Data Sci. Eng. (2016) 1(3):189–200
DOI 10.1007/s41019-016-0017-x

Provenance for Wireless Sensor Networks: A Survey

Changda Wang[1] · Wenyi Zheng[1] · Elisa Bertino[2]

Received: 8 August 2016 / Accepted: 10 August 2016 / Published online: 27 September 2016

Abstract In wireless sensor networks (WSNs), provenance records the data source, forwarding, and aggregating information of data packets on their way to the base station. Provenance is critical for assessing the trustworthiness of the received data, diagnosing network failures, detecting early signs of attacks, etc. However, because the provenance size expands rapidly with the increase in packet transmission hops, the provenance schemes developed for use in wired computer networks are not generally applicable to WSNs. Therefore, specific provenance techniques have been developed for WSNs that take into account the constrained resources of sensor nodes. In this paper, we survey such techniques. Special focus in the paper is devoted to a systematic and comprehensive classification of the solutions proposed in the literature. We review each solution by highlighting its pros and cons. Finally, we discuss recent trends in provenance encoding schemes for WSNs.

Keywords Data provenance · Data trustworthiness · Wireless sensor networks

✉ Changda Wang
changda@ujs.edu.cn

Wenyi Zheng
oraclecc@qq.com

Elisa Bertino
bertino@purdue.edu

[1] School of Computer Science and Communication Engineering, 301 Xuefu Road, Zhenjiang, Jiangsu, China

[2] Cyber Center and Computer Science Department, Purdue University, West Lafayette, IN 47907, USA

1 Introduction

A wireless sensor network (WSN) consists of a number of small and low-cost sensor nodes (nodes, for short). Each node basically has sensing, and data processing and communicating capabilities. WSNs play an important role in data acquisition and transmission for many application domains which use different types of sensors, such as magnetic, thermal, infrared, acoustic, and radar. Compared to the wired computer networks, the nodes of WSNs are often resource-tightened and deployed in unprotected physical environments. Moreover, communications in WSNs depend on multi-hop wireless signal relays. Because of such characteristics, data transmitted across WSNs can be easily tampered. As a result, in order to reliably use data collected from a WSN assessing the trustworthiness of the collected data is critical. Provenance is a key factor for assessing data trustworthiness as provenance records the history of data acquisition and ownership, and the actions performed on the data while being processed and transmitted across the WSN.

The notion of provenance had been originally introduced to document the origin, history, chain of custody, derivation, or process of art objects. When we use provenance in the field of information technology, the dual of the art object is the data. Usually, when data item is processed and transmitted across large-scale systems, the provenance size may largely exceeds the size of the data themselves; for example, Jayapandian et al. report that in the MiMI system, for data with size of 270 MB, the provenance for the data is approximately equal to 6 GB [12]. Therefore, in order to limit the provenance size, different provenance systems retain only certain provenance information. In the context of WSNs, the provenance of a data packet refers to where the packet is produced and how it is delivered, e.g., forwarded and/or aggregated, to the base station (BS) [14]. It is important to notice that in WSNs provenance is useful not only for data trustworthiness assessment, but also for network's troubleshooting.

In a multi-hop WSN, even if the provenance of an individual data packet (packet, for short) only records the trace of the packet, the provenance size rapidly increases with the increase in the number of packet transmission hops. Therefore, provenance schemes for wired computer networks, e.g., [9, 31, 32], are not generally applicable to WSNs. In order to address the limited processing capabilities and limited wireless communication bandwidth at each node, several compact, or lightweight, data provenance schemes have been proposed. The goal of this paper is to provide a deeper understanding of current provenance schemes in WSNs, and to identify open research issues that can be further pursued in this area.

The rest of the paper is organized as follows: Sect. 2 introduces the system model and basic concepts used in this paper. Section 3 classifies existing provenance schemes into five different categories and then reviews each of them. Section 4 summarizes security issues related to provenance encoding and transmission in WSNs. Section 5 discusses recent trends and research directions in provenance schemes for WSNs, and Sect. 6 concludes the paper.

2 Background

In this section, we introduce the WSN model we use throughout the discussion and the WSN provenance models considered in this paper. We also briefly discuss the main challenges in designing and implementing provenance schemes for WSNs.

2.1 Network Model for WSNs

The network model we consider in this paper is that of a multi-hop WSN, consisting of a number of nodes and a BS that collects data from the nodes by rounds.

A round is a time interval, in which the sensors attached to the nodes generate data and then transmit the data to the BS. The use of the rounds reduces the waking time for the nodes and therefore extends the batteries lifetime. For instance, using a TelosB node the average waking and sleeping power consumptions are 3 mW and 225 μW, respectively [29], i.e., in an awake node, even when not transmitting wireless signals, the energy consumption is about 13 times greater than that of a sleeping node.

Each node has a unique identifier n_{ID} and a symmetric cryptographic key k_{ID} assigned by the BS. The key k_{ID} is used to bind the data and its provenance as well as to encrypt the sensitive data. The BS is a node directly connected to a server, and therefore compared to the other nodes, it does not have constraints with respect to energy, storage space and computational capabilities.

Every node, except the BS, has three possible roles: *data source*, *data forwarder* and *data aggregator*. A data source acquires data through the sensors connected to the node and then sends the data in the form of a packet. A data forwarder relays the received packet toward the BS. A data aggregator aggregates two or more smaller packets into a new large packet and then sends the new packet toward the BS [23]. The most important advantage of packets aggregation is to save energy, i.e., when two or more packets are aggregated, the energy required for transmitting the aggregated packet is lower than the energy required for transmitting those packets independently [1]. Nowadays, most WSN transmission protocols support packet aggregation. Any node can be a data aggregator when the aggregation conditions hold during packet transmission [8].

Because packet path loops are prohibited by all practical WSN transmission protocols, formally the network model of a WSN is an acyclic directed graph $G(N, E)$, where $N = \{n_i | 1 \leq i \leq |N|\}$ is the set of nodes, and $E = \{e_{ij} | 1 \leq i, j \leq |N|\}$ is the set of directed edges between nodes. $|N|$ denotes the cardinality of set N and e_{ij} denotes the directed edge from n_i to n_j.

