Journal of Universal Computer Science, vol. 18, no. 9 (2012), 1112-1131
submitted: 1/12/11, accepted: 28/4/12, appeared: 1/5/12 © J.UCS

Directed Path Based Authentication Scheme for the Internet of Things

Huansheng Ning, Hong Liu
(School of Electronic and Information Engineering
Beihang University, Beijing, China
ninghuansheng@buaa.edu.cn; liuhongler@ee.buaa.edu.cn)

Qing Liu, Genlin Ji
(School of Computer Science and Technology
Nanjing Normal University, Nanjing, China
njnulq@163.com; glji@njnu.edu.cn)

Abstract: The Internet of Things (IoT) is emerging as an attractive paradigm, and several IoT models and related security issues have received widespread attentions. In this paper, we focus on an existing U2IoT architecture (i.e., Unit IoT and Ubiquitous IoT), and propose a directed path based authentication scheme (DPAS) to realize security protection for the U2IoT architecture. Particularly, the directed path descriptor is introduced for the secret key distribution and cross-network authentication, and the proof mapping is applied to establish tri-dimensional equivalence relations among diverse nodes for achieving mutual authentication. Moreover, security analysis shows that DPAS achieves data confidentiality and integrity, authentication, anonymity and forward security, and performance analysis indicates that DPAS with moderate communication overhead and computation load is suitable for the IoT applications.

Key Words: Internet of things, security, authentication protocol, directed path.
Category: C.2.2, K.6.5

1 Introduction

The Internet of Things (IoT) becomes an attractive paradigm, in which the cyber and physical objects are attached with social attributes and achieve interconnection. Along with the combination of Internet and general sensor techniques such as radio frequency identification (RFID), near field communication (NFC), and wireless sensor and actuator networks (WSAN), IoT itself is suffering from more severe security challenges [Atzori et al. 2010, Weber 2010]. Several related issues including system architecture, security and privacy, standardization, and human behavior are subsequently raised in IoT applications.

Heterogeneous network architecture brings potential threats from robust adversaries in the IoT. Besides system resources referring to sensor node, actuator, channel, storage, bandwidth, and energy, may induce additional security vulnerabilities. Previous studies on the IoT have been worked on the following areas: system model and methodology [Zhong et al. 2010, Ma et al. 2011, Zhang et al. 2012a], architectural principle [Kortuem et al. 2010], standard [Koshizuka and Sakamura 2010], quality of service (QoS) [Zhang et al. 2011b], and security protocol [Hancke et al. 2010]. Particularly, an architecture that comprises Unit IoT and Ubiquitous IoT (short for U2IoT architecture) has been proposed to present a new paradigm covering both the specific unit IoT and the ubiquitous local/industrial/national IoT [Ning and Wang 2011]. In the U2IoT architecture, conceptions of mankind neural system and social organization framework are introduced for the future IoT, and multiple sensors and communication techniques are involved in the wireless/wire communication channels. Thereinto, wireless sensor networks (WSN) and Internet compose a typical IoT network structure: 1) WSN realizes the real-time data detection, identification, and monitoring, and 2) Internet realizes that the collected data is transmitted and controlled by top management centers. It is significative to establish a node-to-node secure channel between remote sensors, and to realize authentication and hierarchical access control by the back-end servers. Due to the diverse channels, interfaces, and context environments of heterogeneous networks, more challenges should be considered to achieve such cross-network security protection.

Towards the IoT security, several open issues such as cryptographic algorithms, authentication protocol, access control, trust/privacy, and governance frameworks should be considered [Roman et al. 2011a]. Studies mainly focus on specific communication techniques (e.g., RFID) [Hancke et al. 2010], detailed cryptographic mechanisms (e.g., key management) [Roman et al. 2011b], and practical applications (e.g., supply chain management, multimedia traffic) [Xu 2011, Zhou and Chao 2011]. Meanwhile, security frameworks in traditional networks can also provide merits for the IoT. However, existing studies mainly provide security protection for relatively isolated networks, which may not be suitable for the U2IoT architecture that comprises not only the specific unit IoT but also the ubiquitous IoT (local/industrial/national IoT). Hence, it is significant to establish an authentication scheme to realize cross-network protection for U2IoT architecture. The purpose of the paper is to provide comprehensive safeguard for U2IoT architecture to realize interconnection between the front-end sensor networks and the back-end management and data centers.

In this paper, a directed path based authentication scheme (DPAS) to guarantee security protection in the U2IoT architecture. The main contributions are as follows. 1) Directed path descriptor is adopted for the layered U2IoT architecture, which makes realizes interconnection among the distributed sensor nodes and management centers. 2) Proof mapping is applied to establish tri-dimensional equivalence relations among a sensor node, its neighbor nodes, and its storage node, which assists the sensor node and storage node to establish mutual authentication. 3) Proof verification is performed to realize cross-network authentication without using the sensor node's prior knowledge.

The remainder of the paper is organized as follows. Section 2 overviews the related security issues in the IoT, and also introduces the U2IoT architecture. Section 3 presents the detailed scheme procedures, and security analysis and performance analysis are given in Section 4. Finally, Section 5 draws a conclusion.

2 Related Works

2.1 Security Issues in the Internet of Things

Roman *et al.* [Roman et al. 2011a] pointed out that the traditional lightweight cryptography, secure protocol, and privacy assurance are not strong enough for the IoT, and also recommended several approaches to cope with the old and new security threats. The approaches refer to data and privacy, identity management, trust and governance, fault tolerance, cryptography and protocols, identity and ownership, and privacy protection, which provide guidance for the IoT security and privacy researches. Zhou *et al.* [Zhou and Chao 2011] designed an efficient multimedia-aware traffic security architecture which is based on the classified traffic to facilitate various multimedia applications. Such general security architecture is established by considering the characteristics of multimedia traffic, and security service in the IoT.

Bandyopadhyay *et al.* [Bandyopadhyay and Sen 2011] mainly considered two major security and privacy issues (i.e., confidentiality of the business processes, and privacy of the human). Towards data confidentiality, several standard cryptographic technologies are recommended, and it brings new challenges to design more efficient encryption algorithms, authentication protocols, and access control mechanisms. Particularly, key management should be researched as a significant topic to deal with the ubiquitous things' security and privacy protection. Thereinto, the authors suggested that key distribution should be designed considering the restrictions of resource, power, and operational capability in specific communication channels (e.g., small-scale systems and ad-hoc networks). Towards privacy, solutions are more intractable due to the ignorance of the general people. Considering the heterogeneity and mobility of things in IoT, privacy-preservation is still in its infancy. A holistic privacy framework is needed to be established for the resource-limited applications. However, the traditional privacy protection methods are mainly supported by power-unrestricted devices, which may bring another problems for the heterogenous and cross-layer networks.

Hancke *et al.* [Hancke et al. 2010] introduced unique security challenges for user-oriented RFID applications in the IoT systems, and the major challenges including privacy, ownership, data integrity, application integrity and security standards should be enhanced to achieve universal security. Meanwhile, Yan *et al.* [Yan and Wen 2011] applied mobile RFID security protocol to guarantee the security in the IoT, and they introduced a trust-third-party (TTP) based key management protocol to construct a secure session key among the tag, reader and back-end database. The proposed protocol is appropriate for the mobile RFID-based IoT architecture.

