Research Article

International Journal of
Distributed
Sensor Networks

International Journal of Distributed
Sensor Networks
2017, Vol. 13(6)
© The Author(s) 2017
DOI: 10.1177/1550147717713643
journals.sagepub.com/home/ijdsn
SAGE

A secure routing scheme for underwater acoustic networks

Xiujuan Du[1,2], Chunyan Peng[1,2] and Keqin Li[3]

Abstract

Secure and anonymous routing is required in many underwater acoustic network applications such as marine military. However, the characteristics of underwater acoustic networks cause existing secure scheme designed for traditional terrestrial networks to be inapplicable. This article presents a secure routing design for underwater acoustic networks. First, considering the difficulty of setting a trusted third party in underwater acoustic networks, a short signature algorithm without any online trusted third party is proposed and is used in the procedure of route setup for authentication between source and destination node pair. Analysis shows that the proposed signature scheme can resist forgery attacks effectively and improve communication security and signature efficiency. Second, a trap-door scheme in routing messages based on bilinear map is presented, which achieves anonymity of communication nodes to forwarding nodes. Finally, the anonymity of intermediate nodes in the routing path is also realized by encoding their session ID. Simulation results show the secure routing scheme has moderate network performance under the premise of secure communication.

Keywords

Bilinear map, secure routing, short signature, underwater acoustic networks, trap door

Date received: 11 December 2016; accepted: 9 May 2017

Academic Editor: Jaime Lloret

Introduction

Recently, research on underwater acoustic networks (UANs) has attracted significant attention due to its potential application in environmental monitoring, resource investigation, disaster prevention, and so on.[1–4] UANs usually include underwater nodes, surface repeater, ship-based receiving stations, satellites, and ground stations. Underwater acoustic nodes are randomly deployed in the monitoring area. In order to monitor in an all-round way, underwater nodes usually float at different depths. The nodes can move with water currents or other underwater activities and form a network through self-organization. The information collected is forwarded by neighbor nodes hop-by-hop, then arrives at the water surface repeater after several transmissions, and finally reaches the ground base station via satellite or the Internet. The communication model of UANs is shown in Figure 1.

UANs adopt acoustic communication, and acoustic channel is characterized by high bit error, long propagation delay, and narrow bandwidth. Compared with conventional modems, acoustic modems are more energy consuming. However, nodes are battery-powered and it is difficult to recharge and replace the batteries in harsh underwater environments. Furthermore, underwater nodes are usually deployed more sparsely, most nodes can move passively with water currents or other

[1]School of Computer Science, Qinghai Normal University, Xining, China
[2]Key Laboratory of IoT of Qinghai Province, Qinghai Normal University, Xining, China
[3]Department of Computer Science, The State University of New York at New Paltz, New Paltz, NY, USA

Corresponding author:
Xiujuan Du, School of Computer Science, Qinghai Normal University, Xining 810008, Qinghai, China.
Email: dxj@qhnu.edu.cn

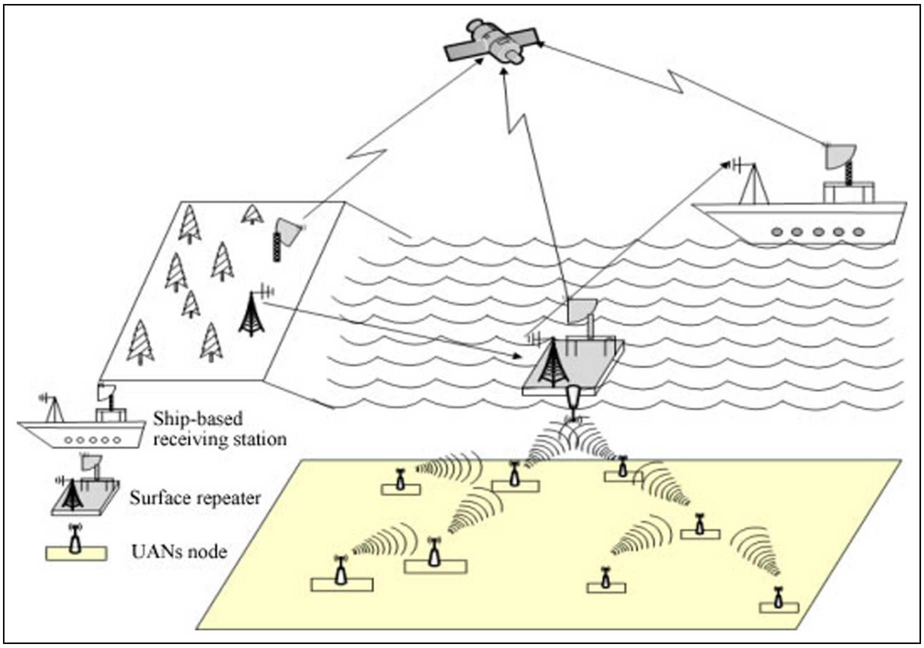

Figure 1. Communication model of UANs.

underwater activities, and some nodes will fail due to energy depletion or hardware fault, so the network topology of UANs usually changes dynamically. All those characteristics result in terrestrial-based communication protocols and are either inapplicable or inefficient for UANs, and UANs call for a new communication protocol.

The application of UANs in some areas such as business and military is usually sensitive, and outsiders are not allowed to access sensitive information. However, the nature of opening and sharing of underwater acoustic channel makes UAN communications vulnerable to eavesdropping and interfering, so secure anonymous communication has a broad application prospect. However, there are few papers on security communication scheme for UANs so far.[5–8] The highly dynamic nature and the lack of centralized management and control bring about great challenges to the design of secure routing protocols.

In this article, a new digital short signature scheme without any real-time trusted third party is proposed. Based on the signature scheme, we realize two-way authentication between source and destination node pair as well as resolve the problem of key escrow in UANs. Furthermore, we present a trap-door scheme for route messages, which achieves anonymity of communication nodes to the intermediate nodes in routing path. Finally, the anonymity of intermediate nodes in the routing path is realized based on the idea of multiprotocol label switching (MPLS).

The main contributions of this article are summarized as follows:

1. Traditional secure routing requires a trusted public key generate (PKG), which has the pair of public and secret keys of each node, and each node has to trust the PKG unconditionally, which is impractical for UANs where attack and hostile nodes are present, and the PKG can forge routing messages. In this article, we provide an algorithm of signature and verification without trusted PKGs.
2. In traditional secure routing, source nodes need to request the public key of destination node to an online PKG, which significantly increases communication delay and is also impractical for UANs. In our design, the PKG publishes its system parameters and generates session IDs (SIDs) for other nodes offline.
3. In our secure anonymous routing scheme, an attack node cannot obtain the identity information of source node, destination node, or forwarding nodes by analyzing routing messages, so our scheme realizes the anonymity of node identity.
4. Our routing scheme realizes two-way authentication between the node pair of source and destination and can resist the forgery attacks from other nodes (e.g. the PKG). In addition, our routing scheme realizes location privacy, that is, an attack node cannot obtain the information of location and hop count of other nodes by analyzing the routing messages.
5. To the best of our knowledge, it is the first work to study secure anonymous routing scheme for UANs.

The remainder of the article is organized as follows. Section "Related work" briefly discusses related work. Section "Design of digital short signature" presents the scheme of digital short signature, and the efficiency of the proposed scheme is analyzed. Section "Design of secure anonymous routing scheme" presents the design of trap door and secure routing in detail. Section "Protocol analysis" analyzes the security and anonymity of the routing scheme. In section "Experiment and conclusion," we evaluate the performance through simulations and make a conclusion.