2.2 A Provenance Model for WSNs

The definition of data provenance varies with different application domains. In the context of WSNs, the provenance of a data item refers to where the item is produced and how it is delivered, e.g., forwarded and/or aggregated, to the BS [14]. Therefore, the formal model for the data provenance in WSNs is as follows [22, 27]:

For a WSN $G(N, E)$, let p be a packet delivered to the BS. The provenance of p is defined as a directed acyclic graph $T(V, E)_p$, where a vertex $v \in V$ is the ID of a node that has generated or forwarded or aggregated p with an unique sequence number seq. For simplicity, the notation (n_{ID}, seq) is used to represent v in the provenance of p. The notation Host(v) denotes the host node's ID of v, i.e., the first element of the pair representing v, that is, $v.n_{ID}$. An edge $e_{ij} \in E$ represents the one hop packet transmission from node Host(v_i) to node Host(v_j), where $v_i, v_j \in V$.

It is worth noting that such a definition is a *node-level provenance* which encodes the nodes involved at each step of data processing. Each node in the provenance graph represents a snapshot of a packet passing by a sensor node. As a result, each packet has an independent provenance graph. The mapping from any node in the provenance graphs to a node in the WSN is single-valued, whereas from a WSN node to a node in the provenance graphs the mapping is multi-valued. Moreover, as the provenance graphs are derived from the WSNs' topology graphs, they are acyclic directed graphs too.

Figure 1 [27] shows two different kinds of provenances; in the figure n_1 serves as the BS. In Fig. 1a, the data item is generated at leaf node n_3 and the nodes in the middle simply forward the data to the BS. Such a provenance is referred as a *simple* or *linear* provenance. In Fig. 1b, the data item is aggregated and forwarded toward the BS, where n_4 aggregates the packets from n_6 and n_7, and n_3 aggregates the packets from n_4 and n_8, respectively. Such a provenance is referred as an *aggregated* or *tree-like* provenance. Note that the aggregated provenance reuses some packet paths, and therefore the length of the aggregated provenance is shorter than the sum of the lengths of the provenances that one would have when transmitting those packets from each data source to the BS independently.

Since a packet usually keeps its sequence number seq in the packet's head, to save energy, most provenance schemes simply transmit node IDs in the provenance. At the BS, the provenance is obtained by combining the received node IDs and seq together.

2.3 Challenges

Although several data provenance techniques have been proposed over the years, research on the provenance techniques for WSNs is a relative new research topic. Existing provenance schemes developed for conventional wired networks cannot be applied to WSNs without being modified due to both the resource-tightened nature of WSNs and the rapid provenance size increase.

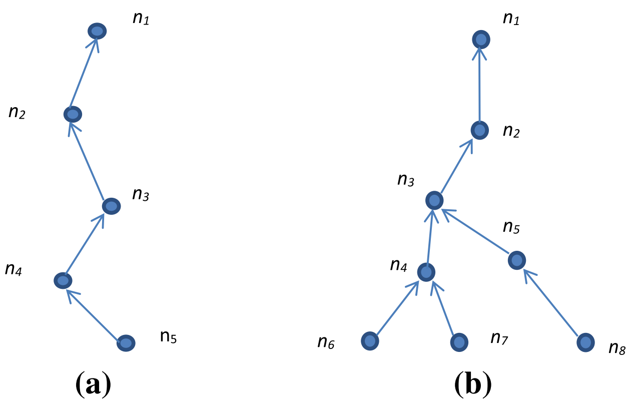

Fig. 1 Examples of provenance graphs, where n_1 represents the BS; **a** simple provenance; **b** aggregated provenance

These designs of provenance schemes for WSNs requires addressing several challenges.

(1) Large-scale and wireless signal relays. When using wireless signals, the relationship between the transmitted power P and the transmitted distance d is $P \propto d^n$, where $3 \leq n \leq 4$. Thus, doubling the transmission distance d requires increasing the transmission power from 8 to 16 times [29]. To save energy, the wireless transmission range of a node is from a few meters to a few hundred meters. Subsequently, even for monitoring areas of moderate size, large amounts of nodes are needed. As a result, packet transmission usually requires a large number of wireless signal relay hops, which increases the provenance's size.
Using the internet, to send a packet from Shanghai to Chicago that are at a distance of about 8,000 miles, only 11 hops are required on average, whereas sending a packet in a WSN to cross 1 mile may require from 10 to 20 hops when using Zigbee nodes.

(2) Limited or no infrastructure. In a WSN, no global IP address is available. Each node only has an ID assigned by the BS according to the transmission protocol adopted in the WSN. Nodes are stationary after deployment, but routing paths may change over time due to node failures, link quality degradation, and resource optimization [27]. Furthermore, nodes are deployed in an ad hoc or, rarely, in a pre-planned manner. Once deployed, the WSN is often left unattended to perform monitoring functions. Generally, a node ID does not contain any location information for the node; thus, network maintenance activities, such as managing connectivity and detecting failures, are difficult.

(3) Limited data processing abilities. Most nodes deployed in today WSNs have less than 10 KB memory, and an 8-bit or 16-bit processor with frequencies from 4 to 7.37 MHz. Generally, the power consumption of a desktop computer is between 200 and 300 W, whereas the power consumption for a TelosB node is only 3 mW [29]. In view of this, most provenance schemes developed for conventional computing systems cannot be applied to WSNs due to the limited computing capabilities of nodes.

3 Provenance Schemes

We classify the known provenance schemes for WSNs into the following categories: *elementary schemes*, *distributed schemes*, *block schemes*, *lossy compression schemes*, and *lossless compression schemes*. In what follows, we review each category and highlight its pros and cons.

3.1 Elementary Provenance Schemes

Elementary provenance schemes satisfy the basic provenance requirements for WSNs, but are not suitable for the resource-tightened nature of nodes.

3.1.1 SPS

In the SPS (Generic Secure Provenance) scheme [10], the provenance of a node with respect to a data item D_i is encoded as $P_i = \langle n_i, hash(D_i), C_i \rangle$, where n_i is the node ID; $hash(D_i)$ is a cryptographic hash of the data item D_i; $C_i = \{hash(n_i, hash(D_i) \| C_{i-1})\}_{k_i}$, i.e., C_i is a hash value signed by n_i with its encryption key k_i.

Under the SPS scheme, as each node in the packet path appends its provenance information to the current provenance, the size of the provenance increases linearly. Assume that the node ID is 2 bytes, if we use SHA-1 (160 bits) for the cryptographic hash operation and the TinyECC library [15] to generate the signature with the length of 160 bits, the provenance size's increase at each hop is of 42 bytes.

