-}IDi}$, $P_{s\text{-}IDi}$, $P_{y\text{-}IDi}$, sd_k, Q_{IDk}, d_{IDk}, $d_{s\text{-}IDk}$, $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$ and pw_2. The information collected by sensor node h could be delivered to the mobile device i with high security.

(2) Two sensor nodes exchange information

The process is shown as Figure12.

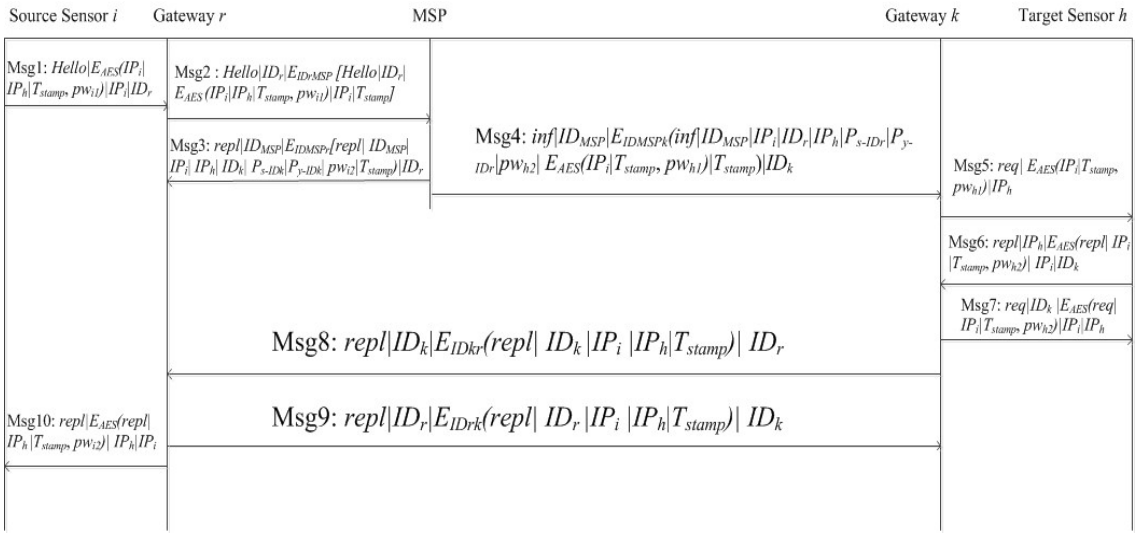

Figure12. The Process2 of Authentication Phase

- Msg1: $Hello|E_{AES}(IP_i|IP_h|T_{stamp}, pw_{i1})|IP_i|ID_r$

Msg1 is the message to initialize a session. When the source sensor node *i* wants to build a new connection with the target sensor node *h* to exchange information, it will compute its symmetric key

$$pw_{i1} = [H_6(IP_i|RL_i[1]|RL_i[2]|\ldots|RL_i[n])]^{128} \text{ and}$$

$$pw_{i2} = [H_6(IP_i|IP_h|RL_i[1]|RL_i[2]|\ldots|RL_i[n])]^{128}.$$

Then the source sensor *i* encrypts the $(IP_i|IP_h|T_{stamp})$ with pw_{i1} and AES function and sends the encrypted message to its gateway ID_r.

- Msg2 : $Hello|ID_r|E_{IDrMSP}[Hello|ID_r|E_{AES}(IP_i|IP_h|T_{stamp}, pw_{i1})|IP_i|T_{stamp}]$

After the source gateway ID_r receives the Msg1, it will utilize the sd_r, Q_{IDr}, d_{IDr}, $P_{s\text{-}MSP}$ and $P_{y\text{-}MSP}$ to encrypt the $ID_r|E_{AES}(IP_i|IP_h|T_{stamp}, pw_{i1})|IP_i|T_{stamp}$. Then, the gateway *r* will send the encrypted message as the Msg2 to the MSP.

- Msg3: $repl|ID_{MSP}|E_{IDMSPr}[repl|ID_{MSP}|IP_i|IP_h|ID_k|P_{s\text{-}IDk}|P_{y\text{-}IDk}|pw_{i2}|T_{stamp})|ID_r$

When the MSP receives the Msg2, it will decrypt Msg2 with sd_{MSP}, $d_{s\text{-}MSP}$ and $P_{s\text{-}IDr}$ according to ID_r. Then the MSP gets the ID_r, $E_{AES}(IP_i|IP_h|T_{stamp}, pw_{i1})$, IP_i and T_{stamp} encrypted in Msg2. The MSP will firstly verify the value of T_{stamp}. If T_{stamp} indicates the Msg2 is invalid, the MSP will ignore the Msg2. Otherwise the MSP continues to verify the ID_r and check whether the encrypted ID_r is the same as the ID_r which is transmitted as plaintext. If ID_r is invalid or the two ID_r is different, it means the Msg2 is forged or tampered and should be discarded. Otherwise the MSP go on to verify the IP_i, compute the

corresponding symmetric key pw_{i1} and decrypt the $E_{AES}(IP_i|IP_h|T_{stamp}, pw_{i1})$ to get the IP_h. After comparing the two IP_i, the MSP continues to compute the symmetric key pw_{i2} and locate the target gateway ID-ID_k according to the target sensor ID-IP_h. Next, the MSP encrypts ID_{MSP}, IP_i, IP_h, ID_k, $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$, pw_{i2} and T_{stamp} with sd_{MSP}, Q_{MSP}, d_{MSP}, $P_{s\text{-}IDr}$ and $P_{y\text{-}IDr}$. Finally, the MSP sends the encrypted message as Msg3 to source gateway r. After the gateway r receives the Msg3, it decrypts the Msg3 with sd_r, $d_{s\text{-}IDr}$ and $P_{s\text{-}MSP}$ and gets the IP_i, IP_h, pw_{i2}, ID_k, $P_{s\text{-}IDk}$ and $P_{y\text{-}IDk}$.

- Msg4: $inf|ID_{MSP}|E_{IDMSPk}(inf|ID_{MSP}|IP_i|ID_r|IP_h|P_{s\text{-}IDr}|P_{y\text{-}IDr}|pw_{h2}|E_{AES}(IP_i|T_{stamp}, pw_{h1})|T_{stamp})|ID_k$

After the MSP sends Msg3 to gateway r, it computes two symmetric keys for ID_k and IP_h:

$$pw_{h1}=[H_6(IP_h|RL_h[1]|RL_h[2]|\ldots|RL_h[n])]^{128} \text{ and}$$

$$pw_{h2}=[H_6(IP_i|IP_h|RL_h[1]|RL_h[2]|\ldots|RL_h[n])]^{128}.$$

Then the MSP encrypts $IP_i|T_{stamp}$ with pw_{h1} and AES function. Then the MSP combines the $E_{AES}(IP_i|T_{stamp}, pw_{h1})$ with ID_{MSP}, IP_i, ID_r, IP_h, $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$, pw_{h2} and T_{stamp} and encrypts the combined message with sd_{MSP}, Q_{MSP}, d_{MSP}, $P_{s\text{-}IDk}$ and $P_{y\text{-}IDk}$. Then the MSP sends the encrypted message as Msg4 to the target gateway k to inform the gateway that the sensor i in the domain of gateway r wishes to build a connection with the sensor node h in its domain.

- Msg5: $req|E_{AES}(IP_i|T_{stamp}, pw_{h1})|IP_h$

After the target gateway k receives the Msg4, it will decrypt the message with sd_k, $d_{s\text{-}IDk}$ and $P_{s\text{-}MSP}$ and get the ID_{MSP}, IP_i, ID_r, IP_h, $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$, pw_{h2}, $E_{AES}(IP_i|T_{stamp}, pw_{h1})$ and T_{stamp}. The gateway k will firstly verify the value of T_{stamp}. If T_{stamp} indicates the Msg4 is invalid, the gateway k will ignore the Msg4. Otherwise the gateway k continues to check whether the encrypted ID_{MSP} is the same as the ID_{MSP} which is transmitted as plaintext. If the two ID_{MSP} is different, it means the Msg4 is forged or tampered and should be discarded. Otherwise the gateway k sends the $E_{AES}(IP_i|T_{stamp}, pw_{h1})$ as Msg5 to the target sensor h.

- Msg6: $repl|IP_h|E_{AES}(repl|\ IP_i\ |T_{stamp}, pw_{h2})|\ IP_i|ID_k$

After the target sensor receives the Msg5, it will compute the pw_{h1} based on its ID and relative node list and decrypt the message to get IP_i. After verifying the value of T_{stamp}, the target sensor continues to compute the pw_{h2} based on the IP_i, IP_h and relative node list. Lastly, target sensor encrypts $IP_i|T_{stamp}$ with pw_{h2} and AES function and sends it to the gateway k to indicate that it's ready to transmit data.