Related work

Network security has been well studied in terrestrial networks and improved solutions are continually being developed.[9–11] However, the nature of the underwater acoustic channel makes existing terrestrial defenses unable to be directly applied to UANs. In UANs, if some nodes are captured, the security of the entire network will be threatened. Therefore, the node's security remains essential, and the security of UANs has been an increasing serious problem, but limited work has been conducted on studying security mechanisms in UANs. In this section, we present a few related works in security-related technologies of UANs.

Dong et al.[5] analyzed the security issues of UANs, investigated the goals and challenges of UAN security, and classified the security threats according to their harm to UAN. However, the authors did not propose a specific security algorithm design. Cong et al.[6] analyzed the threats and attacks in UANs. They pointed out that owing to the unique characteristics, UANs are more vulnerable to malicious attacks. They suggested to design layered security structure to resist hybrid attacks. However, they neither presented how to carry out the security structure nor provided any efficient algorithms. Dini and Lo Duca[7] presented a cryptographic suite which can protect end-to-end communication, and a ciphertext stealing (CTS) mode was used to avoid ciphertext size expansion. The cryptographic suite provided algorithms of membership management service, the key management service and the secure dispatching service, which were used to provide vehicle authentication, messages confidentiality. However, the cryptographic suite requires a trusted real-time group management system (GMS), which is a difficult problem in open UANs. Ji et al.[12] analyzed the requirements and security issues of UANs and considered a symmetric scheme of data encryption and decryption which did not involve the realization of anonymity or authentication. Zhang and Zhang[13] introduced a set of neighbor discovery protocols based on the direction-of-arrival (DoA) estimation of acoustic signals, which consist of four schemes: basic Neighbor Discovery Protocol (B-NDP), double-Verification Neighbor Discovery Protocol (DV-NDP), Strict Double-Verification Neighbor Discovery Protocol (SDV-NDP), and Mobility-Aware Neighbor Discovery Protocol (MA-NDP). All of the four schemes are resilient to wormhole attacks without conventional requirements on secure localization and accurate time synchronization. It is the first secure neighbor discovery protocol for UANs. Wang et al.[14] presented a distributed technique to detect wormhole links in UANs without any special hardware. The nodes can reconstruct the local network topology using multi-dimensional scaling (MDS) and locate fake connections, and adapt to the dynamic network topology of UANs.

From above work, it can be seen that security-related research is being actively conducted, but it is still in its nascent stages. The security research of UANs so far mainly focused on the challenge analysis, classification and detection of attack, wormhole thwarting, algorithms of symmetric encryption and secret key generation,[15–17] and lightweight encryption scheme.[8] A few among them were designed considering anonymous routing aspects of UANs, and the protection of privacy in existing secure technologies for UANs is far from enough. In UANs, an intermediate node can obtain easily the IDs of the source, destination, and forwarding nodes, which is not applicable for those requiring anonymous communication. Additionally, most existing authentication methods for UANs are based on asymmetric cryptosystem which requires an online public key infrastructure (PKI), which is difficult for UANs. So, anonymous routing and secure communication of UANs have become an urgent problem to be solved. In this article, a digital signature scheme without trusted third party is proposed, which can solve the problem of key escrow. Based on the schemes of short signature and trap door, we design an anonymous secure routing scheme and realize secure communication in UANs.

Design of digital short signature

Considering the difficulty of setting a trusted third party in UANs, in this article, we improve the digital short signature algorithm by Chen et al.[9] and propose a more efficient signature scheme without an online trusted third party. Before presenting the improved sign scheme, three intractable problems assumptions are introduced first.

Assumptions

Let G_1 denote a cyclic additive group whose order is a big prime q; let G_2 denote a cyclic multiplicative group of the same order; let Z_q denote an additive group modulo q, $Z_q = \{0, \dots, q-1\}$; and α and b are two different elements in $Z_{q,}^{*} Z_q^{*} = Z_q/\{0\}$. A bilinear pairing $\hat{e}$ is a map $\hat{e} : G_1 \times G_1 \rightarrow G_2$. Given $P, Q, R \in G_1$, we can find

Table 1. Intractable problems assumption.

Abbreviations of intractable problems	Formula and depiction ($\alpha, b, c, n \in Z_q^*$)
DLP	Given $P, Q \in G_1$ or G_2, $Q = nP$, to find $n \in Z_q^*$
CDH	Given $P, \alpha P, bP, \alpha, b \in Z_q^*$, compute $\hat{e}(P,P)^{\alpha b}$ in polynomial time
GDH	Given $P, \alpha P, bP, cP,\ P \in G_1^*$, compute $U = \hat{e}(P,P)^{\alpha bc} \in G_2, \quad \alpha, b, c \in Z_q^*$

DLP: discrete logarithm problem; CDH: computational Diffie–Hellman; GDH: gap Diffie–Hellman.

an efficient algorithm to compute $\hat{e}(P,Q)$, so $\hat{e}$ is considered to be computable and bilinear. The bilinear property of $\hat{e}$ is given by equations (1) and (2)

$$\hat{e}(P, Q + R) = \hat{e}(P,Q) \cdot \hat{e}(P,R) \tag{1}$$

$$\hat{e}(aP, bQ) = \hat{e}(P,Q)^{ab} = \hat{e}(P, abQ) = \hat{e}(abP, Q) \tag{2}$$

Here, three intractable problems are assumed, computational Diffie–Hellman (CDH), decisional Diffie–Hellman (DDH), and discrete logarithm problem (DLP), which are shown in Table 1. If there is an algorithm by which the CDH and DDH problems can be solved in polynomial time but the DLP cannot be solved, we call G_1 as a gap Diffie–Hellman (GDH) group. GDH group can be found in super singular elliptic curve over finite field, and the bilinear pairings can be derived from the Weil or Tate pairings in GDH group.

Scheme design

The nodes in the scheme are classified into three types: signature nodes, authentication nodes, and an untrusty third party. The implementation of the scheme is divided into four phases: system initialization, user key extraction, signature, and verification, which are as follows.

System initialization. The first phase is system initialization. An untrusty third party (as a PKG) constructs a cyclic additive group G_1 and a cyclic multiplicative group G_2 over finite field F_p offline, both with the order of prime q, and the generating element of G_1 is P. Then, the PKG constructs an admissible bilinear map $\hat{e} : G_1 \times G_1 \rightarrow G_2$ based on Weil pairings on elliptic curve and defines two hash functions given by equations (3) and (4)

$$H_1 : \{0,1\}^* \times G_1 \rightarrow Z_q^* \tag{3}$$

$$H_2 : \{0,1\}^* \times G_1 \rightarrow G_1 \tag{4}$$

The PKG randomly selects $s \in Z_q^*$ as the master key of system, computes its public key $P_{pub} = s \cdot P$, and publishes the system parameters $\{G_1, G_2, \hat{e}, q, P, P_{pub}, H_1, H_2\}$ offline.