3.1.2 MP

In the MP (MAC based Provenance) scheme [22], a node uses its ID and a CBC-MAC (cipher block chaining message authentication code) together as the provenance. The CBC-MAC is a chain of blocks, except that the first block at the data source has an initial value, where every block's generation depends on its previous block and the current node's ID in the packet path. Because of such interdependence, any block change causes the final block to change in a way that cannot be predicted without knowing all the encryption keys used at the node where the change has occurred and at all the subsequent nodes.

Under the MP scheme, assume that the node ID is 2 bytes. When we use the TinySec library [13] to compute a 4-byte CBC-MAC, the provenance size increases by 6 bytes linearly at each hop.

3.1.3 Pros and Cons

To the best of our knowledge, the elementary provenance schemes are the simplest and easiest ones to implement in WSNs when compared to the other provenance schemes. Their disadvantage is that their average provenance size expand too fasts, even in middle-scale WSNs, thus making unaffordable the provenance transmission cost.

3.2 Distributed Provenance Schemes

Distributed provenance schemes use a distributed approach for storing provenance information on a series of nodes in a WSN. When the BS requires the provenance of a received data item, it has to send a query to the entire network and then retrieve the provenance by combining the responses from those nodes which have provenance information for the queried data item.

3.2.1 CAPTRA

CAPTRA (Coordinated Packet Traceback) [25] is a typical distributed provenance scheme. Under the CAPTRA scheme, when a node n_i sends a packet p to n_j, such information is not only recorded by n_i and n_j, but by the nearby nodes too. All nodes use the Bloom Filter [19] data structure to compactly store such provenance and provenance witness information. The provenance information is stored at the nodes in the packet path; the provenance witness information is recorded by the nearby nodes who have witnessed the packet transmission at some nodes in the packet path.

When the BS needs to retrieve the provenance of a received packet, it requires the WSN to determine the nodes that were in the packet path. The BS trusts self-claimed information from a node that it was in the packet path if and only if the number of the witness nodes which supporting such self-claimed information is greater than a value K, where $K \in \mathbf{N}^+$ is a preset empirical value.

3.2.2 CTrace

CTrace (Contact-based traceback) [30] is another distributed provenance scheme in which each node uses its contact nodes within R hops and the PPM (Probabilistic Packet Marking) [9] approach to encode provenance. At a node n_i, for each arrived packet p, if p has been marked by the contacts of n_i, n_i will generate a digest for p using a hash function and then store the digest and the ID of p together. Furthermore, if the PPM condition holds (viz., upon reception of p, n_i generates a random number r, and r is less than a preset empirical value v), n_i will add its ID to the provenance of p. If p has not been marked by the contacts of n_i, n_i will add its ID to the provenance of p and then send it to the next node.

When the BS wants to retrieve the provenance of a received packet p, the BS will send a query to one of its contact nodes n_x, and then the packet path of p from n_x to the BS is reconstructed by combining the provenance information from the contact nodes of n_x. Subsequently, n_x will send the same query to one of its contact nodes n_y, and then the packet path of p from n_y to n_x is reconstructed similarly. Such a query stops when the data source node of the packet path of p is found.

3.2.3 Pros and Cons

The distributed provenance schemes spread the provenance information on the nodes along the packet path from the data source node to the BS. As a result, the BS does not receive the entire provenance information together with the corresponding received packet. The advantages of the distributed provenance schemes include both energy and wireless bandwidth savings because of the limited provenance information attached to every packet. The disadvantage of the distributed provenance schemes is that provenance decoding is not robust. Compromised nodes and link degradations, which are normal events in WSNs, may cause provenance decoding failures.

3.3 Block Provenance Schemes

Block provenance schemes partition the provenance into a series of blocks, and then each packet only carries with it one of the blocks. At the BS, all the provenance blocks from the same provenance are aggregated to reconstruct the entire provenance.

3.3.1 PN

Sultana et al. proposed the PN (Pseudo Noise Code) provenance scheme [24] which encodes a large provenance into a series of smaller binary blocks through pseudo noise code and then transmits these binary blocks via inter-packet delay channels. Moreover, the direct sequence spread spectrum (DSSS) technique, an approach used for enabling multiple users to transmit simultaneously on the same frequency range by utilizing distinct PN [18], is used to encode the node IDs along the packet path into the provenance.

Under the PN scheme, each node uses a PN sequence of L_P bits to uniquely represent its ID. No matter how many nodes in the provenance, the medium for provenance transmission is a set of L_P inter-packet delays (IPDs) formed by $L_P + 1$ consecutive packets that transmit on the same packet path. For an inter-packet delay transmitted from the data source to the BS, every node in the packet path only adds one bit from its PN sequence into the delay sequentially. Therefore, assume that there are N nodes in a packet path, when an inter-packet delay arrives at the BS, it contains N bits. Thus, $L_P + 1$ packets, which form L_P inter-packet delays, will exactly transmit $N \cdot L_P$ bits to the BS. Because the PN sequences are orthogonal, even if each delay contains N bits from N different nodes, by multiplying every node ID in the WSN and the received delays together the BS can retrieve the node IDs in the packet path.

It is worth noting that the PN scheme not only uses the inter-packet delays as the media for the provenance, but also uses them to protect both the security and the secrecy of the provenance. The inter-packet delay channel is a side channel in packet-switched networks, which uses different delays between packets as the medium to carry messages [6], e.g., 5, 10, 15 and 20 ms represent binary blocks 00, 01, 10 and 11, respectively (see Fig. 2). At the other side of the channel, the receiver filters the inter-packet delays through the arrived packets and then retrieves the encoded binary blocks. Because the inter-packet delays are not normal media used to encode messages, such channels can penetrate most network firewalls without being noticed. Therefore, using inter-packet delays as the media for the provenance allows one to hide the provenance information. Furthermore, under the PN scheme, each node in the packet path does not just encode one bit of its PN sequence in the form of plain text because the corresponding delay is generated by using the encryption key of the node.

3.3.2 PPF

Fahmy et al. [2] proposed the PPF (Probabilistic Provenance Flow) scheme which probabilistically incorporates the node IDs in the packet path into the provenance, and therefore each packet only carries a block of the provenance to the BS. Consequently, the BS has to collect all the blocks of a provenance for decoding, which makes such a technique reminiscent of the PPM approach [9].