- Msg7: $req|ID_k\ |E_{AES}(req|\ IP_i|T_{stamp}, pw_{h2})|IP_i|IP_h$

After the gateway k receives the Msg6, it will decrypt it with the pw_{h2} corresponding to IP_i and IP_h and checks the T_{stamp} and IP_i. Then it will encrypt the $IP_i|T_{stamp}$ with pw_{h2} and AES function and send the encrypted message to target sensor h to indicate the connection between them is constructed successfully.

- Msg8: $repl|ID_k|E_{IDkr}(repl|\ ID_k\ |IP_i\ |IP_h|T_{stamp})|\ ID_r$

After sending the Msg7, the gateway k will encrypt the $ID_k|IP_i|IP_h|T_{stamp}$ with the sd_k, Q_{IDk}, d_{IDk}, $P_{s\text{-}IDr}$ and $P_{y\text{-}IDr}$. Finally, the gateway k sends the encrypted message as Msg8 to the gateway r.

- Msg9: $repl|ID_r|E_{IDrk}(repl|\ ID_r\ |IP_i\ |IP_h|T_{stamp})|\ ID_k$

After receiving the Msg8, the gateway r decrypts it with sd_r, $d_{s\text{-}IDr}$ and $P_{s\text{-}IDk}$. After checking the T_{stamp} and ID_k, the gateway r will encrypt the $ID_r|IP_i|IP_h|T_{stamp}$ with the sd_r, Q_{IDr}, d_{IDr}, $P_{s\text{-}IDk}$ and $P_{y\text{-}IDk}$. Finally, the gateway r sends the encrypted message as Msg9 to the gateway k to indicate the connection between them is built successfully.

- Msg10: $repl|E_{AES}(repl|\ IP_h\ |T_{stamp}, pw_{i2})|\ IP_h|IP_i$

After sending the Msg9, the gateway r will encrypt the $IP_h|T_{stamp}$ with AES function and the pw_{i2} corresponding to the IP_i and IP_h. Lastly, the gateway r sends the encrypted message as Msg10 to the source sensor i to indicate the session is constructed successfully.

By now, the source sensor i has built a secure session with the target sensor h through utilizing sd_r, Q_{IDr}, d_{IDr}, $d_{s\text{-}IDr}$, $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$, sd_k, Q_{IDk}, d_{IDk}, $d_{s\text{-}IDk}$, $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$, pw_{i2} and pw_{h2}. The information could be exchanged between sensor node i and sensor node h with high security.

- **Key Updating Phase**

The phase is for the practical applications which have the need to update keys to prevent the system from the password guessing attack and reduce the threat of key leakage. In the phase, the public key Q_{ID} and the partial private key d_{ID} stay the same and the $P_{s\text{-}ID}$, $P_{y\text{-}ID}$ and $d_{s\text{-}ID}$ are updated. The updating process is: the entity which wants to update the key

chooses a new random number *sd'* and computes the new $P_{s\text{-}ID}'$, $P_{y\text{-}ID}'$ and $d_{s\text{-}ID}'$. Then the entity encrypts the new $P_{s\text{-}ID}'$ and $P_{y\text{-}ID}'$ with *sd'*, Q_{ID}, d_{ID}, $P_{s\text{-}MSP}$ and $P_{y\text{-}MSP}$ and sends the encrypted message to the MSP. After the MSP receives and decrypts the message, it replaces the $P_{s\text{-}ID}$ and $P_{y\text{-}ID}$ with the new $P_{s\text{-}ID}'$ and $P_{y\text{-}ID}'$ in its database. Then the key updating process finishes.

4.7 Security Analysis

In this section, we will analyze the security function of our authentication scheme. Firstly, we analyze the security design of the authenticated certificateless encryption scheme. Secondly, we analyze the security from the perspective of mutual authentication. Thirdly, we analyze the scheme's ability of standing against various attacks. Finally, we give the protocol analysis with BAN Logic.

1. The Security Design of the Authenticated Certificateless Encryption Scheme

The design of the authenticated certificateless encryption scheme makes the scheme gain two characteristics: 1) the message is authenticated when it is encrypted; 2) the scheme is without key escrow problem.

1) When the mobile device or source gateway ID_i communicates with the target gateway ID_j, the ID_i needs to use sd_i to encrypt the message and the ID_j needs to use $P_{s\text{-}IDi}$ to decrypt the message. Only the correct (sd_i, $P_{s\text{-}IDi}$) pair could ensure the message is encrypted and decrypted correctly. That means only if the message is encrypted by legitimate ID_i, the ID_j could decrypt it by corresponding $P_{s\text{-}IDi}$. So the message is authenticated with the encryption and no more signatures are needed.

2) When a receiver ID_j wants to decrypt a message, it needs to use sd_j and $d_{s\text{-}IDj}$. The sd_j and the $d_{s\text{-}IDj}$ are only known to the receiver. So even the MSP is compromised or the partial private key d_{IDj} is leaked, the message could still only be decrypted by the receiver due to the sd_j and $d_{s\text{-}IDj}$. So the existence of sd_j and $d_{s\text{-}IDj}$ solves the key escrow problem. What's more, the updating of sd_j and $d_{s\text{-}IDj}$ improves the security of the certificateless scheme.

2. Mutual Authentication

A mutual authentication among mobile devices, gateways, sensor nodes and the MSP can be accomplished with the authenticated certificateless encryption scheme and symmetric key generation mechanism. The ID of mobile device or gateway is verified by the MSP via the certificateless mechanism. Only the message encrypted by legitimate mobile devices or gateways could be decrypted by the MSP with the corresponding $P_{s\text{-}ID}$. Moreover, the application of the sub-private key sd_{ID} ensures that only the legitimate mobile device or gateway could make the message authenticated with encryption and only the target gateway could decrypt that message and vice versa.

Besides, only legitimate sensor nodes and the MSP could compute the correct symmetric key according to the symmetric key generation mechanism and only verified gateway could be distributed the corresponding symmetric key in the authentication process. What's more, the symmetric keys for different sensors and different sessions are different with each other. So the authenticated certificateless encryption scheme and symmetric key generation mechanism guarantee the mutual authentication among the MSP, mobile devices, gateways and sensor nodes.

3. Ability against Multiple Outside Attacks

- **Against Man-in-the-middle Attacks**

Man-in-the-Middle attack means an attacker between the sender and receiver receives a message from the sender, tampers the message, inserts or steals the information in the message and forwards the message to the receiver without its attention.

In our scheme, in inter-domain link, although an attacker could obtain the transmitted message to get the ID of sender and receiver, it has no knowledge of the corresponding sd_{ID} and $d_{s\text{-}ID}$. So the attacker could not decrypt the message and insert any illegal information or steal private information in the connection. On the other hand, even the attacker changes the ID which are transmitted as plaintext in a message, the altered message will be ignored since the encrypted message is authenticated. In sensor domain, the symmetric key is various in different sensors and different sessions, so it's infeasible for an attacker to guess the correct key. And if an attacker changes the *IP* which is transmitted as plaintext in a message, the message will be received by a wrong sensor which has no ability to decrypt it correctly. Then the changed message will be dropped immediately.

- **Against Reply Attacks**

Reply attack means an attacker gets a message and replies it to the receiver to pretend that the legitimate sender sends the message again.

However, in our scheme, timestamp T_{stamp} has been also encrypted in a message. Each time a receiver decrypts a message, it will check the value of T_{stamp}. If T_{stamp} indicates the message is invalid, the message will be ignored. Since the attacker is not able to decrypt a

message, the value of T_{stamp} cannot be changed. So even an attacker ceaselessly replies the messages, the replied messages will all be invalid in a certain time as the T_{stamp} expires.

- **Against DoS Attacks**

DoS attack means an attacker could send a "hello" message containing a certain device's ID to the MSP continually. After receiving many messages which contain the same ID, the MSP will reject to serve for the ID since it considers the corresponding device has been compromised. The "hello" message could be a legitimate message transmitted previously or a forged message.

In our system, if an attacker sends a previously legitimate message to the MSP, the message will be ignored due to the value of T_{stamp}. If an attacker forges a "hello" message, the message will also be rejected since the attacker does not have the correct *sd* and the forged message could not be decrypted correctly. So the attacker could never launch a DoS attack.

- **Against Impersonation Attacks**

Impersonation attack means an attacker disguises as a legal entity to communicate with the MSP or legitimate entities.

In our authentication scheme, to impersonate a mobile device or gateway needs the corresponding sub-secret key *sd*. The *sd* is chosen by a legitimate entity randomly, only stored in the entity and never show up in any messages. To impersonate a sensor node needs the IP and corresponding *RL[n]*. The *RL[n]* also never show up in the messages. So it's

very hard for an attacker to impersonate the mobile device, gateway or sensor node since it doesn't have the access to the corresponding *sd* and *RL[n]*.