User key extraction. The second phase is key extraction. Here, the ID of a node is used as the public key of the node. The signature node randomly selects a number $t \in Z_q^*$ as partial private key of itself, then computes $t \cdot P$, submits the ID of itself and $t \cdot P$ to the PKG over a secure channel, and requests another part of private key to the PKG. Upon confirming the identity of the requesting node, the PKG computes an S_{ID} as equation (5), which is used as another part of private key of the requesting node and sends S_{ID} to the node over a secure channel.

$$S_{ID} = s \cdot Q_{ID} = s \cdot H_2(ID, t \cdot P) \in G_1 \tag{5}$$

Signature. The third phase is signature. Given the message M, the signature node computes u and v, as in equations (6) and (7), respectively, then sends $(u, v, t \cdot P)$ as signature information along with the message M to a receiving node

$$u = H_1(M, S_{ID}) \in Z_q^* \tag{6}$$

$$v = \frac{1}{t + u} \cdot S_{ID} \in G_1 \tag{7}$$

Verification. The fourth phase is signature verification. After receiving the signature information $(u, v, t \cdot P)$ and message M, the receiving node computes Q_{ID} given by equation (8) and verifies if equation (9) is true

$$Q_{ID} = H_2(ID, t \cdot P) \tag{8}$$

$$\hat{e}(v, t \cdot P + u \cdot P) = \hat{e}(Q_{ID}, P_{pub}) \tag{9}$$

If equation (9) is true, the receiving node successfully verifies the signature; otherwise, the signature is considered to be forged.

Validity of the signature scheme

Lemma 1. The signature scheme is verifiable.

Proof. From equations (5), (7), and formula $P_{pub} = s \cdot P$, we can get the formula given by equation (10)

$$\hat{e}(v, t \cdot P + u \cdot P) = \hat{e}(v, (t + u) \cdot P) = \hat{e}\left(\frac{1}{t + u} \cdot S_{ID}, (t + u) \cdot P\right) = \hat{e}(Q_{ID}, P_{pub}) \tag{10}$$

So, the following three points are confirmed:

1. The S_{ID} is issued by PKG since only the PKG knows the master key s.

Table 2. Contrast of sign efficiency.

Scheme	Signature	Verification
Chen et al.[9]	$3M + 1H_1 + 1H_2 + 1S$	$1M + 2H_1 + 1H_2 + 3S + 4P$
This scheme	$1M + 1H_1 + 1IN$	$1M + 1H_2 + 1S + 2P$

Table 3. Operation symbols in Table 2.

P: bilinear map ê	M: point multiplication operation of group G_1
H_1: hash operation to Z_q^*	S: addition operation of group G_1
H_2: hash operation to G_1	IN: inverse operation of Z_q^*

Table 4. Contrast of operation time(s).

Length of key	Signature		Verification	
	Chen et al.[9]	This scheme	Chen et al.[9]	This scheme
163/1024	0.178	0.167	0.37	0.233
233/2240	0.404	0.362	0.77	0.554
283/3072	0.552	0.626	1.255	0.895
409/7680	1.093	1.248	2.664	1.791

2. The partial private key t of the signature node is registered with PKG.
3. The signature node has proved that it does know the partial private key without exposing t.

Efficiency analysis

The efficiency contrast between the signature scheme and the scheme in Chen et al.[9] is shown in Table 2.

The operation symbols used in Table 2 are explained in Table 3.

Two signature algorithms are implemented based on C++ on a PC (Windows 10, Intel Core™ CPU 2.20 GHz, 4 GB memory), and the contrast of operation time of two schemes are shown in Table 4.

From Tables 3 and 4, it can be seen that, compared with Chen et al.,[9] our algorithm reduces computational complexity and is more efficient whether in signature or verification process, so applicable for resource-constrained UANs.

Design of secure anonymous routing scheme

Strong anonymity and two-way authentication are the main objectives of secure anonymous routing scheme. Two-way authentication of communication parties is realized by aforementioned short signature algorithm. The identity anonymity of communication nodes to intermediate nodes in the routing path is implemented through a trap-door design, and the identity anonymity of intermediate nodes is realized by encoding their SID. Moreover, in a routing request (RREQ; response) message, a random number is introduced in the fields of hops, which is used by the source (destination) node to protect its location privacy. The secure anonymous routing scheme is shown in Figure 2.

Trap-door design and anonymity realization of communication nodes

In our secure routing scheme, the anonymity of communication nodes is realized by constructing trap door in routing messages. Trap door is a special token attached in packets, which is constructed using encryption function. Only the designated node can open the trap door, other nodes can neither open the trap door nor know which node can open it. In RREQ or response messages, the IDs of source and destination nodes are replaced by trap door, and only the destination node can open it, so the IDs of the node pair which are communicating are anonymous to other nodes. The anonymity of forwarding nodes is realized through encoding their SIDs.

There are usually two ways to construct trap door. One is based on PKI and asymmetric encryption. A source node uses the public key of the destination node to construct trap door, thus a trusted third party is required, but it is impractical for UANs. In the second way, each pair of source and destination nodes shares a secret key. With the increase in network size, each node has to maintain more and more shared keys, consequently increases memory cost and reduces the scalability of the network.

In this article, we use bilinear pairing to construct the shared key of trap door, and the procedure is detailed as follows:

1. In the initialization phase of section "Scheme design," the PKG defines the third hash function $H_3 : G_2 \rightarrow \{0, 1\}^{\beta}$, where β is an integer constant, and issues H_3 with other system parameters.
2. In the phase of user key extraction, the PKG computes $s \cdot H_2(ID_x, P)$ and sends it to the signature node through a secure channel.
3. The source (signature) node computes the shared key $K_{S,D}$ according to equation (11), constructs the trap door as equation (12), and sends a RREQ message to the destination node. The ID of the source node is denoted by ID_S and the one of the destination node is ID_D

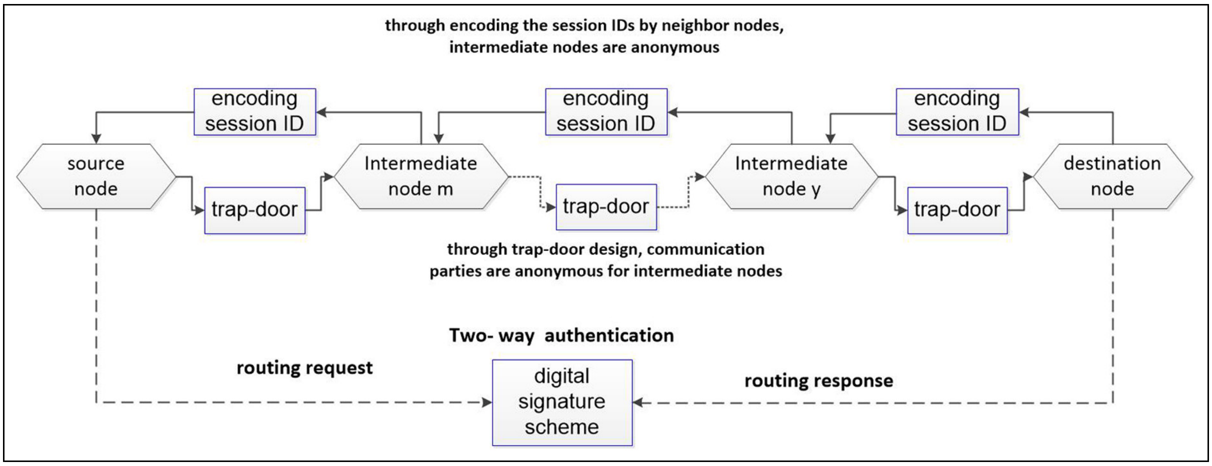

Figure 2. Design of secure routing scheme.

$$K_{S,D} = H_3(\hat{e}(S_{ID_s}, H_2(ID_D, P))) \tag{11}$$

$$<Q_{ID_S}, K_{S,D}(ID_s||ID_D)> \tag{12}$$

In equation (12), $ID_S||ID_D$ represents the tandem connection of string ID_S and ID_D, $K_{S,D}(ID_S||ID_D)$ denotes encrypting $ID_S||ID_D$ using the shared key $K_{S,D}$, Q_{ID_S} is transmitted to other nodes for computing the shared key.