Under the PPM approach, each node along the packet path makes an independent decision about whether to append its ID to a passing by packet. The PPM approach assumes that packet paths are static and that each packet only contains one node ID. If there are N nodes from the data source to the BS, at least N packets shall be involved in provenance decoding. To save energy in WSNs, the PPF

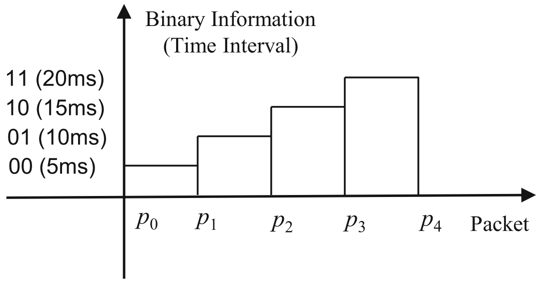

Fig. 2 Different inter-packet delays represent different binary information

scheme probabilistically encodes a connected subgraph of a packet transmission path into the packet instead of one node ID only. Therefore, to decode such a provenance the PPF scheme uses less than N packets when compared to the PPM approach.

To probabilistically encode the subgraph of a packet transmission path into the packet, the PPM scheme uses three different techniques: *rank method*, *prime method* and *Rabin fingerprints method*.

(1) *Rank method* The rank method first juxtaposes all node IDs in a line and then assigns each node a rank(ID) according to the node's position in the line. Then, instead of embedding a node ID directly into a packet, the rank(ID) of the node is embedded. In WSNs, hexadecimal numbers with the same length are typically used to represent node IDs, e.g., 0×2501, 0×2502, etc. Furthermore, if a WSN contains different types of nodes, they may not always use the same approach to represent their node IDs. As a result, by using the rank method the BS can use natural numbers $0, 1, 2, \ldots$ to represent these nodes, and therefore the provenance size is compressed. As the BS knows the bijection between a node ID and its rank(ID), the subgraphs carried by each packet can be decoded.

(2) *Prime method* according to number theory, an integer number greater than 1 can be uniquely presented as the product of a series of prime numbers. The prime method is motivated by the idea of using prime numbers as node IDs and then encoding a set of IDs through their multiplication which can be uniquely factorized. However, prime numbers multiplication incurs computational and spatial overhead when the participating prime numbers become larger [2]. It is worth noting that prime numbers are sparsely distributed with the natural number increase on the number line, which shows the infeasibility of directly using prime numbers as node IDs in WSNs. As a trade off in the prime numbers to represent node IDs, the prime method encodes a sequence of node IDs, which compose the subgraph of a packet path, by multiplying a series of nearby prime numbers for their IDs and summing up the corresponding offset values. The prime method's decoding requires knowing node orders, which can be obtained by applying the rank method first, i.e., after a configurable period of time during which the provenance is constructed using the rank method, the prime method is then applied. Compared to the rank method, the prime method achieves a higher provenance compression ratio. Upon receiving a packet, the BS has to first factor the product to retrieve all the prime numbers and then has to traverse the different decompositions of the offset values in their sum to retrieve the node IDs in the subgraph being encoded in the packet. When all the subgraphs are retrieved, the entire provenance is reconstructed by integrating all the subgraphs together.

(3) *Rabin fingerprints method* When using the prime method, although only the nearby prime numbers for node IDs in a packet path are multiplied, such a product increases rapidly with the increase of the number of nodes in a WSN as well as the number of nodes in the subgraph being encoded into a packet. As an alternative for the prime method, the Rabin fingerprints method [16] uses a polynomial to represent a sequence of bits, noting that all node IDs of an encoded subgraph also consist a sequence of bits. For a sequence of bits $n_1, n_2, \ldots, n_m$, of length m, the Rabin fingerprint is given by the following expression, where α and M are constant integers:

$$\mathrm{RF}(n_1, \ldots, n_m) = (n_1\alpha^{m-1} + n_2\alpha^{m-2} + \ldots + n_m) \bmod M. \tag{1}$$

When the sequence of bits $n_1, n_2, \ldots, n_m$, which represents the subgraph being encoded into a packet, is replaced by its Rabin fingerprints $\mathrm{RF}(n_1, n_2, \ldots, n_m)$, the provenance is compressed.

By sharing α and M between the nodes and the BS, $RF(n_1, n_2, \ldots, n_m)$ can be decoded as $n_1, n_2, \ldots, n_m$ at the BS, and therefore the subgraph carried by every packet is retrieved. The entire provenance is then obtained by integrating all the subgraphs together.

3.3.3 Pros and Cons

The block provenance scheme partitions a longer provenance into a series of smaller blocks and then appends only one block to a packet or an inter-packet delay. Because it can effectively mitigate the provenance size explosion, the block provenance scheme is probably the only one that can be applied in extremely large-scale WSNs. However, if a provenance is divided into N blocks, at least $N + 1$ packets in the PN scheme and N packets in the PPM scheme are required to be transmitted in the same packet path, respectively.

The PN scheme has some additional advantages: (1) The provenance is transmitted through a side channel of the data packet transmission channel, and therefore no extra data is appended to the packet, which saves energy during transmission; (2) the provenance is protected with respect to both security and secrecy. The disadvantage of the PN scheme is its weak robustness. Because the network transmission protocols are not designed to transmit the inter-packet delays, some normal packet transmission events, e.g., packet loss, packet aggregation, etc., can disable the inter-packet delay channels [3].

The Rabin fingerprints method has the best provenance compression ratio, followed by the prime method and the rank method. However, it is a computational intensive method for the nodes in a packet path.

3.4 Lossy Provenance Compression Schemes

In order to further reduce the provenance size, lossy provenance compression approaches have been proposed. Although the node IDs on a packet path are all embedded, the topology graph for those nodes is discarded or only partially kept.

3.4.1 BFP

Shebaro et al. [21] proposed a lightweight secure provenance scheme which we refer to as the BFP (Bloom Filter Based Provenance) scheme.

A Bloom filter (BF, for short) is a simple space-efficient randomized data structure for representing a set in order to support membership queries [5]. The BF uses an array of m bits and k independent hash functions for the probabilistic representation of a set of items $S = \{s_1, s_2, \ldots, s_n\}$. Initially all m bits in the array are set to 0.

To insert an element $s_i \in S$ into a BF, s_i is hashed with all the k hash functions. Each hash function h_i maps s_i uniformly to a position within the range $[0, m-1]$ and then the corresponding bit of that position in the array is changed to 1. To query the membership of an item s_i within S, s_i is hashed by the k hash functions to yield k positions; if any of the corresponding bit in the array is 0, $s_i \notin S$. Otherwise, either $s_i \in S$ or it is a false positive. Figure 3 shows an example of a BF's encoding and decoding.