- **Against Compromised Attacks**

Compromised attack means the legitimate entity or the MSP is compromised and the partial private key d_{ID}, sub-private key $d_{s\text{-}ID}$ or *RL[n]* are leaked. Then the attacker could utilize the d_{ID}, $d_{s\text{-}ID}$ or *RL[n]* to damage the system.

In our scheme, if the MSP is compromised, the attacker could not threat the system since the entities' *sd*s are not stored in the MSP. The attacker still could not decrypt the transmitted messages or impersonate a legitimate entity in the inter-domain link. If a mobile device, gateway or sensor node is compromised, the attacker could only impersonate the compromised entity and decrypt the messages sent to the compromised entity. The d_{ID}, $d_{s\text{-}ID}$ or *RL[n]* of uncompromised entity are still secret to the attacker and the majority of the system keeps safe. So the damages caused by the compromised attack are quite limited.

- **Against Insider Attack**

Insider attack means an attacker would capture a sensor node which has limited resource and usually is deployed in an unmanned situation to change it into an insider of the network. Through manipulating the insider, the attacker could threat the system.

In our study, the ability of an insider are classified into three levels: (1) the insider obtains every parameter distributed by the MSP but has no knowledge of how to utilize them, (2) the insider obtains every parameter distributed by the MSP and knows how to compute a symmetric key while knowing little about the communication policy, (3) the

insider obtains every parameter distributed by the MSP and knows the communication policy.

We first consider the impact of insider to other legitimate sensor nodes. In our mechanism, each sensor node has a unique relative node list *RL[n]*. Obviously, the insider could only compute its own symmetric key since it doesn't have other sensors' *RL[n]* no matter which level it is. So in our mechanism, the insider is not able to eavesdrop or modify other sensors' communication data. Then we only need to consider the impact and countermeasure of the situation in which the insider is the source or target node.

For the first level insider, it only has parameters' value and does not understand how to use them. This kind of insiders could not communicate with others normally as they have no symmetric key. For the second level insiders who could compute a symmetric key, they could cause only a little damage. That is because in the authentication mechanism, there are two symmetric keys to be used in one session and sometimes there is update of relative list. The insider may not be sure to use which key and could not update the list correctly. That would make it fail to build a connection. For the third level insider which is the strongest attacker, it is hard to be distinguished from the other legitimate sensors since the knowledge of parameters and policy helps the insider operates like a legitimate node. To handle the third level insider, an additional scheme could be added: In system initialization phase, the MSP distributes a unique constant value *cons* to each sensor node. When the MSP wants to check whether sensor node IP_i is the insider, it could order the corresponding gateway ID_r to disguise that another sensor wants to set up a session with node IP_i. After the session is

built, the MSP could request the *cons* value of sensor IP_i through ID_r. Then if sensor IP_i is a positive insider, there is a high possibility that sensor IP_i will send a false *cons* value. Once the MSP affirms the sensor IP_i modifies the *cons* value, the sensor will be regarded as an insider and isolated from the system. If sensor IP_i is a passive insider which only collects environmental data for the outside intruder, the method to identify it would be the detection of communication flows. If its outward communication rate is recognized as higher than average and sometimes with unknown destination, the sensor could be considered as an insider. Moreover, to get rid of the threats of the insider, an efficient detection scheme which is not the focus of this work could also be utilized. The effective detection scheme is easy to find since lots of detection schemes for wireless sensor network have been proposed in the literature. With a proper detection scheme, the system could find out the insiders and isolate them from the legitimate group.

4. Protocol Analysis with BAN Logic

BAN logic is the logic of belief for defining and analyzing communication protocols. The goal of using BAN Logic is to analyze authentication protocols by deriving the beliefs that the correct execution of protocol by legitimate principals could generate a trustworthy result. There are four steps in the protocol analysis using BAN Logic.

1. Idealize the protocol.

2. Write initial state assumptions.

3. Protocol annotation.

4. Beliefs derivation with logic.

The notations being used are shown as follows:

D: mobile device *i*. *TS*: target sensor node. *TG*: target gateway. *SS*: source sensor node. *SG*: source gateway. *M*: MSP. *PK(P,k)*: *k* is a public key of *P*. *PRK(P,k)*: *k* is a private key of *P*. $P \overset{k}{\leftrightarrow} Q$: *k* is the shared key between *P* and *Q*.

(1) Mobile device collects data from sensor node

- **Idealization**

Message2: $M \rightarrow D$: $\{ PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk}), fresh(P_{s\text{-}IDk}, P_{y\text{-}IDk}), n_T \}_{KMD}$ from *M*

Message3: $M \rightarrow TG$: $\{ PK(D, P_{s\text{-}IDi}, P_{y\text{-}IDi}), fresh(P_{s\text{-}IDi}, P_{y\text{-}IDi}), \{\{TG \overset{pw2}{\longleftrightarrow} TS\}, fresh(pw_2), n_T\}_{Kpw1}, \{TG \overset{pw2}{\longleftrightarrow} TS\}, fresh(pw_2), n_T \}_{KMTG}$ from *M*

Message4: $TG \rightarrow TS$: $\{\{TG \overset{pw2}{\longleftrightarrow} TS\}, fresh\ (pw_2), n_T \}_{Kpw1}$ from *M*

In idealization form, we regard the T_{stamp} as nonce n_T. We also regard the Q_{ID}, d_{ID}, $P_{s\text{-}ID}$, $P_{y\text{-}ID}$ and $d_{s\text{-}ID}$ which are believed by two entities as the shared key *K* between them. Note that Q_{ID}, d_{ID} and $d_{s\text{-}ID}$ are already trusted since the initialization phase, so making the $P_{s\text{-}ID}$ and $P_{y\text{-}ID}$ trusted by each other is the main concern. Since the Msg1, Msg5, Msg6, Msg7 and Msg8 are without any public key or private key, we omit the Msg1, Msg5, Msg6, Msg7 and Msg8.

- **Initial State Assumption**

P1. D believes $D \overset{KMD}{\longleftrightarrow} M$

P2. TG believes $TG \overset{KMTG}{\longleftrightarrow} M$

P3. TS believes $TS \overset{pw1}{\longleftrightarrow} M$

P4. D believes *M* controls *PK(TG, k)*

P5. TG believes *M* controls *PK(D, k)*

P6. TG believes *M* controls $TG \overset{pw2}{\longleftrightarrow} TS$

P7. TS believes *M* controls $TG \overset{pw2}{\longleftrightarrow} TS$

P8. D believes *M* controls *fresh(PK(TG, k))*

P9. TG believes *M* controls *fresh(PK(D, k))*

P10. TG believes *M* controls *fresh(* $TG \overset{pw2}{\longleftrightarrow} TS$*)*

P11. TS believes *M* controls *fresh(* $TG \overset{pw2}{\longleftrightarrow} TS$*)*

P12. D believes $fresh(n_T)$

P13. TG believes $fresh(n_T)$

P14. TS *believes* $fresh(n_T)$

- **Protocol Annotation**

The annotation states assumptions based on the idealization protocol. It is shown below:

P15. *D* received $\{ PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk}), fresh(P_{s\text{-}IDk}, P_{y\text{-}IDk}), n_T \}_{KMD}$ from *M*

P16. *TG* received $\{ PK(D, P_{s\text{-}IDi}, P_{y\text{-}IDi}), fresh(P_{s\text{-}IDi}, P_{y\text{-}IDi}), \{\{TG \overset{pw2}{\longleftrightarrow} TS\}, fresh(pw_2), n_T\}_{Kpw1}, \{TG \overset{pw2}{\longleftrightarrow} TS\}, fresh(pw_2), n_T \}_{KMTG}$ from *M*

P17. *TS* received $\{\{TG \overset{pw2}{\longleftrightarrow} TS\}, fresh\ (pw_2), n_T \}_{Kpw1}$ from *M*

- **Beliefs derivation**

In the derivations below, every line is followed by the rule by which it was derived.

15. $D \models M \sim \{ PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk}), \#(P_{s\text{-}IDk}, P_{y\text{-}IDk}), n_T \}$:

D believes *M* said $\{PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk}), fresh(P_{s\text{-}IDk}, P_{y\text{-}IDk}), n_T \}$. By Message Meaning using P1, P15.

16. $D \models \#\{ PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk}), \#(P_{s\text{-}IDk}, P_{y\text{-}IDk}), n_T \}$:

D believes *fresh* {*PK(TG,* $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$*)*, *fresh*($P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$), n_T}. By Freshness Conjuncatenation using 1, P12.

17. $D \models M \models \{ PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk}), \#(P_{s\text{-}IDk}, P_{y\text{-}IDk}), n_T \}$:

D believes *M* believes { *PK(TG,* $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$*)*, *fresh*($P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$), n_T }. By Nonce Verification using 2, 1.

18. $D \models M \models PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk})$:

D believes *M* believes *PK(TG,* $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$*)*. By Belief Conjuncatenation using 3.

19. $D \models M \models \# PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk})$:

D believes *M* believes *(fresh PK(TG,* $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$*))*. By Belief Conjuncatenation using 3.