Several security schemes have been proposed to safeguard the WSN-based IoT systems, which mainly include authentication protocol [Delgado-Mohatar et al. 2011], cross-network authentication [Roman et al. 2011b], key establishment [Tsai et al. 2009, Roman et al. 2011c], distribution security framework [Claycomb and Shin 2011], and topology maintenance protocol [Gabrielli et al. 2011]. Thereinto, Delgado-Mohatar *et al.* [Delgado-Mohatar et al. 2011] proposed a lightweight authentication scheme for WSN. The scheme comprises a key management protocol and an authentication protocol, in which simple symmetric cryptographic primitives are adopted to realize both security and efficiency. Particularly, it provides a strong resilience against node capturing, and also provides node-to-node mutual authentication. Roman *et al.* [Roman et al. 2011b] presented the concept of a transversal layer to contain different security mechanisms. In the scheme, all the entities can interact with the security mechanisms which achieves a holistic safeguard. Roman *et al.* [Roman et al. 2011c] also analyzed the applicability of existing public key cryptography and pre-shared key mechanisms for the sensor-based IoT, and focused on the link-layer oriented key management systems to provide high compatibility for the same grouped sensor nodes to realize that two remote entities can anonymously negotiate certain security credentials. Tsai *et al.* [Tsai et al. 2009] proposed a storage-bounded adversary model to deal with the attacking scenario in WSN, in which two key establishment schemes are established by neighbor sensor nodes to obtain the shared keys. Gabrielli *et al.* [Gabrielli et al. 2011] focused on the topology maintenance protocols (TMPs) to analyze the security vulnerabilities in sensor networks, and proposed a meta-protocol (Meta-TMP) to increase the robustness against major attacks.

Furthermore, authentication and authorization schemes have been considered to prevent the attackers' unauthorized and authorized accessing. Meanwhile, intrusion detection [Amin et al. 2009] is introduced to guarantee the IP-based ubiquitous sensor networks, and a scoring classifier based on the statistical process control technique is used to identify the attack-signatures.

2.2 U2IoT Architecture Overview and Network Model

The U2IoT architecture [Ning and Wang 2011] includes two subnetworks that Unit IoT and Ubiquitous IoT. Thereinto, Unit IoT is a man-like nervous (MLN) model, and refers to the basic unit providing solutions for special applications. Ubiquitous IoT resembles social organization framework (SOF) model, and includes the local IoT, industrial IoT, national IoT, and even global IoT. The Ubiquitous IoT is integration of multiple Unit IoTs with ubiquitous features. Towards Unit IoT, its architecture is inspired by man's nervous system with centralized

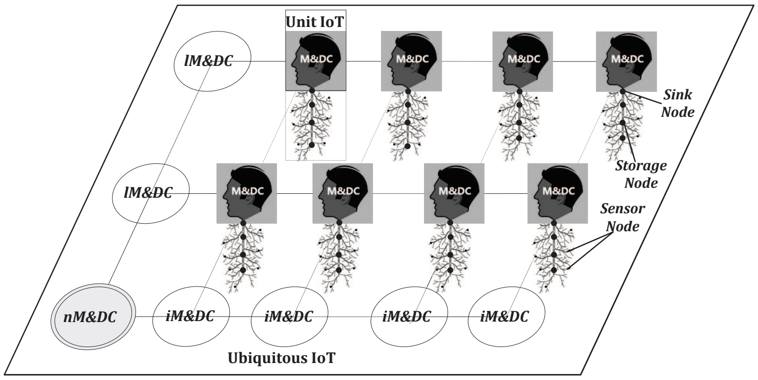

Figure 1: The U2IoT architecture (i.e., Unit IoT and Ubiquitous IoT).

or distributed data centers. Three main components are included: brain (management and centralized data center, M&DC), spinal cord (distributed control nodes), and a network of nerves (IoT networks and sensors). Towards Ubiquitous IoT, local IoT is based on the geographical region to realize interconnection among multiple Unit IoTs. Industrial IoT refers to different industries, and manages the corresponding national-wide Unit IoT's in specific industry, such as logistics, agriculture, power grid, transportation. National IoT contains above two aspects to realize interpenetration between local IoTs and Industrial IoTs.

In the U2IoT architecture, multiple wireless nodes are used to capture data streams, to detect activities and events with diversified identification approached, and to realize specific application functions. The U2IoT architecture refers to multiple sensor technologies (e.g., WSN, RFID, GPS, Zigbee, Bluetooth, WiFi, femtocell) and communication technologies (e.g., Internet, mobile telecommunications). In the paper, we focus on the WSN and Internet based the U2IoT architecture, and the re-organized model is shown in Figure 1. Note that Unit IoT comprises the front-end sensors and M&DC, and Ubiquitous IoT comprises the back-end management and data centers (lM&DC, iM&DC, nM&DC).

- *The front-end wireless sensor subnetwork* adopts the layered node structure, and three types of nodes are defined, including the regular sensor node (sensor node for short), storage node, and sink node. The sensor node performs real-time data acquisition, and gathers raw information and communicates with other neighbor nodes and its storage node via the wireless communication channels. The storage node connecting the sensor node and sink node is used to temporarily store sensing data, to complete the initial relation mapping, and to perform preliminary authentication. Meanwhile, it also buffers the received data and responds with the authorized data to the sink node.

The sink node acts as an intermediary to connect the sensor/ storage nodes and M&DC in Unit IoT.

– *The back-end management and data center subnetwork* includes M&DC of Unit IoT, and lM&DC, iM&DC, nM&DC of the local, industry and national IoTs in Ubiquitous IoT. The management and data centers are independent, and are interconnected via the virtual directed paths. The virtual directed path is a descriptor which indicates the possible transition between two entities. For instance, $Path_X^Y$ represents the virtual path from X to Y. The mentioned "virtual" does not mean that the entities $\{X, Y\}$ establish direct communication channel via $Path_X^Y$, but establish a virtual transition relationship. The directed path is applied to achieve cross-network authentication, and only the achievable paths can be used to realize the interconnection among the different entities.

3 The Proposed Authentication Scheme: DPAS

The proposed DPAS contains three phases: key distribution, initiation and proof mapping, and proof establishment and verification. The detailed notations are introduced in Table 1.

3.1 Key Distribution Phase

In DPAS, there are L sensor nodes in the sensor set S_A, the legal nodes are preloaded with an initial authenticator. Each sensor has at most $L-1$ neighbor nodes and at least one storage node. If there are two or more storage nodes, the sensor node will alternatively select one in random mode. In order to enhance data confidentiality and integrity, three types of secret keys are defined.

Master key: k_m is established based on end-to-end data encryption [Peering et al. 2002], which is periodically updated and considered as a secure secret.

Shared key: k_u is obtained based on Diffie-Hellman (DH) key agreement, and it is applied by MC_u, $\{MC_l, MC_i\}$, and MC_n to establish the corresponding shared keys $(k_{u_{l(i)}}, k_{u_{nl(i)}})$. Thereinto, the directed path descriptors are introduced for these centers to realize mutual authentication. We take k_{u_l} for example to introduce the shared key distribution.

MC_u randomly generates random numbers $\{r^0_{MC_u}, r'^0_{MC_u}\}$, and computes X_{MC_u} and Y_{MC_u} by its identifier and directed path descriptor $Path_{MC_u}^{MC_l}$. Thereafter, MC_u transmits $X_{MC_u} \| Y_{MC_u} \| r'^0_{MC_u} \| id_{MC_u}$ to MC_l.

$$X_{MC_u} = g^{r^0_{MC_u}} \pmod p \tag{1}$$

$$Y_{MC_u} = H(r'^0_{MC_u} \| id_{MC_u}) \oplus Path_{MC_u}^{MC_l} \tag{2}$$

Table 1: Notations

Notation	Description
MC_u	The management and data center of Unit IoT, i.e., (M&DC).
MC_l, MC_i, MC_n	The management and data centers of the local Unit IoT, industry Unit IoT, and national Unit IoT, i.e., (lM&DC, iM&DC, nM&DC).
S_A, S_N, S_B, S_C	The set of sensor nodes, neighbor nodes, storage nodes, and sink nodes.
s_a	The sensor node.
s_n, s_{B_a}, s_{C_a}	The neighbor nodes, storage node, and sink node of s_a.
id	The pseudo-identifier.
Tid	The token identifier.
$Path_X^Y$	The virtual directed path descriptor from X to Y. The path pair that $\{Path_X^Y, Path_Y^X\}$ satisfies the given rule.
k_m	The master key by MC_u and S_C.
$k_{u_{l(i)}}$	The shared key between MC_u and $MC_l(i)$.
$k_{u_{nl(i)}}$	The shared key between between $MC_l(i)$ and MC_n.
k_{auth}	The authentication key by S_A, S_B, S_C.
p, g, q	The large prime numbers. g is primitive root mod p.
$\{M\}_k$	The symmetric encryption algorithm on M by k.