4. The receiving node, whose ID is denoted by ID_X, computes the shared key according to equation (13) and tries to open the trap door

$$K_{S,X} = H_3(\hat{e}(Q_{ID_s}, s \cdot H_2(ID_X, P))) \tag{13}$$

Lemma 2. If the ID_X is equal to the decrypted ID_D, then the receiving node is considered to be destination node.

Proof. If the ID_X is equal to the decrypted ID_D, according to the bilinear property of $\hat{e}$ and $S_{ID} = s \cdot Q_{ID}$ as equation (5), we can get

$$\begin{aligned} K_{S,D} &= H_3(\hat{e}(S_{ID_S}, H_2(ID_D, P))) \\ &= H_3(\hat{e}(s \cdot Q_{ID_S}, H_2(ID_D, P))) \\ &= H_3(\hat{e}(Q_{ID_S}, s \cdot H_2(ID_D, P))) \\ &= H_3(\hat{e}(Q_{ID_S}, s \cdot H_2(ID_X, P))) = K_{S,X} \end{aligned}$$

so the trap door opened by receiving node is $ID_S||ID_D = ID_S||ID_X$.

RREQ and identity authentication

The self-organized characteristic makes UANs lack an authoritative certification party to verify the identity of each node, which can be exploited by malicious nodes to steal information or attack network deliberately. In UANs, the information such as the identity, location, network topology, and moving mode are more sensitive. So, two-way authentication of node pair of source and destination is vital in some situations. In this article, identity authentication is based on the short signature scheme presented in section "Design of digital short signature."

When a source node S has a message to transmit, it searches for the route to the destination node D in its route cache table. If node S fails to find the route, it starts a route discovery process. The node S can obtain the system parameters $\{G_1, G_2, \hat{e}, q, P, P_{pub}, H_1, H_2, H_3\}$, $s \cdot H_2(ID_S, P)$ and its partial private key S_{ID_s} from the PKG offline in advance, and another part of private key $t \in Z_q^*$ is held by itself. In the route discovery process, the node S constructs and broadcasts a RREQ packet as follows

$$\left\langle HEAD, SID_S, hops, \left(SID_{\_seq_{S \to D}}\right)\right\rangle$$

$$HEAD = RREQ, seq, Q_{ID_S}, K_{s,D}(ID_S||ID_D), t \cdot P, u, v, KHQS$$

$$KHQS = K_{S,D}(Hash(Q_{ID_S}, K_{S,D}(ID_S||ID_D), t \cdot P, u, v))$$

$$u = H_1(ID_S||ID_D, S_{ID_s})$$

$$v = \frac{1}{t+u} \cdot S_{ID_s} \in G_1$$

In the head of RREQ, seq is the sequence number of route discovery, which can be generated by $Hash(time() + ID_S)$. S_{ID_S} is partial private key of node S generated by the PKG, and SID_S is the SID of node S. The $hops$ field is filled with hop count, and it is initialized to a random number by the source node, so other nodes are unable to deduce the location of the source node according to the hop count, so the privacy of location and topology is guarded. The field of

$SID_{\_seq_{S\rightarrow D}}$ is filled with the encoded IDs of all the routing nodes from source to destination and is also initialized to a random number by the source node, so the privacy of identities and locations of forwarding nodes is guarded too. *KHQS* is used to verify the integrity of trap door and signature information by the destination node. In the head of RREQ messages, the trap door $Q_{ID_S}, K_{S,D}(ID_S||ID_D)$ is used to achieve anonymity of source–destination nodes to other nodes. Except trap door, the information of signature $t \cdot P, u, v$ of the source node is included in the head of RREQ message. After the destination node opens the trap door successfully, it computes whether equation (14) is true based on Q_{ID_S} and the information of signature $t \cdot P, u, v$ and further authenticates the identity of the source node

$$\hat{e}(v, t \cdot P + u \cdot P) = \hat{e}\left(Q_{ID_s}, P_{pub}\right) \tag{14}$$

Before transmitting a RREQ message, the source node S adds a record in its routing cache as follows

$$\left(seq, ID_D, init\_hops, init\_SID_{\_SEQ_{S\rightarrow D}}, SID_S, K_{S,D}, ?\right)$$

in which the last field is used to record the route to the destination node D. After transmitting the RREQ message, the node S starts a timer for the message. If the node S fails to receive a routing response (RREP) message from node D after the timer expires, the node S starts a new round of RREQ.

Anonymous design of intermediate nodes

Besides the anonymity of source and destination nodes, the anonymity of intermediate nodes in the routing path is also realized in our secure source routing protocol. Inspired by the idea of MPLS, each forwarding node in our protocol creates a SID for each session. In order to achieve anonymity of intermediate nodes, the SIDs in $SID_{\_seq_{S\rightarrow D}}$ of RREQ and RREP messages are encoded. So, an overhearing node can neither track the flow of data nor determine whether the neighbor is the destination node. The only information it can obtain is the SIDs of neighbors on the previous hop or next hop.

Each node in our protocol needs to maintain three tables. The first is the SID table for neighbor nodes shown as Table 5, the second is the SID table for itself shown as Table 6, and the third is the encoding length table shown as Table 7. To facilitate better understanding, we use the topology shown in Figure 3.

In Table 7, the "start" column denotes the start value from which the node encodes the SIDs of upstream neighbors, and the "bits" column denotes the number of bits of encoded SID for neighbor nodes. Different SIDs of local node can use different start values and number of bits for encoding. Using the topology shown in Figure 3, after receiving the RREQ message forwarded by node M for the first time, node X takes out the SID of node M, SID_M from the message, then adds a record in Table 5, generates a local SID SID_{X_2} for this session and adds it in Table 6, then adds a record in Table 7, and sets the "start" and "bits" columns. So, according to value 5 of the "index" field in Table 5, the value 0 of the "start" field, the value 3 of the "bits" field, node X encodes SID_M as "101" and appends the encoded SID "101" into $SID_{\_seq_{S\rightarrow D}}$ field of the RREQ message. When the message finally arrives at node D, the $SID_{\_seq_{S\rightarrow D}}$ field is filled with label source route including all the encoded SIDs of intermediate nodes in routing path, and the anonymity of intermediate nodes is achieved by this way. Meanwhile, short label with fixed length reduces the length of packets and improves protocol efficiency.

Table 5. Session ID table for neighbor nodes of node X.

Index	Session ID (SID)
5	SID_M
7	SID_Y

Table 6. Local session ID table of node X.

seq	Local session ID (SID)
1	SID_{X_1}
2	SID_{X_2}

Table 7. Encoding length table of node X.

Local session ID (SID)	*start*	*bits*
SID_{X_1}	0	4
SID_{X_2}	0	3

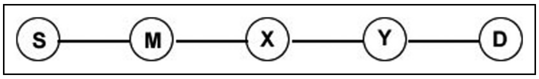

Figure 3. Routing topology.