20. $D \models PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk})$:

D believes *PK(TG,* $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$*)*. By Jurisdiction using 4, P4.

21. $D \models \# PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk})$:

D believes *(fresh PK(TG,* $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$*))*. By Jurisdiction using 5, P8.

We have derived mobile device *i*'s belief in the goodness and freshness of $P_{s\text{-}IDk}$ and $P_{y\text{-}IDk}$. We now turn to target gateway *k*.

22. $TG \models M \mid\sim \{PK(D, P_{s\text{-}IDi}, P_{y\text{-}IDi}), \#(P_{s\text{-}IDi}, P_{y\text{-}IDi}), \{\{TG \overset{pw2}{\longleftrightarrow} TS\}, \#(pw_2), n_T\}_{Kpw1}, \{TG \overset{pw2}{\longleftrightarrow} TS\}, \#(pw_2), n_T\}$:

TG believes *M* said {*PK(D,* $P_{s\text{-}IDi}$, $P_{y\text{-}IDi}$*)*, *fresh*($P_{s\text{-}IDi}$, $P_{y\text{-}IDi}$), $\{\{TG \overset{pw2}{\longleftrightarrow} TS\}$, *fresh*($pw_2$), $n_T\}_{Kpw1}$, $\{TG \overset{pw2}{\longleftrightarrow} TS\}$, *fresh*($pw_2$), n_T }. By Message Meaning using P2, P16.

23. $TG \models \#\{PK(D, P_{s\text{-}IDi}, P_{y\text{-}IDi}), \#(P_{s\text{-}IDi}, P_{y\text{-}IDi}), \{\{TG \overset{pw2}{\longleftrightarrow} TS\}, \#(pw_2), n_T\}_{Kpw1}, \{TG \overset{pw2}{\longleftrightarrow} TS\}, \#(pw_2), n_T\}$:

TG believes *fresh* $\{PK(D, P_{s\text{-}IDi}, P_{y\text{-}IDi}), fresh(P_{s\text{-}IDi}, P_{y\text{-}IDi}), \{\{TG \overset{pw2}{\longleftrightarrow} TS\}, fresh(pw_2), n_T\}_{Kpw1}, \{TG \overset{pw2}{\longleftrightarrow} TS\}, fresh(pw_2), n_T\}$. By Freshness Conjuncatenation using 8, P13.

24. $TG \models M \models \{PK(D, P_{s\text{-}IDi}, P_{y\text{-}IDi}), \#(P_{s\text{-}IDi}, P_{y\text{-}IDi}), \{\{TG \overset{pw2}{\longleftrightarrow} TS\}, \#(pw_2), n_T\}_{Kpw1}, \{TG \overset{pw2}{\longleftrightarrow} TS\}, \#(pw_2), n_T\}$:

TG believes *M* believes $\{PK(D, P_{s\text{-}IDi}, P_{y\text{-}IDi}), fresh(P_{s\text{-}IDi}, P_{y\text{-}IDi}), \{\{TG \overset{pw2}{\longleftrightarrow} TS\}, fresh(pw_2), n_T\}_{Kpw1}, \{TG \overset{pw2}{\longleftrightarrow} TS\}, fresh(pw_2), n_T\}$. By Nonce Verification using 9, 8.

25. $TG \models M \models (TG \overset{pw2}{\longleftrightarrow} TS, PK(D, P_{s\text{-}IDi}, P_{y\text{-}IDi}))$:

TG believes *M* believes $(TG \overset{pw2}{\longleftrightarrow} TS, PK(D, P_{s\text{-}IDi}, P_{y\text{-}IDi}))$. By Belief Conjuncatenation using 10.

26. $TG \models M \models \#(TG \overset{pw2}{\longleftrightarrow} TS, PK(D, P_{s\text{-}IDi}, P_{y\text{-}IDi}))$:

TG believes *M* believes $(fresh (TG \overset{pw2}{\longleftrightarrow} TS, PK(D, P_{s\text{-}IDi}, P_{y\text{-}IDi})))$. By Belief Conjuncatenation using 10.

27. $TG \models (TG \overset{pw2}{\longleftrightarrow} TS, PK(D, P_{s\text{-}IDi}, P_{y\text{-}IDi}))$:

TG believes $(TG \overset{pw2}{\longleftrightarrow} TS, PK(D, P_{s\text{-}IDi}, P_{y\text{-}IDi}))$. By Jurisdiction using 11, P6, P5.

28. $TG \models \#(TG \overset{pw2}{\longleftrightarrow} TS, PK(D, P_{s\text{-}IDi}, P_{y\text{-}IDi}))$:

TG believes $(fresh (TG \overset{pw2}{\longleftrightarrow} TS, PK(D, P_{s\text{-}IDi}, P_{y\text{-}IDi})))$. By Jurisdiction using 12, P10, P9.

We have derived target gateway's belief in the goodness and freshness of $P_{s\text{-}IDi}$, $P_{y\text{-}IDi}$ and pw_2. We now turn to target sensor *h*.

29. $TS \models M \mid\sim \{\{TG \overset{pw2}{\longleftrightarrow} TS\}, \#(pw_2), n_T\}$:

TS believes *M* said $\{\{TG \overset{pw2}{\longleftrightarrow} TS\}, fresh(pw_2), n_T\}$. By Message Meaning using P3, P17.

30. $TS \models \#\{\{TG \overset{pw2}{\longleftrightarrow} TS\}, \#(pw_2), n_T\}$:

TS believes $fresh\{\{TG \overset{pw2}{\longleftrightarrow} TS\}, fresh(pw_2), n_T\}$. By Freshness Conjuncatenation using 15, P14.

31. $TS \models M \models \{\{TG \overset{pw2}{\longleftrightarrow} TS\}, \#(pw_2), n_T\}$:

TS believes *M* believes $\{\{TG \overset{pw2}{\longleftrightarrow} TS\}, fresh(pw_2), n_T\}$. By Nonce Verification using 16, 15.

32. $TS \models M \models \{TG \overset{pw2}{\longleftrightarrow} TS\}$:

TS believes *M* believes $\{TG \overset{pw2}{\longleftrightarrow} TS\}$. By Belief Conjuncatenation using 17.

33. $TS \models M \models \#(pw_2)$:

TS believes *M* believes $fresh\ (pw_2)$. By Belief Conjuncatenation using 17.

34. $TS \models \{TG \overset{pw2}{\longleftrightarrow} TS\}$:

TS believes $\{TG \overset{pw2}{\longleftrightarrow} TS\}$. By Jurisdiction using 18, P7.

35. $TS \models \#(pw_2)$:

TS believes $fresh(pw_2)$. By Jurisdiction using 19, P11.

By now, we have already derived the mobile device's belief in $P_{s\text{-}IDk}$ and $P_{y\text{-}IDk}$ the target gateway's belief in $P_{s\text{-}IDi}$, $P_{y\text{-}IDi}$ and pw_2 and the target sensor node's belief in pw_2. With the beliefs' in Q_{ID}, d_{ID}, $P_{s\text{-}ID}$, $P_{y\text{-}ID}$, $d_{s\text{-}ID}$ and symmetric keys, the mobile device, target gateway and target sensor node could set up a session securely. The protocol analysis with BAN Logic indicates the correctness of protocol design.

(2) Two sensor nodes exchange information

- **Idealization**

Message3: $M \rightarrow SG$: { $PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk})$, $fresh(P_{s\text{-}IDk}, P_{y\text{-}IDk})$, $SG \overset{pwi2}{\longleftrightarrow} SS$, $fresh(pw_{i2})$, n_T }$_{KMSG}$ from M

Message4: $M \rightarrow TG$: { $PK(SG, P_{s\text{-}IDr}, P_{y\text{-}IDr})$, $fresh(P_{s\text{-}IDr}, P_{y\text{-}IDr})$, {$\{TG \overset{pwh2}{\longleftrightarrow} TS\}$, $fresh(pw_{h2})$, n_T}$_{Kpwh1}$, $\{TG \overset{pwh2}{\longleftrightarrow} TS\}$, $fresh(pw_{h2})$, n_T }$_{KMTG}$ from M

Message5: $TG \rightarrow TS$: {$\{TG \overset{pwh2}{\longleftrightarrow} TS\}$, $fresh$ (pw_{h2}), n_T }$_{Kpwh1}$ from M

In idealization form, we regard the T_{stamp} as nonce n_T. Similar as the situation (1), making the $P_{s\text{-}ID}$ and $P_{y\text{-}ID}$ of source gateway and target gateway trusted by each other is the main concern. Note that since the source sensor node could compute the symmetric key pw_2 with its own ID and target sensor ID, it already has the beliefs in the goodness and freshness of pw_{i2}. Since the Msg1, Msg2, Msg6, Msg7, Msg8, Msg9 and Msg10 are without any public key or private key, we omit the Msg1, Msg2, Msg6, Msg7, Msg8, Msg9 and Msg10.