MC_l randomly generates random numbers $\{r^0_{MC_l}, r'^0_{MC_l}\}$, and computes X_{MC_l} and Y_{MC_l}. Thereafter, MC_l transmits $X_{MC_l} \| Y_{MC_l} \| r'^0_{MC_l} \| id_{MC_l}$ to MC_u.

$$X_{MC_l} = g^{r^0_{MC_l}} \pmod p \tag{3}$$

$$Y_{MC_l} = H(r'^0_{MC_l} \| id_{MC_l}) \oplus Path^{MC_u}_{MC_l} \tag{4}$$

Upon receiving the messages, MC_u computes $H(r'^0_{MC_l} \| id_{MC_l})$ by the received $\{r'^0_{MC_l}, id_{MC_l}\}$, and derives $Path^{MC_u}_{MC_l}$ by the XOR operation. MC_u compares whether the derived $Path^{MC_u}_{MC_l}$ and its stored $Path^{MC_l}_{MC_u}$ satisfy the given rule according to an invertible operator I and an anti-homomorphism function $\varphi(AB) = \varphi(B)\varphi(A)$.

$$\varphi((Path^{MC_u}_{MC_l} \oplus I) Path^{MC_l}_{MC_u}) \stackrel{?}{=} \varphi(Path^{MC_l}_{MC_u}) \varphi(Path^{MC_u}_{MC_l} \oplus I) \tag{5}$$

Thereafter, MC_l performs verification on MC_u. If MC_u and MC_l pass the mutual verification, MC_u and MC_l will obtain the shared key k_{u_l},

$$k_{u_l} = (X_{MC_u})^{r^0_{MC_l}} = (X_{MC_l})^{r^0_{MC_u}} = g^{r^0_{MC_u} \cdot r^0_{MC_l}} \tag{6}$$

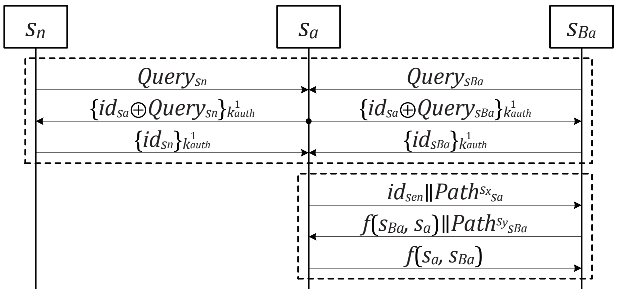

Figure 2: The exchanged messages of the initiation and proof mapping.

Similarly, the shared keys (k_{u_i}, $k_{u_{nl}}$, $k_{u_{ni}}$) are obtained according to the DH key agreement and mutual authentication mode.

Authentication key: k_{auth} is derived from the master key k_m, and it is used by S_A, S_B, S_C for authentication. Thereinto, a matrix $E_{L \times L}$ is defined for the authentication key generation, in which $e_{l,l}$ is its element for $l = [L^{1/2}] + 1$ ([.] denotes the rounding operation).

MC_u assigns k^1_{auth} as the authentication key in the first session, and assigns the updated authentication key k^x_{auth} for the x-th session, in which a large prime number q is used to determine the element's subscripts.

$$k^1_{auth} = k_m \oplus e_{l,l} \tag{7}$$

$$k^x_{auth} = H(k^{x-1}_{auth} \oplus e_{xq \pmod l, 2xq \pmod l}) \tag{8}$$

3.2 Initiation and Proof Mapping Phase

In the case that s_a is a newly joined sensor node that attempts to enter the network, it should establish the mapping between s_a and $\{s_n, s_{B_a}\}$. Figure 2 shows the exchanged messages during the initiation and proof mapping of DPAS. Note that in the case that s_a is not a fresh node, it skips the mapping process in system initiation.

3.2.1 Initiation of the sensor node, neighbor nodes, and storage node.

S_N, $S_B \Rightarrow s_a$: $Query_{s_N}$, $Query_{s_B}$.
$s_a \rightarrow s_n$: $\{id_{s_a} \oplus Query_{s_n}\}_{k^1_{auth}}$, $s_a \rightarrow S_{B_a}$: $\{id_{s_a} \oplus Query_{s_{B_a}}\}_{k^1_{auth}}$.
$s_n \rightarrow s_a$: $\{id_{s_n}\}_{k^1_{auth}}$, $s_{B_a} \rightarrow s_a$: $\{id_{s_{B_a}}\}_{k^1_{auth}}$.

1. S_N and S_B constantly broadcast the omni-bearing $\{Query_{s_N}, Query_{s_B}\}$. s_a initializes to determine its neighbor nodes s_n (n=1, ..., L_n), and ascertains its storage node s_{B_a}.

Table 2: Relation mapping between s_a and $\{s_n,\, s_{B_a}\}$

Nodes	Tri-dimensional relation mapping				
$s_a \to s_n$:	$Path_{s_a}^{s_n}$	$\Leftrightarrow$	$H(id_{s_a} \| id_{s_n})$	$\Leftrightarrow$	$F(Path_{s_a}^{s_*} Path_{s_*}^{s_n})$
$s_a \to s_{B_a}$:	$Path_{s_a}^{s_{B_a}}$	$\Leftrightarrow$	$H(id_{s_a} \| id_{s_{B_a}})$	$\Leftrightarrow$	$F(Path_{s_a}^{s_*} Path_{s_*}^{s_{B_a}})$
$s_n \to s_a$:	$Path_{s_n}^{s_a}$	$\Leftrightarrow$	$H(id_{s_n} \| id_{s_a})$	$\Leftrightarrow$	$F(Path_{s_n}^{s_*} Path_{s_*}^{s_a})$
$s_{B_a} \to s_a$:	$Path_{s_{B_a}}^{s_a}$	$\Leftrightarrow$	$H(id_{s_{B_a}} \| id_{s_a})$	$\Leftrightarrow$	$F(Path_{s_{B_a}}^{s_*} Path_{s_*}^{s_a})$

Note that the cascading operator "$\|$" orderly connects two identifiers;
s_n indicate the neighbor nodes for (n=1, ..., L_n);
s_* refers to a non-self node.

2. s_a respectively encrypts its identifier id_{s_a} and the received queries by the initial authentication key k^1_{auth} (i.e., $k_m \oplus e_{l,l}$), and then respectively replies $\{\{id_{s_a} \oplus Query_{s_n}\}_{k^1_{auth}},\, \{id_{s_a} \oplus Query_{s_{B_a}}\}_{k^1_{auth}}\}$ to $\{s_n,\, s_{B_a}\}$.

3. s_n and s_{B_a} detect the newly joined s_a,
   (a) s_n firstly performs the encryption to obtain $\{id_{s_a} \oplus Query_{s_n}\}_{k^1_{auth}}$ by its assigned id_{s_a}, and checks the validity of s_a by comparing the received value with the recomputed value. If it holds, s_n will encrypt its identifier to obtain $\{id_{s_n}\}_{k^1_{auth}}$. s_n transmits $\{id_{s_n}\}_{k^1_{auth}}$ to s_a.
   (b) Similarly, s_{B_a} firstly verifies s_a by comparing the received value with recomputed value. If it holds, s_{B_a} will encrypt its identifier to obtain $\{id_{s_{B_a}}\}_{k^1_{auth}}$. s_{B_a} transmits $\{id_{s_{B_a}}\}_{k^1_{auth}}$ to s_a.

4. Thereafter, s_a performs the decryption to obtain $\{id_{s_n},\, id_{s_{B_a}}\}$, and establishes the mapping between s_a and $\{s_n,\, s_{B_a}\}$, as shown in Table 2.