RREP and location anonymity

Upon receiving a RREQ message, the node D computes the shared key $K_{S,D} = H_3(\hat{e}(Q_{ID_S}, s \cdot H_2(ID_D, P)))$ using Q_{ID_S} in the RREQ message, decrypts, and opens the trap door $K_{S,D}(ID_S||ID_D)$ in the message using $K_{S,D}$. If the decrypted ID_D is the same as the ID of itself, the receiving node is determined to be the destination node. Furthermore, the node computes $KHQS = K_{S,D}(Hash(Q_{ID_S}, K_{S,D}(ID_S||ID_D), t \cdot P, u, v))$ and compares it with the *KHQS* in the message. If they are the same, the message is considered to be valid. Otherwise, it is considered to be invalid and is dropped. After the RREQ message is verified to be valid, the destination node begins to authenticate the source node and then starts a RREP process.

Before generating a RREP message, the destination node first generates a session ID, SID_D, for this response, then fills the RREP message with $(seq, ID_S, init\_hops, init\_SID\_seq_{D \to S} SID_D, K_{D,S}, ?)$. In order to guard the privacy of location, $SID\_seq_{D \to S}$ is initialized to a random number by the destination node. Even the neighbor nodes of the destination node are unable to guess the hop counts to the destination node. In RREP message, the *init_hops* field is filled with the *hops* in original RREQ message plus a random number, and the node D also keeps the random number in its routing table. The value in *seq* field is a copy of *seq* in corresponding RREQ message. The RREP message is broadcasted to neighbor nodes, and the node Y in Figure 3 will receive the RREP message from node D as follows

$$\begin{aligned}
&\mathrm{HEAD}', SID_Y, SID\_seq_{S \to D}, SID_D, SID\_seq_{D \to S}\\
&\mathrm{HEAD}' = RREP, seq, Q_{ID_D}, K_{D,S}(ID_D \| ID_S),\\
&t_d \cdot P, u_d, V_d, KHQS'\\
&\mathrm{KHQS}' = K_{D,S}(Hash(Q_{ID_D}, K_{D,S}(ID_D \| ID_S), t_d \cdot P, u_d, V_d))\\
&u_d = H_1(ID_D \| ID_S, S_{ID_D})\\
&V_d = \frac{1}{t_d + u_d} \cdot S_{ID_D} \in G_1
\end{aligned}$$

In our protocol, the RREP message is processed according to the flowchart shown in Figure 4. From Figure 4, we can see that node S authenticates the identity of node D by the same way as section "Routing request and identity authentication," so two-way authentication is realized. Finally, node S records the route to node D in its routing table. The format of routing table is shown in Figure 5.

Protocol analysis

Security analysis

1. The attacker is difficult to forge the identities of other nodes.

From section "Design of digital short signature," we can see that S_{ID} is used in the signature process, $S_{ID} = s \cdot Q_{ID} = s \cdot H_2(ID, t \cdot P) \in G_1$. Here, s is held by the PKG, t is held by the node itself, and the ID of the node needs to be verified by the PKG. Based on the intractable DLP problem, given P, $t \cdot P$, and $s \cdot P$, a single attacker is unable to obtain t and s by calculation, as a result it cannot forge signature.

2. If an untrusted PKG forges the signature of other nodes, it can be exposed.

Lemma 3. The PKG can select randomly a number $t' \in Z_q^*$, compute $S'_{ID} = s \cdot Q'_{ID} = s \cdot H_2(ID, t' \cdot P)$, and forge the signature $(u', v', t' \cdot P)$ of the node whose identity is ID, in which $u' = H_1(M, S'_{ID}) \in Z_q^*$ and v' $ls(1/(t' + u')) \cdot S'_{ID}$; however, the signature $(u', v', t' \cdot P)$ can be determined as forge signature by the complainant node and an arbitrator.

Proof.

1. The complainant node sends $t \cdot P$ to an arbitrator;
2. The arbitrator selects randomly $\alpha \in Z_q^*$ and sends $\alpha \cdot P$ to the complainant node;
3. The node computes $\hat{e}(S_{ID}, \alpha \cdot P)$ and sends it to the arbitrator;
4. If $S_{ID} = s \cdot Q_{ID} = s \cdot H_2(ID, t \cdot P)$, it can be determined that the ID and $t \cdot P$ are registered with the PKG since only the PKG knows the master key s. So, we have equation (15)

$$\begin{aligned}
\hat{e}(S_{ID}, \alpha \cdot P) &= \hat{e}(s \cdot H_2(ID, t \cdot P), P)^{\alpha}\\
&= \hat{e}\big(H_2(ID, t \cdot P), P_{pub}\big)^{\alpha}
\end{aligned} \tag{15}$$

The arbitrator computes if $\hat{e}(S_{ID}, \alpha \cdot P)$ is equal to $\hat{e}(H_2(ID, t \cdot P), P_{pub})^{\alpha}$, and if it is true, the arbitrator can determine that $(u', v', t' \cdot P)$ is a forged signature.

3. Multiple attackers are difficult to collude with each other to forge node identity.

Considering the worst case, the PKG is involved in the conspiracy attack, namely, the S_{ID} has been leaked. Given two integers, k and $t \in Z_q^*$, then the collusion attack algorithm with k traitors (k-CAA) is defined as: Known each item in formula given by (16), calculate the formula given by equation (17)[18]

$$\begin{aligned}
&\left\{P, t \cdot P, S_{ID}, H_1(M_1, S_{ID}), \ldots, H_1(M_k, S_{ID}), \frac{S_{ID}}{t + H_1(M_1, S_{ID})},\right.\\
&\left.\qquad \frac{S_{ID}}{t + H_1(M_2, S_{ID})}, \ldots, \frac{S_{ID}}{t + H_1(M_k, S_{ID})}\right\}
\end{aligned} \tag{16}$$

$$\begin{aligned}
&\frac{S_{ID}}{t + H_1(M, S_{ID})}, H_1(M, S_{ID}) \notin\\
&\{H_1(M_1, S_{ID}), \ldots, H_1(M_k, S_{ID})\}
\end{aligned} \tag{17}$$

Here, we define the k–w CDH problem as follows: given the values of $k + 1$ items in $P, yP, y^2P, \ldots, y^kP$, then calculate P/y. In order to give more accurate evaluation of the proposed signature scheme, here we introduce $K + 1$ exponent problem ($k + 1EP$): Given