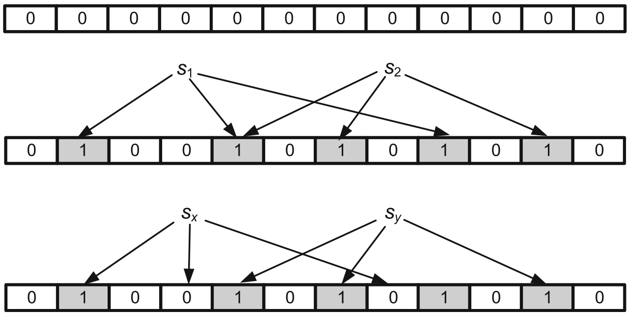

Fig. 3 An example of a Bloom Filter. The filter begins as an array of all 0s. Each item in the set s_i is hashed k times, with each hash yielding a bit location; these bits are set to 1. To check if an element s_x is in the set, hash it k times and check the corresponding bits. The element s_x cannot be in the set, since a 0 is found at one of the bits. The element s_y is either in the set or the filter has yielded a false positive

Let m be the BF size, k be the number of hash functions and D be the maximum number of the elements in S. The false positive probability is equal to that of getting 1 in all the k array positions computed by the hash functions while querying the membership of an element that was not inserted in the BF, i.e., the probability is [17]:

$$F_{FP} = \left(1 - \left(1 - \frac{1}{m}\right)^{kD}\right)^{k} \approx (1 - e^{-\frac{kD}{m}})^{k}. \quad (2)$$

Under the BFP scheme, each node in the packet path encodes its ID into an array through the BF and then appends the array to the passing by packet. Before the data source node ID is encoded, all elements in the array are set to 0. Upon reception of a packet, the BS tests every node in the WSN to get the nodes in the packet path.

We refer to the BFP scheme as a lossy approach because: (1) False positives may arise, i.e., some nodes not in the packet path are possibly decoded as if they were in the path; (2) just based on the recovered node IDs, the BS is unable to recover the packet path's topology, i.e., the provenance graph.

In view of this, the block provenance schemes PN and PPF are lossy schemes too. Under the PN scheme, the BS knows the node IDs but not the topology of those nodes. Under the PPF scheme, although the BS knows a series of subgraphs, from the entire provenance graph one cannot precisely determine if those subgraphs can legitimately compose two or more different provenance graphs.

To reconstruct the entire provenance graph when the BFP scheme is used, Sultana et al. [23] use a recursive backtracking algorithm with the neighboring information of nodes at the BS. Moreover, they chain the adjacent packet sequence numbers together along the packet path to detect provenance forgery and packet dropping attacks. We believe that such a method is applicable to the PPF scheme too, where the PPF needs to integrate the subgraphs instead of the node IDs to reconstruct the provenance.

3.4.2 Pros and Cons

Because the topology for the provenance graph is not included (in the PN, BFP schemes) or only partially included (in the PPF scheme), lossy schemes can achieve a higher provenance compression ratio. Furthermore, as the entire topology is not included, energy is saved at every node.

Although the topology can be reconstructed through a recursive backtracking algorithm and the neighboring information of the nodes at the BS, such an algorithm is computationally intensive and time consuming. In a real-time system, it may negatively affect the data trustworthiness evaluation.

3.5 Lossless Provenance Compression Schemes

Lossless provenance compression schemes encode the entire provenance graph into a single packet at every node in the packet path.

Because provenance size expands rapidly with the increase of the packet transmission hops, the lossless provenance compression schemes require efficient encoding approaches to mitigate the expansion.

3.5.1 ACP

Hussain et al. [11] proposed an arithmetic coding-based provenance (ACP) scheme. Unlike most of the provenance schemes, its provenance size for a packet is not directly proportional to the number of packet transmission hops, but to the occurrence probability of the packet path in the WSN. For instance, consider two packet paths pp_1 and pp_2 that include K_1 and K_2 nodes and assume that the occurrence probabilities of pp_1 and pp_2 are P_1 and P_2, respectively. If $P_1 > P_2$, even if $K_1 \gg K_2$, the provenance size of pp_1 is smaller than that of pp_2.

Arithmetic coding is a lossless data compression technique which achieves a compression ratio at most one bit longer than the compressed file's entropy [26, 28]. According to Shannon's theory, the entropy of a file is the upper bound of the file's lossless compression. Each codeword in arithmetic coding is a half-open subinterval of the half-open unit interval [0.0, 1.0), where each subinterval's length is proportional to its codeword's occurrence probability in the file to be compressed.

Figure 4 shows an example of arithmetic coding's procedure. Assume that a message only contains three symbols, *a*, *d*, *i*, and that their occurrence probabilities are 0.4, 0.4, and 0.2, respectively. Suppose that we need to encode a new message *aid* composed by those three symbols. The encoding procedure starts by dividing the half-open unit interval [0, 1) into three half-open subintervals: [0.0, 0.4) for *a*, [0.4, 0.8) for *d*, and [0.8, 1.0) for *i*. As *a* is the first node on the path, its interval [0.0, 0.4) is further divided into three subintervals [0.0, 0.16), [0.16, 0.32), and [0.32, 0.4), where the ratios of the new subintervals are the same as the original occurrence probabilities of the symbols. Subsequently, to encode *i*, the corresponding interval [0.32, 0.4) is selected and then further divided into [0.32, 0.352), [0.352, 0.384), and [0.384, 0.4) using the same ratio mentioned above. Finally, the last symbol *d* falls into interval [0, 616, 0.648) and thereby the *aid* is represented as [0.352, 0.384) through arithmetic coding.

The arithmetic coding decoder recovers a message from an interval [*a*, *b*), where $0 \leq a, b \leq 1$, through a procedure similar to that of the encoder. The decoder begins with the unit half-open interval [0.0, 1.0) and divides it into the same subintervals as the encoder. The first symbol is recovered by locating the subinterval in which the destination interval [*a*, *b*) resides. The subinterval is further divided in the same manner to recover the subsequent symbols. The procedure terminates when the current interval is equal to [*a*, *b*). for details about arithmetic coding, we refer the readers to [20, 26].