The tri-dimensional mapping shows the equivalent relations among directed paths, hashed values, and algebraic function values. For instance, a neighbor node s_x points to the sensor node s_a (i.e., $s_x \to s_a$), which is correlated with $Path_{s_x}^{s_a}$, $H(id_{s_x} \| id_{s_a})$ and $F(Path_{s_x}^{s^*} Path_{s_*}^{s_a})$. Thereinto, $F(.)$ is a pre-defined algebraic function, and it satisfies that $F(X) \oplus F(Y) = F(XY)$. Only the legal sensors can establish the legitimate relation mapping. Note that such tri-dimensional mapping that $Path_Y^X \Leftrightarrow H(id_X \| id_Y) \Leftrightarrow F(Path_X^* Path_*^Y)$ is applied for all the legal entities in Unit IoT (i.e., $\{s_a,\, s_n,\, s_{B_a},\, s_{C_a},\, MC_u\}$).

3.2.2 Authentication between the sensor node and the storage node.

$s_a \to s_{B_a}$: $id_{sen} \| Path_{s_a}^{s_x}$.
$s_{B_a} \to s_a$: $f(s_{B_a}, s_a) \| Path_{s_{B_a}}^{s_y}$, s_a checks s_{B_a}.
$s_a \to s_{B_a}$: $f(s_a, s_{B_a})$, s_{B_a} checks s_a.

1. s_a generates a session identifier id_{sen}, extracts a random path descriptor $Path_{s_a}^{s_x}$, and transmits $id_{sen}\|Path_{s_a}^{s_x}$ to s_{B_a}. It means that s_a wants to establish connection with s_{B_a} via an intermediary node s_x.

2. s_{B_a} establishes the corresponding mapping operation.
   (a) s_{B_a} determines the path descriptor $Path_{s_a}^{s_x}$ to retrieve the specific identifier id_{s_x}. s_{B_a} cascades $id_{s_{B_a}}$ and id_{s_x} to obtain $id_{s_{B_a}}\|id_{s_x}$, and computes the hashed value $H(id_{s_{B_a}}\|id_{s_x})$.

   (b) s_{B_a} derives $Path_{s_x}^{s_a}$ by the received $Path_{s_a}^{s_x}$ according to the given rule, and applies function $F(.)$ on $Path_{s_x}^{s_a}$ to obtain $F(Path_{s_x}^{s_a})$. Thereafter, s_{B_a} establishes the proof mapping that $f: s_{B_a} \rightarrow f(s_{B_a}, s_a)$.

$$f(s_{B_a}, s_a) = id_{sen} \oplus [H(id_{s_{B_a}}\|id_{s_x})\|F(Path_{s_x}^{s_a})] \tag{9}$$

   Note that underflow should be considered, and zero is padded to the higher bits. $F(Path_{s_x}^{s_a})$ equals to $F(Path_{s_x}^{s_{B_a}} Path_{s_{B_a}}^{s_a})$ according to the formula $Path_{s_x}^{s_a} = Path_{s_x}^{s_{B_a}} Path_{s_{B_a}}^{s_a}$. Based the relation mapping, we obtain that $H(id_{s_x}\|id_{s_a})$ is equivalent to $F(Path_{s_x}^{s_{B_a}} Path_{s_{B_a}}^{s_a})$.

$$f(s_{B_a}, s_a) \Leftrightarrow id_{sen} \oplus [H(id_{s_{B_a}}\|id_{s_x})\|H(id_{s_x}\|id_{s_a})] \tag{10}$$

   (c) s_{B_a} randomly chooses another random path descriptor $Path_{s_{B_a}}^{s_y}$, and transmits $f(s_{B_a}, s_a)\|Path_{s_{B_a}}^{s_y}$ to s_a.

3. Upon receiving $f(s_{B_a}, s_a)\|Path_{s_{B_a}}^{s_y}$, s_a performs the following operations.
   (a) s_a derives $\{H(id_{s_{B_a}}\|id_{s_x}), H(id_{s_x}\|id_{s_a})\}$ by $f(s_{B_a}, s_a) \oplus id_{sen}$, and computes $H'(id_{s_x}\|id_{s_a})$ with its stored identifiers $\{id_{s_x}, id_{s_a}\}$. s_a verifies s_{B_a} by comparing the locally computed $H'(id_{s_x}\|id_{s_a})$ with the derived $H(id_{s_x}\|id_{s_a})$. If it holds, the protocol will continue; otherwise, s_{B_a} will be regarded as an illegal storage node.

   (b) s_a retrieves $Path_{s_x}^{s_a}$ by the derived $H(id_{s_x}\|id_{s_a})$ according to the relation mapping. s_a checks whether the derived $Path_{s_x}^{s_a}$ and its own $Path_{s_a}^{s_x}$ satisfy the given rule. If it holds, the protocol will continue; otherwise, s_{B_a} will be regarded as an illegal storage node.

4. s_a establishes the corresponding mapping operation.
   (a) s_a determines the path descriptor $Path_{s_{B_a}}^{s_y}$ to retrieve the specific identifier id_{s_y}, then cascades id_{s_a} and id_{s_y} to obtain $id_{s_a}\|id_{s_y}$. Thereafter, s_a obtains the hashed value $H(id_{s_a}\|id_{s_y})$.

(b) s_a derives $Path_{s_y}^{s_{B_a}}$ by the received $Path_{s_{B_a}}^{s_y}$ according to the given rule. s_a extracts $Path_{s_a}^{s_y}$, and obtains $F(Path_{s_a}^{s_y} Path_{s_y}^{s_{B_a}})$. Thereafter, s_a establishes the proof mapping that $f : s_a \to f(s_a, s_{B_a})$.

$$f(s_a, s_{B_a}) = id_{sen} \oplus [H(id_{s_a} \| id_{s_y}) \| F(Path_{s_a}^{s_y} Path_{s_y}^{s_{B_a}})] \tag{11}$$

Thereinto, $F(Path_{s_a}^{s_y} Path_{s_y}^{s_{B_a}})$ is equivalent to $H(id_{s_a} \| id_{s_{B_a}})$ based on the relation mapping.

$$f(s_a, s_{B_a}) \Leftrightarrow id_{sen} \oplus [H(id_{s_a} \| id_{s_y}) \| H(id_{s_a} \| id_{s_{B_a}})] \tag{12}$$

5. Upon receiving $f(s_a, s_{B_a})$, s_{B_a} performs the following operations.

(a) s_{B_a} extracts the session identifier id_{sen}, and computes $f(s_a, s_{B_a}) \oplus id_{sen}$ to derive $\{H(id_{s_a} \| id_{s_y}),\ H(id_{s_a} \| id_{s_{B_a}})\}$. s_{B_a} computes $H'(id_{s_a} \| id_{s_{B_a}})$ with its stored identifiers $\{id_{s_a},\ id_{s_{B_a}}\}$.

(b) s_{B_a} compares the locally computed $H'(id_{s_a} \| id_{s_{B_a}})$ with the derived $H(id_{s_a} \| id_{s_{B_a}})$. If it holds, the protocol will continue and s_{B_a} and s_a will establish mutual authentication; otherwise, s_a will regard s_{B_a} as an illegal storage node and terminate the protocol.

(c) s_{B_a} derives $Path_{s_a}^{s_y}$ by the derived $H(id_{s_a} \| id_{s_y})$ based on the relation mapping. s_{B_a} integrates $\{Path_{s_{B_a}}^{s_a},\ Path_{s_a}^{s_y}\}$ into $Path_{s_{B_a}}^{s_y}$, and checks whether the integrated $Path_{s_{B_a}}^{s_y}$ and its formerly sent $Path_{s_y}^{s_{B_a}}$ satisfy the given rule. If it holds, the protocol will continue; otherwise, s_{B_a} will regard s_a as an illegal sensor node and terminate the protocol.

The proof mapping is established by the tri-dimensional equivalence relations among $\{s_a,\ s_n,\ s_{B_a}\}$, which provides assistance for the sensors to establish mutual authentication.

3.3 Proof Establishment and Verification Phase

In the phase, the data from the front-end sensors is transmitted to the layered management and data centers (i.e., M&DC, lM&DC, iM&DC, and nM&DC) for further authentication, and Figure 3 shows the exchanged messages.