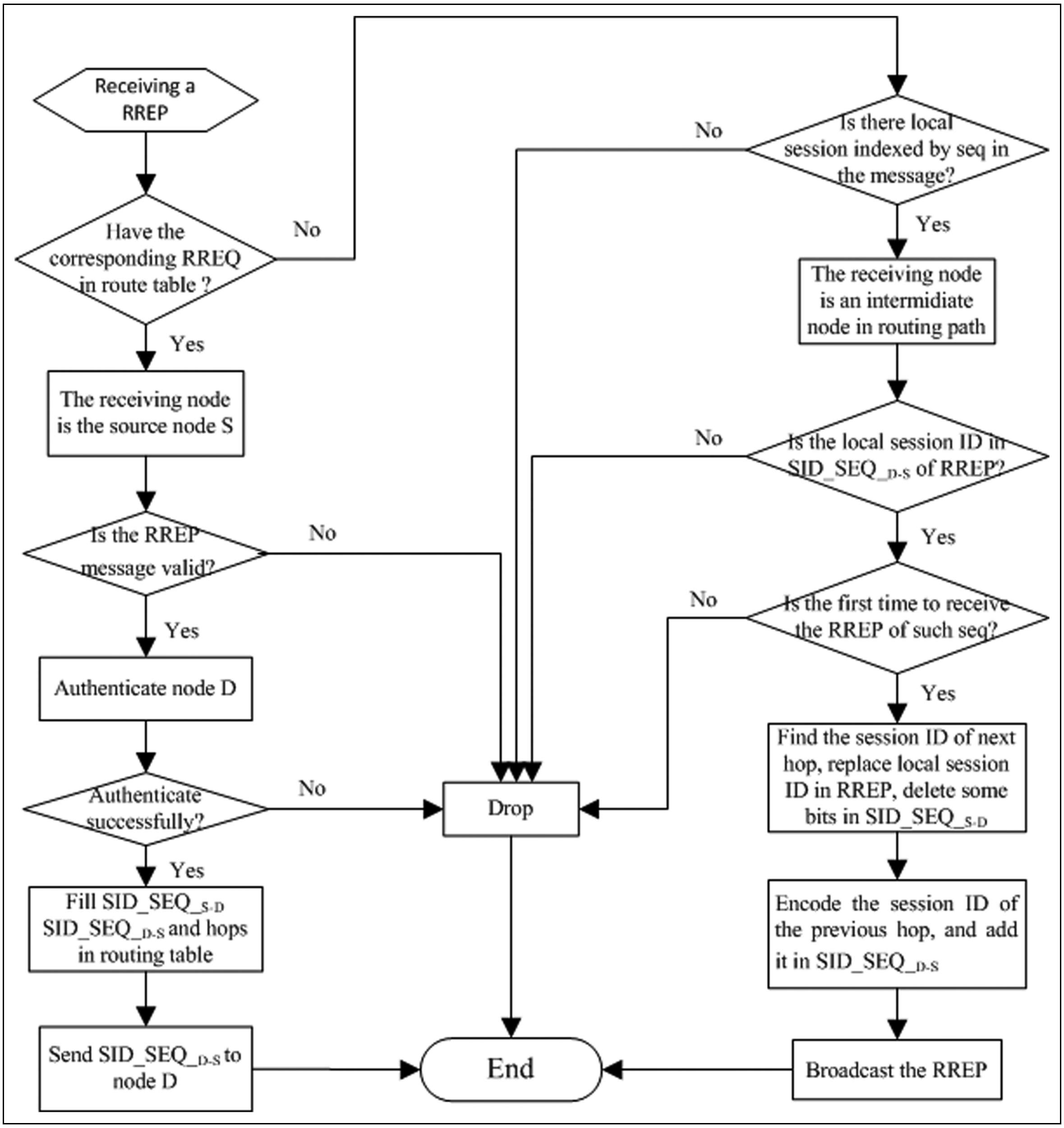
Receiving a RREP
Have the corresponding RREQ in route table ?
No
Yes
The receiving node is the source node S
Is the RREP message valid?
No
Yes
Authenticate node D
Authenticate successfully?
No
Yes
Fill SID_SEQ_S-D SID_SEQ_D-S and hops in routing table
Send SID_SEQ_D-S to node D
Drop
End
Is there local session indexed by seq in the message?
No
Yes
The receiving node is an intermidiate node in routing path
Is the local session ID in SID_SEQ_D-S of RREP?
No
Yes
Is the first time to receive the RREP of such seq?
No
Yes
Find the session ID of next hop, replace local session ID in RREP, delete some bits in SID_SEQ_S-D
Encode the session ID of the previous hop, and add it in SID_SEQ_D-S
Broadcast the RREP

Figure 4. Processing flowchart of RREP message.

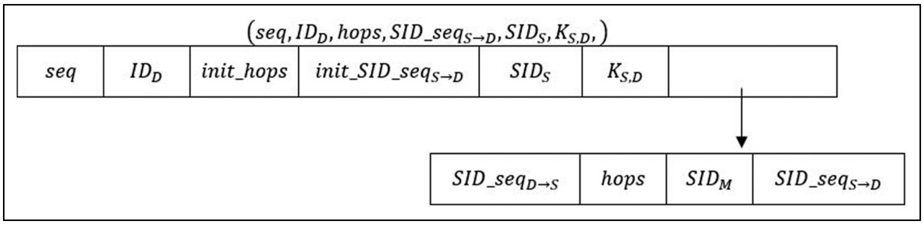

Figure 5. The format of routing table.

the values of $k+1$ items in $P, yP, y^2P, \ldots, y^kP$, calculate $y^{k+1}P$.

Now, we can prove k–w CDH problem is equivalent to the $k+1EP$ problem in polynomial time as follows.

Proof. Given the values of $k + 1$ items in $P, yP, y^2P, \ldots, y^kP$, let $t = y^{-1}$ and $Q = y^kP$, then $t^{-1}Q = y^{k+1}P$, $tQ = y^{k-1}P$, $t^2Q = y^{k-2}P, \ldots, t^{k-1}Q = yP$, $t^kQ = P$, so the input of k–w CDH problem turns into $t^kQ, t^{k-1}Q, t^{k-2}Q, \ldots, t^1Q, Q$ and the output turns into calculating $t^{k+1}Q$. So, k–w CDH problem is equivalent to the $k + 1EP$ problem in polynomial time.

As in Mitsunari et al.,[18] the k-CAA problem can be proved equivalent to $k - 1$–w CDH problem in polynomial time. Based on above analysis, intractable k-CAA problem is equivalent to k exponent problem (kEP) problem in polynomial time. So, statistically, there is no algorithm to solve the kEP problem in polynomial time, and the signature scheme is secure under random oracle model.

Anonymity analysis

From the trap-door scheme presented in section "Design of secure anonymous routing scheme," we can see that the IDs of source and destination nodes are encrypted in either route request or response message. The encryption key is calculated by bilinear mapping, and only the source and destination nodes can reach a common key and open the trap door. Other nodes can neither open the trap door nor be aware of which node can open it, so our protocol can realize the strong anonymity of communication nodes.

In our protocol, the forwarding nodes are identified by their SIDs, and the SIDs are encoded by neighbor nodes. Each node can encode the SIDs of its neighbor with different lengths. The encoded IDs are concatenated into routing sequence $SID\_seq_{D\rightarrow S}$ and $SID\_seq_{S\rightarrow D}$. The intermediate nodes can only get the SID of the previous node, rather than of all the forwarding nodes. The SID has nothing to do with its real identification and location. For a node, different sessions correspond to different SIDs, and attack nodes are unable to track data flow by analyzing packets, so secure communication is achieved. Moreover, the source or destination node is unable to obtain the real identification of intermediate nodes from the route label, so our protocol achieves anonymity between communication nodes and intermediate nodes. By introducing random numbers in fields of $hops$, $init\_SID\_seq_{D\rightarrow S}$ and $init\_SID\_seq_{S\rightarrow D}$, the location privacy of communication nodes is achieved.

Computation overhead analysis

Overhead for anonymity of forwarding nodes. The anonymity design of forwarding nodes is inspired by the idea of MPLS. The label routing sequences $SID\_seq_{D\rightarrow S}$ and $SID\_seq_{S\rightarrow D}$ are generated by concatenating the encoded SIDs of nodes on routing path. Different SIDs are encoded by different neighbor nodes with different lengths, and the coding has only local significance, so the anonymity of forwarding nodes to communication nodes is achieved. The encoding procedure of SID presented in section "Design of secure anonymous routing scheme" does not involve any complex operation, and the anonymity design of forwarding nodes is low overhead and can be applied to UANs.