- **Initial State Assumption**

P1. SG believes $SG \overset{KMSG}{\longleftrightarrow} M$

P2. TG believes $TG \overset{KMTG}{\longleftrightarrow} M$

P3. TS believes $TS \overset{pwh1}{\longleftrightarrow} M$

P4. SG believes *M* controls *PK(TG, k)*

P5. SG believes *M* controls $SG \overset{pwi2}{\longleftrightarrow} SS$

P6. TG believes *M* controls *PK(SG, k)*

P7. TG believes *M* controls $TG \overset{pwh2}{\longleftrightarrow} TS$

P8. TS believes *M* controls $TG \overset{pwh2}{\longleftrightarrow} TS$

P9. SG believes *M* controls *fresh(PK(TG, k))*

P10. SG believes *M* controls *fresh(* $SG \overset{pwi2}{\longleftrightarrow} SS$)

P11. TG believes *M* controls *fresh(PK(SG, k))*

P12. TG believes *M* controls *fresh(* $TG \overset{pwh2}{\longleftrightarrow} TS$)

P13. TS believes *M* controls *fresh(* $TG \overset{pwh2}{\longleftrightarrow} TS$)

P14. SG believes *fresh*(n_T)

P15. TG believes *fresh*(n_T)

P16. TS *believes fresh*(n_T)

- **Protocol Annotation**

The annotation states assumptions based on the idealization protocol. It is shown below:

P17. *SG* received { *PK(TG,* $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$), *fresh*($P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$), $SG \overset{pwi2}{\longleftrightarrow} SS$, *fresh*($pw_{i2}$), n_T }$_{\text{KMSG}}$ from *M*

P18. *TG* received { *PK(SG,* $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$), *fresh*($P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$), {{ $TG \overset{pwh2}{\longleftrightarrow} TS$ }, *fresh*(pw_{h2}), n_T}$_{\text{Kpwh1}}$, {$TG \overset{pwh2}{\longleftrightarrow} TS$}, *fresh*($pw_{h2}$), n_T }$_{\text{KMTG}}$ from *M*

P19. *TS* received {{$TG \overset{pwh2}{\longleftrightarrow} TS$}, *fresh* ($pw_{h2}$), n_T }$_{\text{Kpwh1}}$ from *M*

- **Beliefs derivation**

In the derivations below, every line is followed by the rule by which it was derived.

1. $SG \models M \mid\sim$ { *PK(TG,* $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$), #($P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$), $SG \overset{pwi2}{\longleftrightarrow} SS$, #($pw_{i2}$), n_T }:

SG believes *M* said { *PK(TG,* $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$), *fresh*($P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$), $SG \overset{pwi2}{\longleftrightarrow} SS$, *fresh*($pw_{i2}$), n_T }. By Message Meaning using P1, P17.

2. $SG \models$ #{ *PK(TG,* $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$), #($P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$), $SG \overset{pwi2}{\longleftrightarrow} SS$, #($pw_{i2}$), n_T }:

SG believes *fresh* { $PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk})$, $fresh(P_{s\text{-}IDk}, P_{y\text{-}IDk})$, $SG \overset{pwi2}{\longleftrightarrow} SS$, $fresh(pw_{i2})$, n_T }. By Freshness Conjuncatenation using 1, P14.

3. $SG \models M \models \{ PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk}), \#(P_{s\text{-}IDk}, P_{y\text{-}IDk}), SG \overset{pwi2}{\longleftrightarrow} SS, \#(pw_{i2}), n_T \}$:

   SG believes *M* believes { $PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk})$, $fresh(P_{s\text{-}IDk}, P_{y\text{-}IDk})$, $SG \overset{pwi2}{\longleftrightarrow} SS$, $fresh(pw_{i2})$, n_T }. By Nonce Verification using 2, 1.

4. $SG \models M \models (SG \overset{pwi2}{\longleftrightarrow} SS, PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk}))$:

   SG believes *M* believes $(SG \overset{pwi2}{\longleftrightarrow} SS, PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk}))$. By Belief Conjuncatenation using 3.

5. $SG \models M \models \#(SG \overset{pwi2}{\longleftrightarrow} SS, PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk}))$:

   SG believes *M* believes $(fresh (SG \overset{pwi2}{\longleftrightarrow} SS, PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk})))$. By Belief Conjuncatenation using 3.

6. $SG \models (SG \overset{pwi2}{\longleftrightarrow} SS, PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk}))$:

   SG believes $(SG \overset{pwi2}{\longleftrightarrow} SS, PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk}))$. By Jurisdiction using 4, P5, P4.

7. $SG \models \#(SG \overset{pwi2}{\longleftrightarrow} SS, PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk}))$:

   SG believes $(fresh (SG \overset{pwi2}{\longleftrightarrow} SS, PK(TG, P_{s\text{-}IDk}, P_{y\text{-}IDk})))$. By Jurisdiction using 5, P10, P9.

We have derived the source gateway's belief in the goodness and freshness of $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$ and pw_{i2}. We now turn to target gateway *k*.

8. $TG \models M \mid\sim \{PK(SG, P_{s\text{-}IDr}, P_{y\text{-}IDr}), \#(P_{s\text{-}IDr}, P_{y\text{-}IDr}), \{\{TG \overset{pwh2}{\longleftrightarrow} TS\}, \#(pw_{h2}), n_T\}_{Kpwh1}$, $\{TG \overset{pwh2}{\longleftrightarrow} TS\}, \#(pw_{h2}), n_T \}$:

TG believes *M* said {*PK(SG, $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$), fresh*($P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$), {{ $TG \overset{pwh2}{\longleftrightarrow} TS$ }, *fresh*(pw_{h2}), n_T}$_{\text{Kpwh1}}$, {$TG \overset{pwh2}{\longleftrightarrow} TS$}, *fresh*($pw_{h2}$), n_T }. By Message Meaning using P2, P18.

9. *TG* ⊨ #{*PK(SG, $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$), #*($P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$), {{ $TG \overset{pwh2}{\longleftrightarrow} TS$ }, #(pw_{h2}), n_T}$_{\text{Kpwh1}}$, {$TG \overset{pwh2}{\longleftrightarrow} TS$}, #($pw_{h2}$), n_T }:

   TG believes *fresh* {*PK(SG, $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$), fresh*($P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$), {{ $TG \overset{pwh2}{\longleftrightarrow} TS$ }, *fresh*(pw_{h2}), n_T}$_{\text{Kpwh1}}$, {$TG \overset{pwh2}{\longleftrightarrow} TS$}, *fresh*($pw_{h2}$), n_T }. By Freshness Conjuncatenation using 8, P15.

10. *TG* ⊨ *M* ⊨ {*PK(SG, $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$), #*($P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$), {{$TG \overset{pwh2}{\longleftrightarrow} TS$}, #($pw_{h2}$), n_T}$_{\text{Kpwh1}}$, {$TG \overset{pwh2}{\longleftrightarrow} TS$}, #($pw_{h2}$), n_T }:

    TG believes *M* believes {*PK(SG, $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$), fresh*($P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$), {{$TG \overset{pwh2}{\longleftrightarrow} TS$}, *fresh*($pw_{h2}$), n_T}$_{\text{Kpwh1}}$, {$TG \overset{pwh2}{\longleftrightarrow} TS$}, *fresh*($pw_{h2}$), n_T }. By Nonce Verification using 9, 8.

11. *TG* ⊨ *M* ⊨ *($TG \overset{pwh2}{\longleftrightarrow} TS$, PK(SG, $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$))*:

    TG believes *M* believes *($TG \overset{pwh2}{\longleftrightarrow} TS$, PK(SG, $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$))*. By Belief Conjuncatenation using 10.

12. *TG* ⊨ *M* ⊨ # *($TG \overset{pwh2}{\longleftrightarrow} TS$, PK(SG, $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$))*:

    TG believes *M* believes *(fresh ($TG \overset{pwh2}{\longleftrightarrow} TS$, PK(SG, $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$)))*. By Belief Conjuncatenation using 10.

13. *TG* ⊨ *($TG \overset{pwh2}{\longleftrightarrow} TS$, PK(SG, $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$))*:

    TG believes *($TG \overset{pwh2}{\longleftrightarrow} TS$, PK(SG, $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$))*. By Jurisdiction using 11, P7, P6.

14. *TG* ⊨ #*($TG \overset{pwh2}{\longleftrightarrow} TS$, PK(SG, $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$))*:

TG believes *(fresh* $(TG \overset{pwh2}{\longleftrightarrow} TS$, *PK(SG,* $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$*)))*. By Jurisdiction using 12, P12, P11.

We have derived target gateway's belief in the goodness and freshness of $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$ and pw_{h2}. We now turn to target sensor *h*.