3.3.1 Authentication between the storage node and the sink node.

$s_{B_a} \to s_{C_a}$: $id_{sen} \| r_{s_{B_a}}^1$.
$s_{C_a} \to s_{B_a}$: $r_{s_{C_a}}^1 \| e_{l_{C_a}^1, l_{C_a}'^1}$, s_{B_a} checks s_{C_a}.
$s_{B_a} \to s_{C_a}$: $\psi_{s_{B_a}}^{s_{C_a}} \| e_{l_{B_a}, l'_{B_a}}$, s_{C_a} checks s_{B_a}.

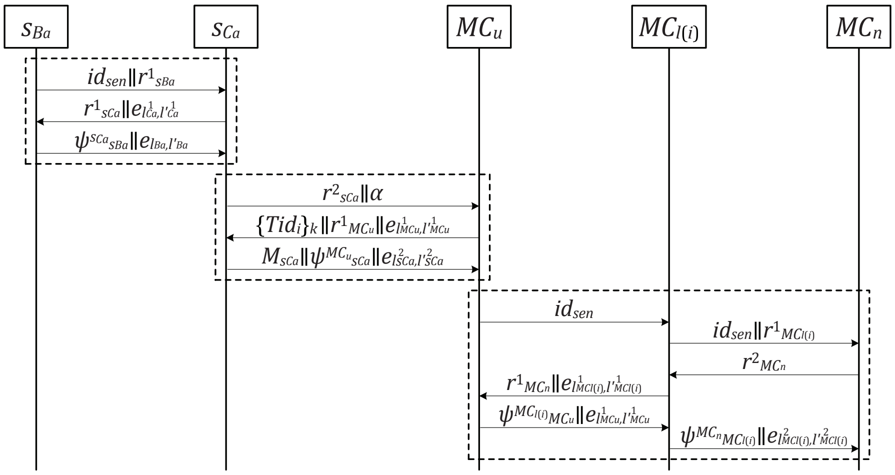

Figure 3: The exchanged messages of the proof verification.

1. s_{B_a} generates $r^1_{s_{B_a}}$, and transmits $id_{sen}\|r^1_{s_{B_a}}$ to s_{C_a}. s_{C_a} computes the subscripts $\{l^1_{C_a} = [r^1_{s_{B_a}}] \pmod{l}, l'^1_{C_a} = 2[r^1_{s_{B_a}}] \pmod{l}\}$ to extract the element $e_{l^1_{C_a}, l'^1_{C_a}}$ in the matrix $E_{L\times L}$. s_{C_a} generates a random number $r^1_{s_{C_a}}$, and replies $r^1_{s_{C_a}}\|e_{l^1_{C_a}, l'^1_{C_a}}$ to s_{B_a} for authentication.

2. Upon receiving $r^1_{s_{C_a}}\|e_{l^1_{C_a}, l'^1_{C_a}}$, s_{B_a} performs the following operations.
   (a) s_{B_a} verifies s_{C_a} by comparing the locally computed $e'_{l^1_{C_a}, l'^1_{C_a}}$ with the received $e_{l^1_{C_a}, l'^1_{C_a}}$. If it holds, the protocol will continue; otherwise, s_{B_a} will regard s_{C_a} as an illegal sink node and terminate the protocol.
   (b) s_{B_a} computes $\{l_{B_a} = [r^1_{s_{C_a}}] \pmod{l}, l'_{B_a} = 2[r^1_{s_{C_a}}] \pmod{l}\}$ to extract the element $e_{l_{B_a}, l'_{B_a}}$ in the matrix $E_{L\times L}$. s_{B_a} establishes an authentication array $\psi^{s_{C_a}}_{s_{B_a}}$, in which the hashed value $H(id_{s_a})$ replaces the real identifier id_{s_a}. s_{B_a} transmits $\psi^{s_{C_a}}_{s_{B_a}}\|e_{l_{B_a}, l'_{B_a}}$ to s_{C_a}.

$$\psi^{s_{C_a}}_{s_{B_a}} = \{\{id_{s_{B_a}}\}_{k^1_{auth}}, \{H(id_{s_a})\}_{Path^{s_{C_a}}_{s_{B_a}}}, H(id_{s_a}\|id_{s_{B_a}}) \oplus F(Path^{s_{C_a}}_{s_{B_a}})\} \tag{13}$$

3. Upon receiving $\psi^{s_{C_a}}_{s_{B_a}}\|e_{l_{B_a}, l'_{B_a}}$, s_{C_a} performs the following operations.
   (a) s_{C_a} verifies s_{B_a} by comparing the locally computed $e'_{l_{B_a}, l'_{B_a}}$ with the received $e_{l_{B_a}, l'_{B_a}}$. If it holds, s_{C_a} will decrypt $\{id_{s_{B_a}}\}_{k^1_{auth}}$ to determine the specific identifier $id_{s_{B_a}}$. Otherwise, s_{C_a} will regard s_{B_a} as an illegal storage node and terminate the protocol. Thereafter, s_{C_a} extracts $Path^{s_{B_a}}_{s_{C_a}}$

to obtain the corresponding $Path_{s_{B_a}}^{s_{C_a}}$. Therefore, $H(id_{s_a})$ is obtained by using the $Path_{s_{B_a}}^{s_{C_a}}$ for decryption.

(b) s_{C_a} applies the mapping relation to obtain $Path_{s_a}^{s_{C_a}}$.

$$\begin{aligned} &H(id_{s_a} \| id_{s_{B_a}}) \oplus F(Path_{s_{B_a}}^{s_{C_a}}) = F(Path_{s_a}^{s_*} Path_{s_*}^{s_{B_a}}) \oplus F(Path_{s_{B_a}}^{s_{C_a}}) \\ =&F(Path_{s_a}^{s_*} Path_{s_*}^{s_{B_a}} Paths_{s_{B_a}}^{s_{C_a}}) = F(Path_{s_a}^{s_{C_a}}) \Leftrightarrow Path_{s_a}^{s_{C_a}} \end{aligned} \tag{14}$$

Till now, s_{B_a} and s_{C_a} have established mutual authentication, and $\{H(id_{s_a}), Path_{s_a}^{s_{C_a}}\}$ are available by s_{C_a}.

3.3.2 Authentication between the sink node and M&DC.

$s_{C_a} \rightarrow MC_u$: $r_{s_{C_a}}^2 \| \alpha$.
$MC_u \rightarrow s_{C_a}$: $\{Tid_i\}_k \| r_{MC_u}^1 \| e_{l_{MC_u}^1, l'^1_{MC_u}}$, s_{C_a} checks MC_u.
$s_{C_a} \rightarrow MC_u$: $M_{s_{C_a}} \| \psi_{s_{C_a}}^{MC_u} \| e_{l_{C_a}^2, l'^2_{C_a}}$, MC_u checks s_{C_a}.

s_{C_a} and MC_u perform mutual verification by the randomly chosen element e_{l_*, l_*}, besides MC_u performs additional verification on s_{C_a} by the non-prior knowledge. Assume that s_{C_a} and MC_u share a set of token identifiers $\{Tid_i\}_j$ $(i = \{1, ..., j\})$, which satisfy that,

$$\sum_{i=1}^{j} Tid_i \cdot id_{s_{C_a}} \equiv 1 \pmod p \tag{15}$$

It means that the sum of any product that $Tid_i \cdot id_{s_{C_a}}$ (for $i = 1, ...j$), has the same remainder with 1 upon division by the large prime number p.

1. s_{C_a} generates $r_{s_{C_a}}^2$, computes $\alpha = (r_{s_{C_a}}^2)^l \pmod p$, and transmits $r_{s_{C_a}}^2 \| \alpha$ to MC_u as a query.

2. MC_u randomly chooses k token identifiers $\{Tid_i\}_k = \{Tid_1, Tid_2, ..., Tid_k\}$ as a response to s_{C_a}. Hereafter, MC_u generates $r_{MC_u}^1$, computes $e_{l_{MC_u}^1, l'^1_{MC_u}}$, and replies $\{Tid_i\}_k \| r_{MC_u}^1 \| e_{l_{MC_u}^1, l'^1_{MC_u}}$ to s_{C_a}, in which the subscripts satisfy $\{l_{MC_u}^1 = [r_{s_{C_a}}^2] \pmod l, l'^1_{MC_u} = 2[r_{s_{C_a}}^2] \pmod l\}$.