Overhead for anonymity of communication nodes. The anonymity of communication nodes to intermediate nodes is realized by trap-door design, and the computation overhead is mainly introduced by the following calculation:

The source node

$$K_{S,D} = H_3(\hat{e}(S_{ID_s}, H_2(ID_D, P)))$$
$$K_{S,D}(ID_S||ID_D)$$

The destination node

$$K_{S,M} = H_3(\hat{e}(Q_{ID_s}, s \cdot H_2(ID_M, P)))$$
$$K_{S,M}(ID_S||ID_M)$$

We can see that trap-door construction requires two hash operations, one bilinear pairing map, and one symmetric encryption operation, while trap-door opening requires one hash operation, one bilinear pairing map, and one symmetric encryption operation. The overhead for anonymity of communication node is irrelevant to the number of nodes on routing path, which can be applied to large-scale UANs.

Overhead for identity authentication of communication nodes. The identity authentication of communication nodes includes two steps, signature and verification. The signature procedure involves one point multiplication, one hash operation, and one inverse operation. By contrast, verification involves one point multiplication, one hash operation, one addition operation, and two bilinear pairing map operation. Compared with the signature scheme in Chen et al.,[9] in which the verification process involved four bilinear pairing operations, the identity authentication of our protocol has low computational complexity and can be applied to resource-constrained UANs.

Experiment and conclusion

Experiment

In this section, we evaluate the performance of our routing scheme by simulation experiments. All simulations are performed using the network simulator (NS2) with an underwater sensor network simulation package AquaSim. We randomly deploy 49, 64, 81, and 100 sensor nodes successively in a three-dimensional (3D) region

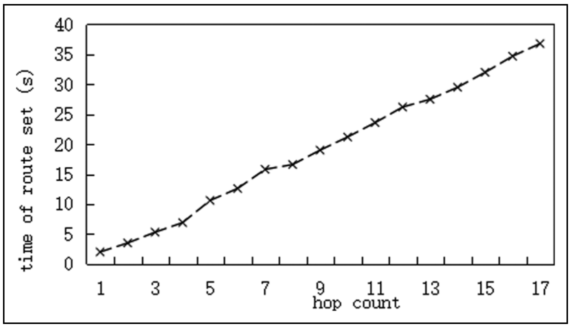

Figure 6. Route setup time versus hop count.

Table 8. Simulation parameters.

Parameter	Value
Data generation rate	80 bits/s
Packet load, *l*	100 bytes
Bandwidth	5 kbps
Traffic	CBR
Transmission range	1500 m
MAC protocol	Slotted FAMA

MAC: medium access control.

of 9000 m × 9000 m × 4000 m, and a stationary sink node is randomly deployed on the water surface. The simulation parameters are shown in Table 8. After numerous experiments, we vary the simulation scenario by deploying the nodes evenly in the region. Figure 6 shows the experiment results of route setup time and hop count.

To the best of our knowledge, our secure anonymous routing scheme is the first source routing protocol for UANs so far, and route setup time is one of the main performance criteria for source routing. From Figure 6, we can observe the average route setup time increases with hop counts, which is understandable. A route can be setup only after the source node sends a RREQ and receives the RREP from the destination node. For RREQ and RREP messages, the more the hop counts, the longer the end-to-end delay. The aforementioned secure anonymous routing protocol has no asymmetric encryption and decryption operations. The signature scheme of our secure routing protocol is based on node identity. The source node does not have to apply for the public key of the destination node to PKG, which avoids additional delay, bandwidth, and energy consumption for finding route to the PKG. So, the proposed routing scheme improves the performance and scalability of UANs and is particularly suitable for UANs characterized by high bit error, long propagation delay, and narrow bandwidth.

Furthermore, we conducted a number of experiments to compare the performance of our scheme with geographic information routing protocol based on partial network coding (GPNC) in Hao et al.[19] and level-based adaptive geo-routing (LB-AGR) in Du et al.[20] in terms of energy consumption, throughput, and packet delivery rate as in Figure 7.

From Figure 7(a), we can see that the total energy consumption of all the three protocols increases with the number of nodes, and the total energy consumption in our protocol is slightly higher than LB-AGR and GPNC, which is reasonable. The more nodes participated in routing and forwarding, the more energy consumed. Since our scheme realizes authentication and anonymity, the implementations of signing, verifying, encrypting, and encoding introduce extra energy consumption. Moreover, our scheme is essentially an on-demand routing, in which RREQ messages need to be flooded across the whole network during the routing discovery process and consume more energy.

From Figure 7(b), we can see that the throughput of all the three protocols increases with the number of nodes, and the throughput of our scheme is slightly lower than LB-AGR, but higher than GPNC. GPNC uses broadcast routing. Broadcast consumes large bandwidth and is congestion-prone in UANs with limited bandwidth. When the nodes in network outnumber a threshold, the broadcast packets from each node bring about the rapid consumption of bandwidth and network congestion problems. So, as shown in Figure 7(b), the throughput of GPNC with broadcast routing is lower than both our scheme and LB-AGR. Figure 7(b) shows that network congestion occurs when the number of nodes is more than 49, and the throughput starts to decrease in networks using GPNC routing protocol. Both our scheme and LB-AGR use unicast routing. However, our scheme is an on-demand routing and the RREQ messages flooded across the whole network consume more bandwidth, so the throughput of our scheme is lesser than LB-AGR.

Figure 7(c) shows that the packet delivery rate increases with the number of nodes in both our scheme and LB-AGR. However, for GPNC, the packet delivery rate increases at first with the number of nodes, then decreases quickly. When the number of nodes is lesser than 49, there is no congestion in network regardless of which scheme, and GPNC achieves the highest packet delivery rate using broadcast routing. When the number of nodes is more than 49, GPNC results in network congestion and the packet delivery rate decreases quickly. Since our scheme is an on-demand routing and the routing path is usually updated timely than LB-AGR using periodic updating, the packet delivery rate of our scheme is always higher than LB-AGR as in Figure 7(c).

Conclusion

Without a trusted third party, the presented digital signature technology solves the key escrow problem of