15. $TS \models M \mid\sim \{\{TG \overset{pwh2}{\longleftrightarrow} TS\}, \#(pw_{h2}), n_T\}$:

    TS believes *M* said $\{\{TG \overset{pwh2}{\longleftrightarrow} TS\}, fresh(pw_{h2}), n_T\}$. By Message Meaning using P3, P19.

16. $TS \models \#\{\{TG \overset{pwh2}{\longleftrightarrow} TS\}, \#(pw_{h2}), n_T\}$:

    TS believes $fresh\{\{TG \overset{pwh2}{\longleftrightarrow} TS\}, fresh(pw_{h2}), n_T\}$. By Freshness Conjuncatenation using 15, P16.

17. $TS \models M \models \{\{TG \overset{pwh2}{\longleftrightarrow} TS\}, \#(pw_{h2}), n_T\}$:

    TS believes *M* believes $\{\{TG \overset{pwh2}{\longleftrightarrow} TS\}, fresh(pw_{h2}), n_T\}$. By Nonce Verification using 16, 15.

18. $TS \models M \models \{TG \overset{pwh2}{\longleftrightarrow} TS\}$:

    TS believes *M* believes $\{TG \overset{pwh2}{\longleftrightarrow} TS\}$. By Belief Conjuncatenation using 17.

19. $TS \models M \models \#(pw_{h2})$:

    TS believes *M* believes $fresh\ (pw_{h2})$. By Belief Conjuncatenation using 17.

20. $TS \models \{TG \overset{pwh2}{\longleftrightarrow} TS\}$:

    TS believes $\{TG \overset{pwh2}{\longleftrightarrow} TS\}$. By Jurisdiction using 18, P8.

21. $TS \models \#(pw_{h2})$:

    TS believes $fresh(pw_{h2})$. By Jurisdiction using 19, P13.

By now, we have already derived the source gateway's belief in $P_{s\text{-}IDk}$, $P_{y\text{-}IDk}$ and pw_{i2}, the target gateway's belief in $P_{s\text{-}IDr}$, $P_{y\text{-}IDr}$ and pw_{h2} and the target sensor node's belief in pw_{h2}. With the beliefs' in Q_{ID}, d_{ID}, $P_{s\text{-}ID}$, $P_{y\text{-}ID}$, $d_{s\text{-}ID}$ and symmetric keys, the gateways and sensor nodes could set up a session securely. The protocol analysis with BAN Logic indicates the correctness of protocol design.

4.8 Efficiency Analysis

We evaluate the computation cost and storage cost of our scheme. The notations are shown in Table VI.

TABLE VI. DEFINITION OF NOTATION

C_{H1}	Computation cost of hash function H_1
C_{H2}	Computation cost of hash function H_2
C_{H3}	Computation cost of hash function H_3
C_{H4}	Computation cost of hash function H_4
C_{H5}	Computation cost of hash function H_5
C_{H6}	Computation cost of hash function H_6
C_p	Computation cost of pairing operation e
C_e	Computation cost of exponentiation in G
C_m	Computation cost of scalar or point multiplic ation in G
C_o	Computation cost of bitwise exclusive-or
C_{aes}	Computation cost of AES function

According to the authenticated certificateless encryption scheme, the computation cost of encryption is $C_{H2}+C_{H3}+C_{H4}+C_{H5}+C_p+C_e+2C_m+2C_o$, the computation cost of decryption is $C_{H2}+C_{H3}+C_{H4}+C_{H5}+C_p+2C_m+2C_o$. According to the authentication process, the total computation cost is shown as follow:

(1) Mobile device collects data from sensor node

- For mobile devices, there are two times of certificateless encryption and two times of certificateless decryption. So the total computation cost is $4C_{H2}+4C_{H3}+4C_{H4}+4C_{H5}+4C_p+2C_e+8C_m+8C_o$.
- For the MSP, there are two times of certificateless encryption, one time of certificateless decryption and one time of AES encryption. Note that the MSP also need to compute the Q_{IDi} and Q_{IDk}. So the total computation cost is $2C_{H1}+3C_{H2}+3C_{H3}+3C_{H4}+3C_{H5}+3C_p+2C_e+6C_m+6C_o+2\ C_{H6}+C_{aes}$.
- For sensor nodes, there is one time of AES encryption, two times of AES decryption and two times of computing symmetric key, so the total computation cost is $3C_{aes}+2C_{H6}$.
- For gateways, there is one time of certificateless encryption, two times of certificateless decryption, one time of AES encryption and one time of AES decryption. So the total computation cost is $3C_{H2}+3C_{H3}+3C_{H4}+3C_{H5}+3C_p+C_e+6C_m+6C_o+2C_{aes}$.

(2) Two sensor nodes exchange information

Note that we calculate the cost of source side and target side together in this scenario. So the total cost is about two times of the cost in the previous scenario.

- For the gateways, there are three times of certificateless encryption, four times of certificateless decryption, two times of AES encryption and one time of AES decryption. So the total computation cost is $7C_{H2}+7C_{H3}+7C_{H4}+7C_{H5}+7C_p+3C_e+14C_m+14C_o+3C_{aes}$.

- For sensor nodes, there are four times of computing symmetric keys, two times of AES encryption and three times of AES decryption. So the total computation cost is $4C_{H6}+5C_{aes}$.
- For the MSP, there two times of certificateless encryption, one time of certificateless decryption, one time of AES encryption, one time of AES decryption and four times of computing symmetric key. So the total computation cost is $2C_{H1}+3C_{H2}+3C_{H3}+3C_{H4}+3C_{H5}+3C_p+2C_e+6C_m+6C_o+4C_{H6}+2C_{aes}$.

It's clear that no matter for which scenario, the heavy computation cost has been distributed to mobile devices, gateways and MSP. The AES function applied in the domain of sensor nodes save the computation resource of sensor nodes significantly.

Since the sensor communication is involved in our mechanism, we also compare the computation cost of sensor domain with the cost of sensor in wireless sensor network [62] [63] [64] [65] [66]. The result is shown as Table VII.

TABLE VII. COMPARISON OF COMPUTATION COST IN DIFFERENT SCHEMES

Entity	[62]	[63]	[64]	[65]	[66]	Proposed Scheme
Sensor Node	C_{H6}	$2C_{H6}$	$2C_{H6}$	$2C_{aes}+C_{H6}$	$2C_{H6}$	$2C_{aes}+2C_{H6}$

When compared with the traditional wireless sensor network, the proposed scheme costs a little more computation resource. However, the proposed scheme provides higher level security since [62], [63], [64], [66] didn't apply encryption function to sensor nodes and the symmetric key used by sensor nodes in [65] is less than that of the proposed scheme.

From the above analysis, the strong certificateless mechanism guarantees the security of inter-domain link and the AES function provides an acceptable security for inner-domain communication in an efficient way. So the hybrid encryption mechanism equipped in our scheme achieves a good balance between the security and performance.

4.9 Summary

The proposed scheme can ensure a safe session among mobile devices and sensor nodes in different domains. The application of authenticated certificateless encryption scheme could reduce the threat of compromise attack to a great extent. The application of efficient AES could save the computation resource of sensor nodes. The proposed secret key management and symmetric key generation mechanism could not only generate and update secret key conveniently but also reduce the risk of key leakage. Our analysis with BAN Logic indicates that the mutual authentication and the ability of withstanding multiple attacks could be accomplished by the proposed solution.

Chapter 5. CONCLUSION AND FUTURE WORK

In the thesis, we propose three authentication schemes for M2M communication: a dynamic-encryption authentication scheme for M2M security in CPS, an authentication scheme with IBC for M2M security in CPS and an authentication scheme for multi-domain M2M security in CPS. In the dynamic-encryption authentication scheme, the group of complex encryption algorithms provides high security between the mobile device and the MSP. The dynamic key generation mechanism utilizes an initial key space and a seed to generate a one-time-password with low costs. The level of security can be adjusted by changing the frequency of initial key materials updating. A lightweight encryption algorithm has been employed to save the resource of sensor nodes. In the authentication scheme with IBC, the application of integrated IBC scheme could achieve the message authentication without key escrow problem and reduce the threat of compromise attack to a great extent. The regular updating of secret key could also make the key guessing attack meaningless. In the third scheme, the hybrid encryption scheme which integrates complex certificateless scheme and efficient AES algorithm could provide high level security and save the sensor nodes' resource at the same time and the communication scenario in which the sensor nodes from different domains communicate with each other without human intervention has also been considered. Our analysis indicates that the mutual authentication and the ability of withstanding multiple attacks could be accomplished by the proposed solutions and the balance between system performance and security has been achieved.

The work done in the thesis has only focused on the authentication issues in the M2M domain which is an important part of the entire M2M communication system. In the future, more efforts should be made to improve the security and performance of the M2M s communication systems in terms of two aspects.