3. Upon receiving the messages, s_{C_a} performs the following operations.
   (a) s_{C_a} verifies MC_u by checking whether the locally computed $e'_{l_{MC_u}^1, l'^1_{MC_u}}$ equals to the received $e_{l_{MC_u}^1, l'^1_{MC_u}}$. If it holds, the protocol will continue; otherwise, s_{C_a} will regard MC_u as an illegal M&DC and terminate the protocol.

(b) s_{C_a} computes $M_{s_{C_a}}$ by its stored token identifiers Tid_i, and establishes an authentication array $\psi_{s_{C_a}}^{MC_u}$.

$$M_{s_{C_a}} = r_{s_{C_a}}^2 \prod_{i=1}^{k} e^{Tid_i} \pmod p \tag{16}$$

$$\begin{aligned}\psi_{s_{C_a}}^{MC_u} =&\{\{id_{sen}\|id_{s_{C_a}}\}_{k_m}, F(Path_{s_a}^{s_{C_a}}) \oplus F(Path_{s_{C_a}}^{MC_u}),\\ &\{H(id_{s_a})\}_{Path_{s_{C_a}}^{MC_u}}\}\end{aligned} \tag{17}$$

(c) s_{C_a} computes the subscripts $\{l_{C_a}^2 = [r_{MC_u}^1] \pmod l,\ l'^2_{C_a} = 2[r_{MC_u}^1] \pmod l\}$ to extract $e_{l_{C_a}^2, l'^2_{C_a}}$, and transmits $M_{s_{C_a}}\|\psi_{s_{C_a}}^{MC_u}\|e_{l_{C_a}^2, l'^2_{C_a}}$ to MC_u.

4. Upon receiving the messages, MC_u performs the following operations.

(a) MC_u firstly verifies s_{C_a} by checking whether the locally computed $e'_{l_{C_a}^2, l'^2_{C_a}}$ equals to the received $e_{l_{C_a}^2, l'^2_{C_a}}$. If it holds, MC_u will perform decryption on $\{id_{sen}\|id_{s_{C_a}}\}_{k_m}$ to determine the identifier $id_{s_{C_a}}$; otherwise, MC_u will regard s_{C_a} as an illegal sink node and terminate the protocol.

(b) MC_u computes $M_{MC_u} = (e^{id_{s_{C_a}}})^k \pmod p$ by its identifiers $id_{s_{C_a}}$. Afterwards, MC_u computes $(M_{s_{C_a}} \cdot M_{MC_u})^l$, and compares αe^l with $(M_{s_{C_a}} \cdot M_{MC_u})^l$.

$$(M_{s_{C_a}} \cdot M_{MC_u})^l = (r_{s_{C_a}}^2 e^{\sum_{i=1}^{k} Tid_i \cdot id_{s_{C_a}}})^l \pmod p \tag{18}$$

$$\alpha e^l = (r_{s_{C_a}}^2 \cdot e)^l \pmod p \overset{?}{=} (M_{s_{C_a}} \cdot M_{MC_u})^l \tag{19}$$

If it holds, the protocol will continue; otherwise, MC_u will regard s_{C_a} as an illegal sink node and terminate the protocol.

(c) MC_u applies the algebraic function to obtain $Path_{s_a}^{MC_u}$,

$$\begin{aligned}&F(Path_{s_a}^{s_{C_a}}) \oplus F(Path_{s_{C_a}}^{MC_u}) = F(Path_{s_a}^{s_{C_a}} Path_{s_{C_a}}^{MC_u})\\ =&F(Path_{s_a}^{MC_u}) \Leftrightarrow Path_{s_a}^{MC_u}\end{aligned} \tag{20}$$

Till now, s_{C_a} and MC_u have established mutual authentication, the directed path descriptor $Path_{s_a}^{MC_u}$ is available by MC_u, and $Path_{MC_u}^{s_{C_a}}$ is derived via the invertible operator and anti-homomorphism function which is applied to obtain the hashed identifier $H(id_{s_a})$.

3.3.3 Authentication between the management and data centers (i.e., M&DC, lM&DC, iM&DC, and nM&DC).

$MC_u \to MC_{l(i)}$: id_{sen}.
$MC_{l(i)} \to MC_n$: $id_{sen} \| r^1_{MC_{l(i)}}$.
$MC_n \to MC_{l(i)}$: $r^2_{MC_n}$.
$MC_{l(i)} \to MC_u$: $r^1_{MC_n} \| e_{l^1_{MC_{l(i)}}, l'^1_{MC_{l(i)}}}$, MC_u checks $MC_{l(i)}$.
$MC_u \to MC_{l(i)}$: $\psi^{MC_{l(i)}}_{MC_u} \| e_{l^1_{MC_u}, l'^1_{MC_u}}$, $MC_{l(i)}$ checks MC_u.
$MC_{l(i)} \to MC_n$: $\psi^{MC_n}_{MC_{l(i)}} \| e_{l^2_{MC_{l(i)}}, l'^2_{MC_{l(i)}}}$, MC_n checks $MC_{l(i)}$.

1. $\{MC_u, MC_{l(i)}, MC_n\}$ perform the following operations.
   (a) MC_u firstly challenges $MC_{l(i)}$ by the derived session identifier id_{sen}; $MC_{l(i)}$ generates a random number $r^1_{MC_{l(i)}}$, and transmits $id_{sen} \| r^1_{MC_{l(i)}}$ to MC_n; MC_n generates a random number $r^1_{MC_n}$, and computes $r^2_{MC_n} = id_{sen} \oplus r^1_{MC_{l(i)}} \oplus r^1_{MC_n}$, and replies $r^2_{MC_n}$ to $MC_{l(i)}$.
   (b) $MC_{l(i)}$ derives $r^1_{MC_n}$, computes $r^2_{MC_{l(i)}} = r^2_{MC_n} \oplus r^1_{MC_{l(i)}}$, and extracts the element $e_{l^1_{MC_{l(i)}}, l'^1_{MC_{l(i)}}}$, in which the subscripts satisfy $\{l^1_{MC_{l(i)}} = [r^2_{MC_{l(i)}}] \pmod{l}, l'^2_{MC_{l(i)}} = 2[r^2_{MC_{l(i)}}] \pmod{l}\}$. Then, $MC_{l(i)}$ transmits $r^1_{MC_n} \| e_{l^1_{MC_{l(i)}}, l'^1_{MC_{l(i)}}}$ to MC_u for authentication.
2. Upon receiving the message, MC_u performs the following operations.
   (a) MC_u computes $r'^2_{MC_{l(i)}} = id_{sen} \oplus r^1_{MC_n}$, and extracts $e'_{l^1_{MC_{l(i)}}, l'^1_{MC_{l(i)}}}$ by $r'^2_{MC_{l(i)}}$ according to the same algorithm as $e_{l^1_{MC_{l(i)}}, l'^1_{MC_{l(i)}}}$. MC_u compares whether the locally computed value equals to the received value. If it holds, the protocol will continue.
   (b) MC_u continues to compute $e_{l^1_{MC_u}, l'^1_{MC_u}}$, in which the subscripts satisfy $\{l^1_{MC_u} = [r^1_{MC_n}] \pmod{l}, l'^2_{MC_u} = 2[r^1_{MC_n}] \pmod{l}\}$. MC_u establishes an authentication array $\psi^{MC_{l(i)}}_{MC_u}$, and transmits $\psi^{MC_{l(i)}}_{MC_u} \| e_{l^1_{MC_u}, l'^1_{MC_u}}$ to $MC_{l(i)}$ for authentication.

$$\begin{aligned}\psi^{MC_{l(i)}}_{MC_u} =& \{\{id_{MC_u}\}_{k_{u_{l(i)}} \oplus id_{sen}}, F(Path^{MC_u}_{s_a}) \oplus F(Path^{MC_{l(i)}}_{MC_u}),\\ & \{H(id_{s_a}) \oplus Path^{MC_{l(i)}}_{s_a}\}_{Path^{MC_{l(i)}}_{MC_u} \oplus id_{sen}}\}\end{aligned} \tag{21}$$

3. Upon receiving the messages, $MC_{l(i)}$ performs the following operations.
   (a) $MC_{l(i)}$ extracts $e'_{l^1_{MC_u}, l'^1_{MC_u}}$ by the stored $r^1_{MC_{l(i)}}$. Thereafter, $MC_{l(i)}$ compares whether the local value equals to the received value. If it holds, the protocol will continue.