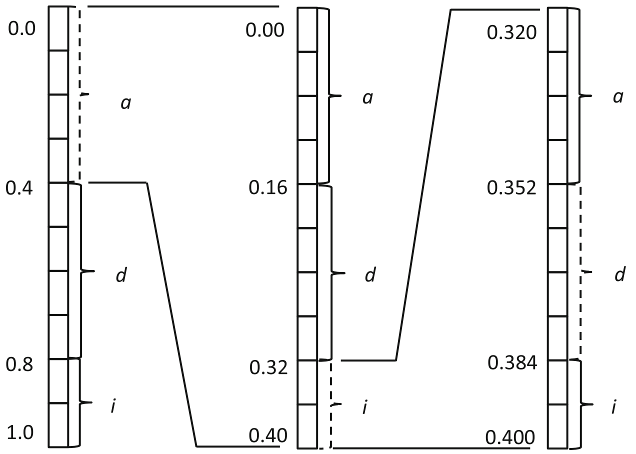

Fig. 4 An example of an arithmetic coding. The occurrence probabilities of *a*, *d*, *i* are 0.4, 0.4, 0.2 respectively; the cumulative occurrence probabilities assigned to *a*, *d*, *i* are [0.0, 0.4), [0.4, 0.8), and [0.8, 1.0). The arithmetic coding interval for *aid* is then [0.352, 0.384)

In a WSN, according to a node's occurrence probability among all the used packet paths the ACP scheme assigns each node a global cumulative probability. Furthermore, the conditional probability is computed for each pair of connected nodes. Such conditional probabilities are used to generate the cumulative probabilities for the directed edges in the provenance graph. Given a packet path in a WSN, the ACP scheme uses the global cumulative probability of the data source as the first coding interval, and then uses the cumulative probabilities derived from the conditional probabilities for all connected node pairs to generate the provenance through arithmetic coding's encoding algorithm. Under the ACP scheme, the provenance is represented by a subinterval of [0.0, 1.0). With the same global cumulative probabilities for nodes and the same conditional probabilities for node pairs, the provenance can be decoded at the BS through arithmetic coding's decoding algorithm.

3.5.2 DP

In the past for a long time, people thought that a file's entropy is the upper bound for the file's lossless compression until the dictionary-based approach [33] was proposed.

The dictionary-based approach scans a file, in the form of a symbol string, for sequences of symbols occurring multiple times, and then these sequences of symbols are indexed and stored in a dictionary. Subsequently, the compressed file is generated by replacing the repetitive sequences of symbols with their indices. When a file contains long repetitive sequences of symbols, the length of the compressed file can be even smaller than the file's entropy.

Wang et al. [27] proposed a dictionary-based provenance encoding (DP) scheme which treats each node ID as a symbol and a packet path is then a symbol string. As a result, the dictionary-based compression approach can be applied to encode the provenances in WSNs.

Because no loop on a packet path is allowed, there is no repetitive sequence of symbol on a packet path. As a result, the DP scheme uses the past packet paths to generate the dictionary.

It is worth noting that each node has a dictionary of itself and the BS keeps the dictionary of every node in the WSN. Thus, the provenance can be encoded distributively at each node and centrally decoded at the BS by looking up those dictionaries. Table 1 shows the dictionary generated at each node using the two packet paths $\langle n_{10}n_8n_7n_4n_2\mathrm{BS}\rangle$ and $\langle n_9n_5n_3n_1\mathrm{BS}\rangle$ in Fig. 5. After the dictionary in Table 1 has been built at each node and shared with the BS, the new packet path $\langle n_{10}n_8n_7n_6n_5n_3n_1\mathrm{BS}\rangle$ can be compressed as $\langle(n_{10},n_7)n_6(n_5,n_1)\mathrm{BS}\rangle$. After such a packet path has been stored in the dictionaries, it can be further compressed as $\langle(n_{10},n_1)\mathrm{BS}\rangle$.

3.5.3 Pros and Cons

Because the entire provenance graph is encoded into a single packet, the lossless provenance compression schemes are more robust compared to the block provenance schemes and the distributed provenance schemes. Moreover, the lossless provenance compression schemes generate a moderate average provenance size.

Table 1 Dictionaries generation at each node for the two packet paths $\langle n_{10}n_8n_7n_4n_2\mathrm{BS}\rangle$ and $\langle n_9n_5n_3n_1\mathrm{BS}\rangle$ in Fig. 5, where the index for a packet path snippet is composed of the first and the last node IDs of the packet path snippet

Node ID	Packet path	Path index
n_{10}	$\langle n_{10}\rangle$	$(n_{10},\emptyset)$
n_8	$\langle n_{10}n_8\rangle$	(n_{10},n_8)
n_7	$\langle n_{10}n_8n_7\rangle$	(n_{10},n_7)
n_4	$\langle n_{10}n_8n_7n_4\rangle$	(n_{10},n_4)
n_2	$\langle n_{10}n_8n_7n_4n_2\rangle$	(n_{10},n_2)
n_6	$\langle n_6\rangle$	$\emptyset$
n_9	$\langle n_9\rangle$	$(n_9,\emptyset)$
n_5	$\langle n_9n_5\rangle$	(n_9,n_5)
n_3	$\langle n_9n_5n_3\rangle$	(n_9,n_3)
n_1	$\langle n_9n_5n_2n_1\rangle$	(n_9,n_1)

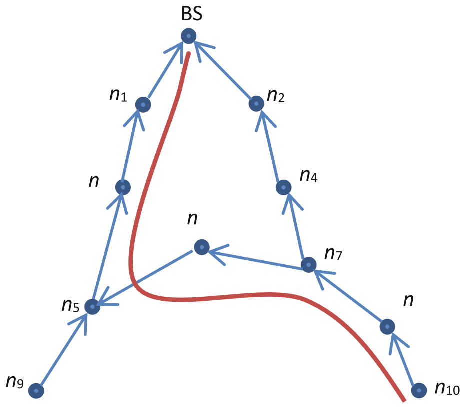

Fig. 5 Packet path dictionaries generation through the past packet paths $\langle n_{10}n_8n_7n_4n_2\mathrm{BS}\rangle$ and $\langle n_9n_5n_3n_1\mathrm{BS}\rangle$

The ACP scheme needs a training phase in order to assign the occurrence probability to each node. If the occurrence probabilities are not accurate, the compressed provenance size will deviate from the optimum size that can be achieved by arithmetic coding. Furthermore, the ACP scheme requires transmitting two real numbers to define the coding interval, which expands the provenance size.