The first aspect should still be the security issues in M2M domain. As for the authentication in M2M domain, we could adopt the group authentication architecture to improve the proposed authentication schemes. Since the amount of M2M devices would be very large in the future, applying the group authentication architecture could significantly reduce the communication overhead and energy consumption of systems. Besides the authentication issue, a security framework which fulfills more security requirements, such as access control and privacy preserving, for M2M domain should be constructed. The framework should include the detection ability and recovering ability. The detection ability could help the systems to detect the compromised entities and isolate them to guarantee the security. The recovering ability could enable the M2M devices recovering from malicious attacks autonomously which is significant to the M2M systems because most of the M2M devices are deployed unattended.

The second aspect should be the security issues in the network domain and the application domain. The emerging technologies 6LoWPAN and CoAP correspond to the network domain and application domain of M2M system, respectively. The 6LoWPAN manages to assign each M2M device a unique identity by introducing an adaptation layer between IPv6 layer and IEEE802.15.4 MAC layer. However, the new adaptation layer may bring some potential

security threats, so future research should be carried out to identify those threats and come up with corresponding solutions. In addition, the CoAP is designed to enable the resource limited devices could be accessed through Internet remotely and will allow people to trace and control M2M devices through web applications. Since CoAP is constructed based on a subset of HTTP functionalities with considering the resource constraints of M2M devices, it will not only inherit the security weaknesses of the HTTP, but also possess some new vulnerabilities due to the consideration of resource limitation. So the security of CoAP is worth to pay attention to in the future.

BIBLIOGRAPHY

1. L. Sha et al., "Cyber-Physical Systems: A New Frontier," Proceedings of IEEE International Conference on Sensor Networks, Ubiquitous and Trustworthy Computing (SUTC), Taichung, Taiwan, China, June 2008, pp. 1-9.

2. F. Michahelles, S. Karpischek and A. Schmidt, "What Can the Internet of Things Do for the Citizen?" IEEE Pervasive Computing, Vol. 9, No.4, 2010, pp. 102-104.

3. A.A. Cardenas, S. Amin and S. Sastry, "Secure Control: Towards Survivable Cyber-Physical Systems," Proceedings of IEEE 28th International Conference on Distributed Computing Systems Workshops (ICDCS), Beijing, China, June 2008,pp. 495–500.

4. Z. Yuchen, D. Weijun and W. Fubao, "Architecture and real-time characteristics analysis of the cyber-physical system," Proceedings of IEEE 3rd International Conference on Communication Software and Networks (ICCSN), Xi'an, Shaanxi, China, May 2011, pp. 317-320.

5. Min Chen, Jiafu Wan and Fang Li, "Machine-to-Machine Communications: Architectures, Standards, and Applications," KSII Transactions on Internet and Information Systems, Vol. 6, No. 2, February 2012, pp. 480-497.

6. S. Pandey et al., "Towards management of machine to machine networks," Proceedings of the 13th Asia-Pacific Network Operations and Management Symposium (APNOMS), Taipei, Taiwan, China, September 2011, pp. 1-7.

7. H. Chen, Z. Fu and D. Zhang, "Security and trust research in M2M system," Proceedings of IEEE International Conference on Vehicular Electronics and Safety (ICVES), Beijing, China, July 2011, pp. 286-290.

8. M. Saedy and V. Mojtahed, "Ad Hoc M2M communications and security based on 4G cellular system," Proceedings of the Wireless Telecommunications Symposium (WTS), New York, American, April 2011, pp. 1-5.

9. L. Rongxing et al., "GRS: The green, reliability, and security of emerging machine to machine communications," IEEE Communications Magazine, Vol. 49, No. 4, April 2011, pp. 28-35.

10. C. Dong et al., "A Novel Secure Architecture for the Internet of Things," Proceedings of the fifth International Conference on Genetic and Evolutionary Computing (ICGEC), Xiamen, Fujian, China, August-September 2011, pp. 311-314.

11. Z.M. Fadlullah et al., "An early warning system against malicious activities for smart grid communications," IEEE Network, Vol. 25, No. 5, September 2011, pp. 50-55.

12. Ioannis Broustis, Ganapathy S. Sundaram, and Harish Viswanathan, “Detecting and Preventing Machine-to-Machine Hijacking Attacks in Cellular Networks,” Bell Labs Technical Journal, Vol. 17, No. 1, June 2012, pp. 125-140.

13. Cagalaban, Giovanni, Seoksoo Kim, and Minho Kim, "A Mobile Device-Based Virtualization Technique for M2M Communication in Cloud Computing Security," Computer Applications for Security, Control and System Engineering, Vol. 339, 2012, pp. 160-167.

14. Tien-Dung Nguyen, Aymen Al-Saffar, and Eui-Nam Huh, “A Dynamic ID-based Authentication Scheme,” Proceedings of the Sixth International Conference on Networked Computing and Advanced Information Management (NCM), Seoul, South Korea, August 2010, pp. 248-253.

15. S. Agarwal, C. Peylo, R. Borgaonkar, and J.-P. Seifert, “Operator-based Over-the-air M2M Wireless Sensor Network Security,” Proceedings of the 14th International Conference on Intelligence in Next Generation Networks (ICIN), Berlin, Germany, October 2010, pp. 1-5.

16. Yingying He, Liquan Chen, and Lingling Wang, “An Improved Direct Anonymous Attestation Scheme for M2M Networks,” Procedia Engineering, Vol. 15, 2011, pp. 1481-1486.

17. A. Bartoli, J. Hernandez-Serrano, M. Soriano, M. Dohler, A. Kountouris, and D. Barthel, "Secure lossless aggregation over fading and shadowing channels for smart grid M2M networks," IEEE Transactions on Smart Grid, Vol. 2, No. 4, December 2011, pp. 844-864.

18. A. Bartoli, J. Hernandez-Serrano, M. Soriano, M. Dohler, A. Kountouris, and D. Barthel, "Optimizing energy-efficiency of PHY-Layer authentication in machine-to-machine networks," Proceedings of 2012 IEEE Globecom Workshops (GC Wkshps), Anaheim, CA, American, December 2012, pp. 1663-1668.

19. A. Bartoli, J. Hernandez-Serrano, O. Leon, A. Kountouris, and D. Barthel, “Energy-efficient physical layer packet authenticator for machine-to-machine networks,” Transactions on Emerging Telecommunications Technologies, Vol. 24, No. 4, June 2013, pp. 401-412.

20. Hyundong Lee and Mokdong Chung, “Context-Aware Security System for the Smart Phone-based M2M Service Environment,” KSII Transactions on Internet and Information Systems (TIIS), Vol. 6, No. 1, January 2012, pp. 64-83.

21. Liang Hu, Ling Chi, Hong-tu Li, Wei Yuan, Yuyu Sun, and Jian-feng Chu, “The Classic Security Application in M2M: the Authentication Scheme of Mobile Payment,” KSII Transactions on Internet and Information Systems (TIIS), Vol. 6, No. 1, January 2012, pp. 131-146.

22. Chengzhe Lai, Hui Li, Yueyu Zhang, and Jin Cao, “Security Issues on Machine to Machine Communications,” KSII Transactions on Internet and Information Systems (TIIS), Vol. 6, No. 2, February 2012, pp. 498-514.

23. X. Sun, S. Men, C. Zhao, and Z. Zhou, "A security authentication scheme in machine-to-machine home network service," Security Comm. Networks, May 2012, doi: 10.1002/sec.551.

24. Ye Yan, Yi Qian, and Rose Qingyang Hu, “A Secure and Efficient Scheme for Machine-to-Machine Communications in Smart Grid,” Proceedings of the 2012 IEEE International Conference on Communications (ICC), Ottawa, ON, June 2012, pp. 167-172.

25. Inshil Doh, Kijoon Chae, Jiyoung Lim, and Min Young Chung, “An Improved Security Approach based on Kerberos for M2M Open IPTV System,” Proceedings of the 15th International Conference on Network-Based Information Systems (NBiS), Melbourne, VIC, September 2012, pp. 754-759.

26. Wujun Zhang, Yueyu Zhang, Jie Chen, Hui Li, and Yumin Wang, “End-to-end Security Scheme for Machine Type Communication Based on Generic Authentication Architecture,” Proceedings of 2012 4th International Conference on Intelligent Networking and Collaborative Systems (INCoS), Bucharest, Romenia, September 2012, pp. 353-359.

27. Nenad Gligoric, Tomislav Dimcic, Dejan Drajic, and Nhon Chu, “Application-Layer Security Mechanism for M2M communication over SMS,” Proceedings of the 20th Telecommunications Forum (TELFOR), Belgrade, November 2012, pp. 5-8.

28. I. Broustis, G.S. Sundaram and H. Viswanathan, "Group Authentication: A New Paradigm for Emerging Applications," Bell Labs Technical Journal, Vol.17, No. 3, 2012, pp. 157-173.

29. Jin Cao, Maode Ma, and Hui Li, “A Group-based Authentication and Key Agreement for MTC in LTE Networks,” Proceedings of 2012 IEEE Global Communications Conference (GLOBECOM), Anaheim, CA, American, December 2012, pp. 1017-1022.