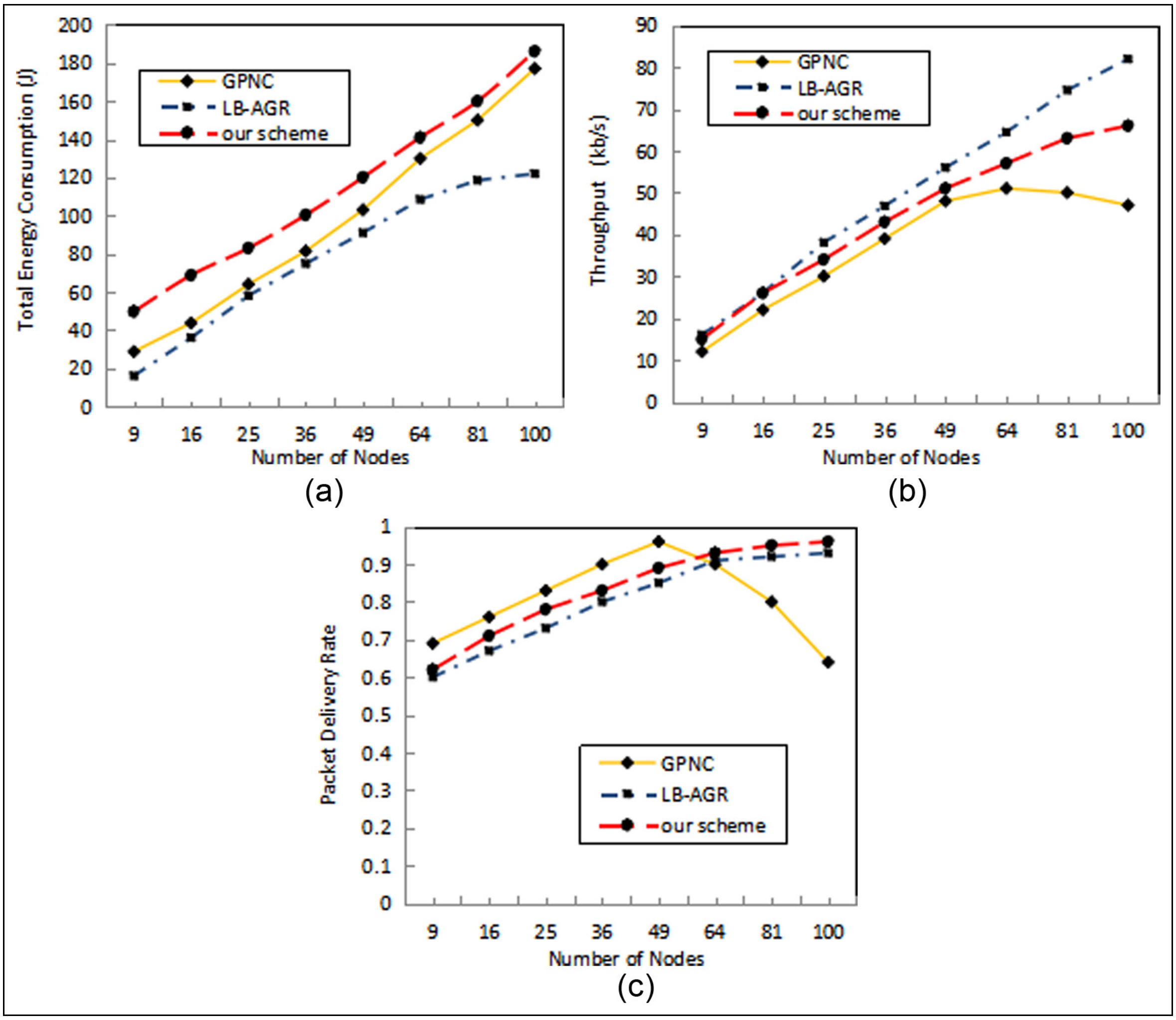

Figure 7. Performance comparison: (a) energy consumption, (b) throughput, and (c) packet delivery rate.

UANs, improves communication efficiency, and avoids the third party PKG from colluding with other nodes for forgery attack. Only in the initializing phase, the source node applies for a partial private key to the third party PKG, avoiding the intractable problem of real-time PKG in UANs. By the designs of digital signature and bilinear map trap door, the routing scheme achieves strong anonymity and two-way authentication of source and destination nodes, avoids identity deception between the source and destination nodes, and provides security for UANs communication. Moreover, the trap-door design in our protocol avoids overheads of maintaining a large number of pre-shared keys, and opening trap door only includes one hash operation and one bilinear mapping operation, which is feasible in UANs.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the National Natural Science Foundation Projects of China (61162003), Qinghai Office of Science and Technology (2015-ZJ-904, 2015-ZJ-718 and 2017-Z-Y21), and Hebei Engineering Technology Research Center for IOT Data Acquisition & Processing.

References

1. Petillo S, Schmidt H and Balasuriya A. Constructing a distributed AUV network for underwater plume-tracking operations. *Int J Distrib Sens N* 2012; 8(1): 1–12.
2. Tran TM and Oh SH. UWSNs: a round-based clustering scheme for data redundancy resolve. *Int J Distrib Sens N* 2014; 10(4): 1–6.
3. Baek CU and Jung JW. High throughput receiver structure for underwater communication. *Int J Distrib Sens N* 2015; 11(11): 1–6.
4. Babiker AE, Zakaria MNB, Yosif H, et al. An efficient energy adaptive hybrid error correction technique for

underwater wireless sensor networks. *World Acad Sci Eng Tech* 2011; 2011(51): 1389–1395.

5. Dong Y and Liu P. Security consideration of underwater acoustic networks. In: *Proceedings of the 20th international congress on acoustics*, Sydney, NSW, Australia, 23–27 August 2010. PACS.
6. Cong Y, Yang G, Wei Z, et al. Security in underwater sensor network. In: *Proceedings of the 2010 international conference on communication and mobile computing*, Shenzhen, China, 12–14 April 2010. New York: IEEE.
7. Dini G and Lo Duca A. A cryptographic suite for underwater cooperative applications. In: *Proceedings of the 2011 IEEE symposium on computers and communications*, Corfu, 28 June–1 July 2011. New York: IEEE.
8. Peng C, Du X, Li K, et al. An ultra-lightweight encryption scheme in underwater acoustic networks. *J Sensors* 2016; 2016(3): 8763528 (10 pp.).
9. Chen X, Zhang F, Konidala DM, et al. New ID-based threshold signature scheme from bilinear pairings. In: *Proceedings of the international conference on cryptology*, Kolkata, India, 11–14 December 2004. Berlin: Springer.
10. Zhang Y, Fan Z and He X. Anonymous security of multipath routing in mobile ad hoc networks. *Chinese J Electron* 2005; 33(11): 2022–2030.
11. Leinmuller T, Maihofer C, Schoch E, et al. Improved security in geographic ad hoc routing through autonomous position verification. In: *Proceedings of the 3rd ACM international workshop on vehicular ad hoc networks (VANET)*, Los Angeles, CA, 29 September 2006. New York: ACM.
12. Ji EK, Yun NY, Muminov S, et al. Security in underwater acoustic sensor network: focus on suitable encryption mechanisms. *Comm Com Inf Sc* 2012; 2012(324): 160–168.
13. Zhang R and Zhang Y. Wormhole-resilient secure neighbor discovery in underwater acoustic networks. In: *Proceedings of the 29th IEEE international conference on computer communications (INFOCOM)*, San Diego, CA, 14–19 March 2010. New York: IEEE.
14. Wang W, Kong J, Bhargava B, et al. Visualisation of wormholes in underwater sensor networks: a distributed approach. *Int J Secur Network* 2008; 3(1): 10–23.
15. Luo Y, Pu L, Peng Z, et al. RSS-based secret key generation in underwater acoustic networks: advantages, challenges and performance improvements. *IEEE Commun Mag* 2016; 54(2): 32–38.
16. Liu Y, Jing J and Yang J. Secure underwater acoustic communication based on a robust key generation scheme. In: *Proceedings of the 9th international conference on signal processing*, Beijing, China, 26–29 October 2008. New York: IEEE.
17. Domingo MC. Securing underwater wireless communication networks. *IEEE Wirel Commun* 2011; 18(1): 22–28.
18. Mitsunari S, Sakai R and Kasahara M. A new traitor tracing. *IEICE T Fund Electr* 2002; 85(2): 481–484.
19. Hao K, Jin Z and Shen H. An efficient and reliable geographic routing protocol based on partial network coding for underwater sensor networks. *Sensors* 2015; 15(5): 12720–12735.
20. Du X, Huang K, Lan S, et al. LB-AGR: level-based adaptive geo-routing for underwater sensor network. *J China Univ Post Telecommun* 2014; 21(1): 54–59.