Although the DP scheme can compress a provenance to a size which is even smaller than the provenance's entropy, such a property only holds when the topology of the WSN is relatively stable, because in a WSN with unstable topology the packet path dictionaries are difficult to build. In the worst case, if no packet can reuse a past packet path, the provenance encoded by the DP scheme is not compressed at all.

4 Provenance Security

As provenance is a key factor for data trustworthiness evaluation in WSNs, it needs to be protected.

4.1 Security Requirements

Key security requirements for provenance are: *confidentiality*, *integrity* and *availability*.

Provenance confidentiality requires that from observing data packets and their associated provenance, it is computationally infeasible for attackers to gain information about nodes and their topology in the provenance.

Provenance integrity can be further categorized as *origin integrity* and *data integrity*. Origin integrity requires that a data packet cannot reuse a provenance from one of other packets as its own provenance without being detected by the BS. Data integrity requires that an attacker or a set of colluding attackers cannot selectively add or remove nodes from the provenance generated by benign nodes without being detected by the BS. It is worth noting that some approaches [23, 24] use the notion of provenance freshness instead of provenance origin integrity. In fact, these two notions have the same connotation.

Provenance availability requires that the BS can use the provenance information of interest with reasonable computational costs. Provenance availability is important for data trustworthiness evaluation, because the evaluation only can be performed if the BS is able to gather the provenance information with affordable computational costs. In view of this, as Sultana et al. [23] use a computational intensive backtracking algorithm to reconstruct the provenance, its availability is weaker compared to the provenance schemes that can reconstruct the provenance with lower computational costs.

4.2 Provenance Binding

To prevent unauthorized modifications, except for the elementary provenance schemes using MAC (message authentication code) to encode provenance, the other provenance schemes have to bind the data and the provenance through additional MACs. Consequently, if either the provenance or the data are tampered, the BS is able to detect such an unauthorized modifications.

The most common MAC approaches for assuring the integrity of data are based on cryptographic hash functions, such as MD5 and SHA-1. Assuming that we apply these MAC approaches, the binding of data generated by MD5 or SHA-1 will contribute 128 bits or 160 bits to the provenance size at each node respectively, which is very expansive for resource-tightened WSNs.

To address such issues, a distributed message digest scheme, the AM-FM sketch scheme [7] with adjustable output length relating to the false positive rate has been adopted in recent provenance schemes [11, 22, 23, 27]. The AM-FM sketch scheme prevents the binding data's size from growing beyond the range $[(1-\epsilon)k, (1+\epsilon)2]$ with probability $1-\delta$, where k is the sample size of the provenance; $0<\delta<1$ and $\epsilon<1$ are the false positive and false negative rates related to k assuming that $O(k \geq \frac{\log(2/\delta)}{\epsilon^2})$. Furthermore, when distributively computing the digital digest, the AM-FM scheme also uses a symmetric encryption based digital signature approach at each node to protect the provenance.

As most compressed MAC schemes have false positives [4], only with a certain statistical confidence we can assume that unauthorized provenance modification can be detected by using the AM-FM sketch.

5 Future Work

Although the provenance for a packet only records the packet's forwarding and aggregation information, the average provenance size expands with the increases of the packet transmission hops and the amount of nodes in a WSN. Even if several different encoding techniques have been proposed, when dealing with extra large-scale WSNs, these schemes suffer from the following shortcomings: (1) Querying each node to retrieve a packet's provenance in an extra large-scale WSN is not only time consuming, but also the broadcast flooding can deplete the battery on every node; the distributed provenance schemes suffer from this shortcoming; (2) using the neighboring information of each node and a recursive backtracking algorithm at the BS to recover the provenance graph is an NP-complete problem when the WSN has a large amount of nodes; the lossy provenance schemes suffer from this shortcoming; (3) integrating all provenance blocks of the same packet to recover the provenance requires that all the provenance blocks are transmitted on the same packet path; the block provenance schemes are not robust enough to deal with a large number of packet transmission hops; (4) even if the compression methods can mitigate the provenance size's expansion, in an extra large-scale WSN the compressed provenance's size will exceed the capacity of a packet very likely and then the lossless provenance compression schemes do not work properly.

The shortcomings of the elementary provenance schemes are deliberately not discussed because even for a WSN of moderate size, such schemes are not suitable because their provenance size expands too fast to be transmitted through wireless channels.

To address such shortcomings, a promising approach is based on an *incremental resolution provenance scheme*, which reconstructs the provenance from a coarse-grained provenance graph to the fine-grained ones. Such a provenance scheme combines the block provenance methods and the lossless provenance compression methods. When the provenance of a smaller size can be appended to a single packet, the number of the provenance block is equal to one, i.e., the incremental resolution provenance scheme becomes a lossless provenance compression scheme. When the provenance is large and cannot be appended to a single packet, the provenance will be transmitted as a series of blocks, where the first block contains the coarse-grained provenance graph and the following sequences of blocks contain the incremental information for retrieving the fine-grained information about the provenance graph.

Under such a scheme, the BS does not need to wait for all the provenance blocks to be received properly in order to start decoding. The BS can incrementally reconstruct the provenance graph from the coarse-grained one to the fine-grained ones until the precise provenance graph is reconstructed. Even if some provenance blocks are lost during transmission, the BS can decode the provenance at a certain granularity, and therefore assess the data trustworthiness at such a granularity.

6 Conclusions

In this paper we have surveyed the main approaches to encode provenance in WSNs. Special attention has been devoted to a systematic and comprehensive classification of the solutions proposed in the literature. As to the five different kinds of provenance schemes identified in the paper, it is difficult to determine which one is always better than the others. Each kind of scheme has its own application scenarios. The elementary provenance schemes are the simplest to implement in small-scale WSNs. The distributed provenance schemes store provenance information at both the nodes in the packet path and the nearby nodes of the packet path, and therefore do not transmit the provenance with the corresponding packet. If the WSN is located in a protected environment and the BS rarely needs to verify the received data, such schemes are good choices. The block provenance schemes are able to transmit provenance of large size through a series of provenance blocks. Such schemes are the only ones that do not suffer from the packet capacity overload problem. The lossy provenance compression schemes can achieve very high compression ratio at the cost of discarding the topology of the provenance graph. Moreover, in a sparse WSN (the matrix for the WSN's topology graph is a sparse one) or a WSN with a small number of nodes, the topology can be retrieved through a recursive backtracking algorithm based on information on the node neighbors. The lossless provenance compression schemes append the entire provenance to each single packet; these schemes are thus the most robust when compared with the other provenance schemes.