(b) For one hand, $MC_{l(i)}$ decrypts $\{id_{MC_u}\}_{k_{u_{l(i)}} \oplus id_{sen}}$ to obtain id_{MC_u}, which is used to determine the specific identity of $MC_{l(i)}$. $MC_{l(i)}$ retrieves the corresponding $Path_{MC_u}^{MC_{l(i)}}$. For the other hand, $MC_{l(i)}$ obtains $Path_{s_a}^{MC_{l(i)}}$ by $F^{-1}(F(Path_{s_a}^{MC_u}) \oplus F(Path_{MC_u}^{MC_{l(i)}}))$ according to the algebraic function $F(X) \oplus F(Y) = F(XY)$. Thereafter, $MC_{l(i)}$ decrypts $\{H(id_{s_a}) \oplus Path_{s_a}^{MC_{l(i)}}\}_{Path_{MC_u}^{MC_{l(i)}} \oplus id_{sen}}$, and obtains the desired $H(id_{s_a})$.

(c) $MC_{l(i)}$ extracts $e_{l^2_{MC_{l(i)}}, l'^2_{MC_{l(i)}}}$, in which $\{l^2_{MC_{l(i)}} = [r^2_{MC_{l(i)}}] \pmod l$, $l'^2_{MC_{l(i)}} = 2[r^2_{MC_{l(i)}}] \pmod l\}$. Thereafter, $MC_{l(i)}$ establishes an authentication array $\psi_{MC_{l(i)}}^{MC_n}$, and transmits $\psi_{MC_{l(i)}}^{MC_n} \| e_{l^2_{MC_{l(i)}}, l'^2_{MC_{l(i)}}}$ to MC_n for the final authentication.

$$\begin{aligned}\psi_{MC_{l(i)}}^{MC_n} =& \{\{id_{MC_{l(i)}} \oplus id_{sen}\}_{k_{u_{nl(i)}}}, F(Path_{s_a}^{MC_{l(i)}}) \oplus F(Path_{MC_{l(i)}}^{MC_n}),\\ & \{H(id_{s_a}) \oplus Path_{s_a}^{MC_n}\}_{Path_{MC_{l(i)}}^{MC_n}}\}\end{aligned} \tag{22}$$

4. Upon receiving the messages, MC_n performs the following operations.

(a) MC_n locally computes $r'^2_{MC_{l(i)}}$, and extracts $e'_{l^2_{MC_{l(i)}}, l'^2_{MC_{l(i)}}}$. MC_n performs decryption to obtain $id_{MC_{l(i)}}$. MC_n retrieves $Path_{MC_n}^{MC_{l(i)}}$ by the derived $id_{MC_{l(i)}}$, and further obtains the corresponding $Path_{MC_{l(i)}}^{MC_n}$.

(b) Similarly, MC_n obtains $Path_{s_a}^{MC_n}$ according to the algebraic function, which realizes that the directed path transition from s_a to MC_n is available. Finally, $H(id_{s_a})$ is obtained by applying $\{Path_{s_a}^{MC_n}, Path_{MC_{l(i)}}^{MC_n}\}$ for decryption.

Till now, $MC_{l(i)}$ has independently established the corresponding trust with MC_u, and MC_n has authenticated $MC_{l(i)}$, in which $\{MC_u, MC_{l(i)}, MC_n\}$ are interdependent to establish trusts under the layered organization structure.

4 Security and Performance Analysis

4.1 Security Analysis

In the U2IoT architecture, the wireless open channels among the sensor nodes, storage nodes, and sink nodes are confronting severe circumstances, besides the back-end management and data centers are also suffering from traditional threats. In the section, security analysis is performed to show that DPAS has been designed to satisfy security properties.

4.1.1 Data Confidentiality and Integrity

Data confidentiality and integrity require that the exchanged messages are securely transmitted without any unauthorized disclosure, skimming, modification, and destruction during authentication sessions. In DPAS, three types of keys $\{k_{auth},\ k_m,\ k_u\}$ and the one-way hash function are defined to provide strong data protection.

- The sensor nodes, the neighbor nodes and the storage node transmit ciphertexts based on k^1_{auth} in the first session, and the updated k^x_{auth} in the x-th session. Thereinto, the secret matrix $E_{L\times L}$ is introduced to extract an element that acts as an operator for authentication key updating. Meanwhile, the storage node and the sink node perform proof verification by the authentication arrays, in which k^1_{auth} and / or k^x_{auth} are applied to hide their real identifiers.
- The sink node and M&DC of Unit IoT use the strong master key k_m to protect $id_{s_{C_a}}$ against the unauthorized entities. The management data center MC_u of Unit IoT, and the centers (MC_l, MC_i, MC_n) of Ubiquitous IoT use the corresponding shared keys $(k_{u_l}, k_{u_i}, k_{u_{nl}}, k_{u_{ni}})$ to realize data protection. Meanwhile, the session identifier id_{sen} is also introduced as an operator for encryption.
- The hash function is applied to enhance data integrity, and the hashed values cannot be deduced by the malicious attackers. Even if a robust attacker succeeds to modify the exchanged data, the legal nodes will not deduce the inconsistent values, and will recognize the illegal attacker.

4.1.2 Authentication

Mutual authentication is established to realize cross-network authentication.

- During key distribution phase, DH key agreement is applied to establish the shared keys $(k_{u_l},\ k_{u_i},\ k_{u_{nl}},\ k_{u_{ni}})$, in which mutual authentication is performed by checking the consistence of the derived path descriptor and the local path descriptor according to the anti-homomorphism function.
- During mapping phase, s_a and s_{B_a} establish mutual trust by the mapping operators $f(s_{B_a}, s_a)$ and $f(s_a, s_{B_a})$, in which tri-dimensional equivalence relation and the pre-defined function are introduced to prove that only the legal entity can achieve the consistent values.
- During verification phase, $\{\psi^{s_{C_a}}_{s_{B_a}},\ \psi^{MC_u}_{s_{C_a}},\ \psi^{MC_{l(i)}}_{MC_u}, \psi^{MC_n}_{MC_{l(i)}}\}$ are used as authentication operators, in which the directed path and its extended relations are also used for non-prior knowledge verification, along with the elements

e_{l_*,l'_*} are extracted based on the random numbers' modulo and rounding operations. Note that any cross-network accessing without an available intermediary path cannot be permitted.

4.1.3 Anonymity and Forward Security

Anonymity is ensured by using pseudo-random identifiers (including node identifier, management and data center identifier, and session identifier), hashed identifiers, and the token identifiers instead of exposing the real identifiers. The illegal attackers cannot track or ascertain which entity the intercepted messages belong to due to the irregular pseudo-random values. The top management and data centers can recognize specific sensor node based on the hashed identifier $H(id_{s_a})$ without needing acquire the real identifier id_{s_a}, which provides a non-reversible data protection. Additionally, the token identifier Tid_i satisfies the given algebraic function with the real identifier $id_{s_{C_a}}$, which can establish the zero-knowledge proof.

Forward security is achieved by the dynamically updated numbers. The pseudo-random session identifier id_{sen} is used to ensure the freshness of each authentication session, the virtual directed paths are interactive to realize verification, and the one-way hashed identifier $H(id_{s_a})$ is transmitted to inform {M&DC, lM&DC, iM&DC, nM&DC} about the active sensor node. The attacker regards the prior session as random even if the nodes and M&DCs get corrupted, and regards the later session as random even if the attacker can access the current session.

4.2 Performance Analysis

Performance is another fundamental aspect besides security, and the balance between security and performance is necessary. In DPAS, the performance is evaluated with respect to communication overhead and computation load.