30. Wei Ren, Linchen Yu, Liangli Ma, and Yi Ren, “How to Authenticate a Device? Formal Authentication Models for M2M Communications Defending against Ghost Compromising Attack,” International Journal of Distributed Sensor Networks (IJDSN), Vol. 2012, November 2012, pp. 1-9.

31. J.-M. Kim, H.-Y. Jeong and B.-H. Hong, "A study of privacy problem solving using device and user authentication for M2M environments," Security and Communication Networks, 2013, doi: 10.1002/sec.695.

32. Y.Ben Saied, A. Olivereau, and D. Zeghlache, “Energy Efficiency in M2M Networks: A Cooperative Key Establishment System,” Proceedings of the 3rd International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), Budapest, Hungary, October 2011, pp. 1-8.

33. Yosra Ben Saied, Alexis Olivereau, and Maryline Laurent, “A Distributed Approach for Secure M2M Communications,” Proceedings of the 5th International Conference on New Technologies, Mobility and Security (NTMS), Istanbul, May 2012, pp. 1-7.

34. B. S. Adiga, P. Balamuralidhar, M. A. Rajan, Ravishankara Shastry, and V. L. Shivraj, "An identity based encryption using elliptic curve cryptography for secure M2M communication," Proceedings of the First International Conference on Security of Internet of Things, ACM, New York, USA, August 2012, pp. 68-74.

35. Xin Ma, and Wei Luo, “The analysis of 6LowPAN technology,” Proceedings of Pacific-Asia Workshop on Computational Intelligence and Industrial Application (PACIIA), Wuhan, Hubei, China, December 2008, pp. 963-966.

36. W. Colitti, K. Steenhaut, and N. De Caro, “Integrating Wireless Sensor Networks with the Web,” Proceedings of the workshop on Extending the Internet to Low power and Lossy Networks (IP+SN 2011), Chicago, IL, USA, April 2011, pp. 1-5.

37. S. Raza, S. Duquennoy, T. Chung, D. Yazar, T. Voigt, and U. Roedig, “Securing Communication in 6LoWPAN with Compressed IPsec,” Proceedings of International Conference on Distributed Computing in Sensor Systems and Workshops (DCOSS), Barcelona, Spain, June 2011, pp. 1-8.

38. Yuanyuan Zhou, ZhipingJia, Xianli Sun, Xin Li, and Lei Ju, “Design of Embedded Secure Gateway Based on 6LoWPAN,” Proceedings of IEEE 13th International Conference on Communication Technology (ICCT), Jinan, Shandong, China, September 2011, pp. 732-736.

39. Arjun Kumar and HoonJae Lee, “Performance Comparison of Identity Based Encryption and Identity Based Signature,” International Journal of Security and Its Applications, Vol.6, No. 6, July 2012, pp. 19-27.

40. Shushan Zhao, A. Aggarwal, R. Frost and Xiaole Bai, “A Survey of Applications of Identity-Based Cryptography in Mobile Ad-Hoc Networks,” IEEE Communications Surveys & Tutorials, Vol. 14, No.2, 2012, pp. 380-400.

41. A. Shamir, “Identity-based cryptosystems and signature schemes,” Advances in Cryptology – CRYPTO’84, LNCS., Vol. 196, 1984, pp. 47-53.

42. D. Boneh and M. Franklin, “Identity-based encryption from the Weil Pairing,” Advances in Cryptology - CRYPTO 2001, LNCS., Springer-Verlag, Vol. 2139, 2001, pp. 213-229.

43. C. Cocks, “An Identity Based Encryption Scheme Based on Quadratic Residues,” Proceedings of the 8th IMA International Conference on Cryptography and Coding, Cirencester, UK, December 2001, pp. 360-363.

44. Ben Lynn, “Authenticated Identity-Based Encryption,” IACR Cryptology ePrint Archive, Report 2002/072, 2002. http://eprint.iacr.org/2002/072.

45. Craig Gentry and Alice Silverberg, “Hierarchical ID-based cryptography,” Advances in cryptology—ASIACRYPT 2002, LNCS, Springer Berlin Heidelberg, Vol. 2501, 2002, pp. 548-566.

46. Yao Danfeng, et al., “ID-based encryption for complex hierarchies with applications to forward security and broadcast encryption,” Proceedings of the 11th ACM conference on Computer and communications security, ACM, New York, American, 2004, pp. 354-363.

47. R. Sakai and M. Kasahara, “ID based Cryptosystems with Pairing on Elliptic Curve,” IACR Cryptology ePrint Archive, Report 2003/054, 2003. http://eprint.iacr.org/2003/054.

48. M. Abid, et al., “Efficient identity-based authentication for IMS based services access,” Proceedings of the 7th International Conference on Advances in Mobile Computing and Multimedia, ACM, New York, American, 2009, pp. 260-266.

49. Li Hongwei, et al., “Identity-based authentication for cloud computing,” Cloud Computing, Springer Berlin Heidelberg, Vol. 5931, 2009, pp. 157-166.

50. Na Sang-Ho, et al., “Identity-based secure protocol scheme for wireless sensor network,” Proceedings of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human. ACM, New York, American, 2009, pp. 555-560.

51. Sun Jinyuan, et al., “An identity-based security system for user privacy in vehicular ad hoc networks,” IEEE Transactions on Parallel and Distributed Systems, Vol. 21, No. 9, 2010, pp. 1227-1239.

52. H. Nicanfar, et al., “Efficient Authentication and Key Management Mechanisms for Smart Grid Communications,” IEEE System Journal, Vol. PP, No. 99, pp. 1-12.

53. Li Fagen, et al., “Efficient Signcryption for Heterogeneous Systems,” IEEE Systems Journal, Vol. 7, No.3, September 2013, pp. 420-429.

54. Li Fagen, et al., “Practical Secure Communication for Integrating Wireless Sensor Networks Into the Internet of Things,” IEEE Sensors Journal, Vol. 13, No. 10, Octorber 2013, pp. 3677-3684.

55. Byoungcheon Lee, et al., “Secure key issuing in ID-based cryptography,” Proceedings of the second workshop on Australasian information security, Data Mining and Web Intelligence and Software Internationalisation, Australian Computer Society, Inc., Vol. 32, 2004, pp. 69-74.

56. Jin Wang, et al., “Protecting against key escrow and key exposure in identity-based cryptosystem,” Theory and Applications of Models of Computation, Springer Berlin Heidelberg, Vol. 4484, 2007, pp. 148-158.

57. Sherman S.M. Chow, “Removing escrow from identity-based encryption,” Public Key Cryptography–PKC 2009, Springer Berlin Heidelberg, Vol. 5443, 2009, pp. 256-276.

58. Yan Zhu, et al., “Efficient identity-based encryption without pairings and key escrow for mobile devices,” Wireless Algorithms, Systems, and Applications, Springer Berlin Heidelberg, Vol. 7992, 2013, pp. 42-53.

59. Young-Ran Lee and Hyang-Sook Lee, “An Authenticated Certificateless Public Key Encryption Scheme,” IACR Cryptology ePrint Archive, Report 2004/150, 2004. http://eprint.iacr.org/2004/150.

60. D. Boneh and M. Franklin, “Identify-based encryption from the Weil Pairing,” SIAM J. Computing, Vol. 32, No. 3, March 2003, pp. 586-615.

61. P. Barreto, H. Kim, B. Lynn, and M. Scott, “Efficient algorithms for pairing-based cryptosystems,” Advances in Cryptology-CRYPTO’02, Vol. 2442, 2002, pp. 354-369.

62. M.L.Das, “Two-Factor User Authentication in Wireless Sensor Networks,” IEEE Transactions on Wireless Communications, Vol. 8, No. 3, March 2009, pp. 1086-1090.

63. B. Vaidya, “Improved Two-factor User Authentication in Wireless Sensor Networks,” Proceedings of the IEEE 6th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob), Niagara Falls, ON, Canada, October 2010, pp. 600-606.

64. K.S. Arikumar and K. Thirumoorthy, “Improved User Authentication in Wireless Sensor Networks,” Proceedings of the 2011 International Conference on Emerging Trends in Electrical and Computer Technology (ICETECT), Tamil Nadu, India, March 2011, pp. 1010-1015.

65. P. Kumar, Sang-Gon Lee, and Hoon-Jae Lee, “A User Authentication for Healthcare Application using Wireless Medical Sensor Networks,” Proceedings of the IEEE 13th International Conference on High Performance Computing and Communications (HPCC), Banff, AB, Canada, Sepetember 2011, pp. 647-652.

66. Wen-Bin Hsieh and Jenq-Shiou Leu, “A Dynamic Identity User Authentication Scheme in Wireless Sensor Network,” Proceedings of the 9th International Wireless Communications and Mobile Computing Conference (IWCMC), Sardinia, Italy, July 2013, pp. 1132-1137.