The communication overhead mainly refers to the data packets communicated in the communication channels. Assume that the real identifier/session identifier id_*/id_{sen} has the standard length l (e.g., 64 bits). The hashed value $H(.)$, and encrypted value $\{.\}_k$, and function value $F(.)$ have the length of $2l$ (128 bits). $Query$, random number r_*, path descriptor $Path_*$, token identifier Tid_*, and element e_{l_*,l'_*} have the length of $l/4$ (16 bits). In the proof mapping phase, the communication overload is 34 bytes between s_a and s_n, and is 78 bytes between s_a and s_{B_a}. In the proof verification phase, 1) s_{C_a} exchanges 64 bytes with s_{B_a}, and exchanges 60+$2k$ bytes with MC_u, in which k is the number of the chosen token identifier; 2) $MC_{l(i)}$ exchanges 62 bytes and 60 bytes to MC_u and MC_n. The protocol completes via 21 steps, which realizes the layered authentication from the basic sensor node to the top national management and data center. Note that the amount of data packets in both forward and backward channels are suitable for universal network environments.

The main computation load consists of the pseudo-random number generation, bitwise logic operation, hash operation, encryption operation, and other algebraic function. Particularly, the element e_{l_*,l'_*} of the pre-shared matrix $E_{L\times L}$ is introduced for preliminary authentication, which makes that the protocol may not perform redundant computations on an illegal entity. The ultralightweight functions (bitwise logical function and algebraic function), the lightweight functions (hash function and pseudo-random function) are applied as the main operations, and the efficient cryptography algorithms (e.g., elliptic curve cryptography) can be applied to alleviate the total computation load.

5 Conclusion

In this paper, a directed path based authentication scheme (DPAS) is proposed for the U2IoT architecture. The main motivation is to promote mapping and verification for the sensor nodes in the U2IoT architecture. In DPAS, authentication keys among the sensor node and its neighbor nodes are distributed by performing the path based verification, and the mapping is established in the form of tri-dimensional equivalence relation. The directed path descriptor is introduced to provide cross-network authentication for the layered architecture. It indicates that DPAS is appropriate for the U2IoT architecture and other IoTs.

Acknowledgements

This work is jointly funded by National Natural Science Foundation of China (NSFC) and Civil Aviation Administration of China (CAAC) (61079019). This work is also supported by the Fundamental Research Funds for the Central Universities (YWF-11-02-264).

References

[Amin et al. 2009] Amin S. O., Siddiqui M. S., Hong C. S., and Lee S.: "RIDES: Robust Intrusion Detection System for IP-Based Ubiquitous Sensor Networks"; Sensors, vol. 9, no. 5, pp. 3447-3468, 2009.

[Atzori et al. 2010] Atzori, L., Iera A., and Morabito G.: "The Internet of Things: A Survey"; Computer Networks, vol. 54, no.15, pp. 2787-2805, 2010.

[Bandyopadhyay and Sen 2011] Bandyopadhyay D. and Sen J.: "Internet of Things: Applications and Challenges in Technology and Standardization"; Wireless Personal Communications, vol. 58, no. 1, pp. 49-69, 2011.

[Claycomb and Shin 2011] Claycomb W. R. and Shin D.: "A Novel Node Level Security Policy Framework for Wireless Sensor Networks"; Journal of Network and Computer Applications, vol. 34, no. 1, pp. 418-428, 2011.

[Delgado-Mohatar et al. 2011] Delgado-Mohatar O., Fuster-Sabater A., and Sierra J. M.: "A Light-Weight Authentication Scheme for Wireless Sensor Networks"; Ad Hoc Networks, vol. 9, no. 5, pp. 727-735, 2011.

[Gabrielli et al. 2011] Gabrielli A., Mancini L. V., Setia S., and Jajodia S.: "Securing Topology Maintenance Protocols for Sensor Networks"; IEEE Transactions on Dependable and Secure Computing, vol. 8, no. 3, pp. 450-465, 2011.

[Hancke et al. 2010] Hancke G. P., Markantonakis K., and Mayes K. E.: "Security Challenges for User-Oriented RFID Applications within the "Internet of Things""; Journal of Internet Technology, vol. 11, no. 3, pp. 307-313, 2010.

[Kortuem et al. 2010] Kortuem G., Kawsar F., Fitton D., and Sundramoorthy V.: "Smart Objects as Building Blocks for the Internet of Things"; IEEE Internet Computing, vol. 14, no. 1, pp. 44-51, 2010.

[Koshizuka and Sakamura 2010] Koshizuka N. and Sakamura K.: "Ubiquitous ID: Standards for Ubiquitous Computing and the Internet of Things"; IEEE Pervasive Computing, vol. 9, no. 4, pp. 98-101, 2010.

[Ma et al. 2011] Ma J., Wen J., Huang R., and Huang B.: "Cyber-Individual Meets Brain Informatics"; IEEE Intelligent Systems, vol. 26, no. 5, pp. 30-37, 2011.

[Ning and Wang 2011] Ning H. and Wang Z.: "Future Internet of Things Architecture: Like Mankind Neural System or Social Organization Framework?"; IEEE Communications Letters, vol. 15, no.4, pp. 461-463, 2011.

[Peering et al. 2002] Peering A., Szewczyk R., Wen V., Cullar D., Tygar J. D.: "Spins: Security Protocols for Sensor Networks"; Wireless Networks, vol. 8, no. 5, pp. 521-534, 2002.

[Roman et al. 2011a] Roman R., NajeraP., and Lopez J.: "Securing the Internet of Things"; Computer, vol. 44, no. 9, pp. 51-58, 2011.

[Roman et al. 2011b] Roman R., Lopez J., and Najera P.: "A Cross-Layer Approach for Integrating Security Mechanisms in Sensor Networks Architectures"; Wireless Communications & Mobile Computing, vol. 11, no. 2, pp. 267-276, 2011.

[Roman et al. 2011c] Roman R., Alcaraz C., Lopez J., and Sklavos N.: "Key Management Systems for Sensor Networks in the Context of the Internet of Things"; Computers & Electrical Engineering, vol. 37, no.2, pp. 147-159, 2011.

[Tsai et al. 2009] Tsai S. C., Tzeng W. G., and Zhou K. Y.: "Key Establishment Schemes Against Storage-Bounded Adversaries in Wireless Sensor Networks"; IEEE Transactions on Wireless Communications, vol. 8, no. 3, pp. 1218-1222, 2009.

[Weber 2010] Weber R. H.: "Internet of Things-New Security and Privacy Challenges"; Computer Law & Security Review, vol. 26, no. 1, pp. 23-30, 2010.

[Xu 2011] Xu L. D.: "Information Architecture for Supply Chain Quality Management"; International Journal of Production Research, vol. 49, no. 1, pp. 183-198, 2011.

[Yan and Wen 2011] Yan, T. and Wen, Q.: "Building the Internet of Things Using a Mobile RFID Security Protocol Based on Information Technology"; Advances in Intelligent and Soft Computing, vol. 104, pp. 143-149, 2011.

[Zhang et al. 2012a] Zhang D., Wan J., Liu Q., Guan X., and Liang X.: "A Taxonomy of Agent Technologies for Ubiquitous Computing Environments"; KSII Transactions on Internet and Information Systems, vol. 6, no. 2, pp. 547-565, 2012.

[Zhang et al. 2011b] Zhang Y., Yu R., Xie S., Yao W., Xiao Y., and Guizani M.: "Home M2M Networks: Architectures, Standards, and QoS Improvement"; IEEE Communications Magazine, vol. 49, no. 4, pp. 44-52, 2011.

[Zhong et al. 2010] Zhong N., Ma J., Huang R., Liu J., Yao Y., Zhang Y., and Chen J.: "Research Challenges and Perspectives on Wisdom Web of Things (W2T)"; Journal of Supercomputing, DOI 10.1007/s11227-010-0518-8.

[Zhou and Chao 2011] Zhou L. and Chao H. C.: "Multimedia Traffic Security Architecture for the Internet of Things"; IEEE Network, vol. 25, no. 3, pp. 35-40, 2011.