We also discussed novel approaches based on the incremental resolution provenance schemes. As a combination of the block provenance schemes and the lossless provenance compression schemes, such provenance schemes could outperform the other provenance schemes with respects to both provenance compression ratio and provenance decoding efficiency.

Acknowledgments This work is supported by the National Science Foundation under grants CNS-1111512 and CICI-1547358, by the National Science Foundation of China under grant 61672269, by Jiangsu Provincial Science and Technology Project under grant BA2015161, and by the Bajian Project for Selected Researchers in Jiangsu University under grant 1213000013.

References

1. Akyildiz IF, Su WL, Sankarasubramaniam Y, Cayirci E (2002) A survey on sensor networks. IEEE Commun Mag 40(8):102–114. doi:10.1109/mcom.2002.1024422
2. Alam SI, Fahmy S (2014) A practical approach for provenance transmission in wireless sensor networks. Ad Hoc Netw 16:28–45. doi:10.1016/j.adhoc.2013.12.001
3. Archibald R, Ghosal D (2012) A covert timing channel based on fountain codes. In: 2012 IEEE 11th international conference on trust, security and privacy in computing and communications, IEEE. pp. 970–977. doi:10.1109/TrustCom.2012.21
4. Bishop M (2002) Computer security: art and science. Addison Wesley, Englewood Cliffs
5. Broder A, Mitzenmacher M (2004) Network applications of bloom filters: a survey. Internet Math 1(4):485–509. doi:10.1080/15427951.2004.10129096
6. Cabuk S, Brodley CE, Shields C (2009) IP covert channel detection. ACM Trans Inf Syst Secur 12(4):1–29. doi:10.1145/1513601.1513604.1513604
7. Garofalakis M, Hellerstein JM, Maniatis P (2007) IEEE: proof sketches: verifiable in-network aggregation. In: IEEE 23rd international conference on data engineering, pp 971–980. doi:10.1109/ICDE.2007.368958
8. Gnawali O, Fonseca R, Jamieson K, Kazandjieva M, Moss D, Levis P (2013) Ctp: an efficient, robust, and reliable collection tree protocol for wireless sensor networks. ACM Trans Sens Netw 10(1):16:1–16:49. doi:10.1145/2529988
9. Goodrich MT (2008) Probabilistic packet marking for large-scale ip traceback. IEEE/ACM Trans Netw 16(1):15–24. doi:10.1109/TNET.2007.910594
10. Hasan R, Sion R, Winslett M (2009) The case of the fake picasso: preventing history forgery with secure provenance. In: Proceedings of the 7th conference on File and storage technologies, FAST '09, USENIX Association, Berkeley, CA, USA, pp 1–14
11. Hussain S, Wang C, Sultana S, Bertino E (2014) Secure data provenance compression using arithmetic coding in wireless sensor networks. In: Performance computing and communications conference (IPCCC), 2014 IEEE international, pp 1–10. doi:10.1109/PCCC.2014.7017068
12. Jayapandian M, Chapman A, Tarcea VG, Yu C, Elkiss A, Ianni A, Liu B, Nandi A, Santos C, Andrews P, Athey B, States D, Jagadish HV (2007) Michigan molecular interactions (mimi): putting the jigsaw puzzle together. Nucleic Acids Res 35(suppl 1):D566–D571. doi:10.1093/nar/gkl859
13. Karlof C, Sastry N, Wagner D (2004) Tinysec: a link layer security architecture for wireless sensor networks. In: Proceedings of the 2nd international conference on Embedded networked sensor systems—SenSys '04, vol 3. ACM Press, New York, New York, USA, pp 162–175. doi:10.1145/1031495.1031515
14. Lim HS, Moon YS, Bertino E (2010) Provenance-based trustworthiness assessment in sensor networks. In: Proceedings of the seventh international workshop on data management for sensor networks, DMSN '10. ACM, New York, NY, USA, pp 2–7. doi:10.1145/1858158.1858162

15. Liu A, Ning P (2008) Tinyecc: a configurable library for elliptic curve cryptography in wireless sensor networks. In: 2008 international conference on information processing in sensor networks (ipsn 2008), IEEE. pp 245–256. doi:10.1109/IPSN.2008.47
16. Rabin MO (1981) Fingerprinting by random polynomials. Report, tr-cse-03-01, Center for Research in Computing Technology, Harvard University
17. Mitzenmacher M (2002) Compressed bloom filters. IEEE/ACM Trans Netw 10(5):604–612. doi:10.1109/TNET.2002.803864
18. Robert CD (1994) Spread spectrum systems with commercial applications, 3rd edn. Wiley, Hoboken
19. Rothenberg C, Macapuna C (2011) In-packet bloom filters: Design and networking applications. Comput Netw 55(6):1364–1378. doi:10.1016/j.comnet.2010.12.005
20. Rubin F (1979) Arithmetic stream coding using fixed precision registers. IEEE Trans Inf Theory 25(6):672–675. doi:10.1109/TIT.1979.1056107
21. Shebaro B, Sultana S, Reddy Gopavaram S, Bertino E (2012) Demonstrating a lightweight data provenance for sensor networks. In: Proceedings of the 2012 ACM conference on computer and communications security, CCS '12. ACM, New York, NY, USA, pp 1022–1024. doi:10.1145/2382196.2382312
22. Sultana S, Ghinita G, Bertino E, Shehab M (2012) A lightweight secure provenance scheme for wireless sensor networks. In: 2012 IEEE 18th international conference on parallel and distributed systems (ICPADS), pp 101–108. doi:10.1109/ICPADS.2012.24
23. Sultana S, Ghinita G, Bertino E, Shehab M (2015) A lightweight secure scheme for detecting provenance forgery and packet dropattacks in wireless sensor networks. IEEE Trans Dependable Secure Comput 12(3):256–269. doi:10.1109/TDSC.2013.44
24. Sultana S, Shehab M, Bertino E (2013) Secure provenance transmission for streaming data. IEEE Trans Knowl Data Eng 25(8):1890–1903. doi:10.1109/TKDE.2012.31
25. Sy D, Bao L (2006) Captra: coordinated packet traceback. In: Proceedings of the 5th international conference on information processing in sensor networks, IPSN '06, ACM, New York, NY, USA, pp 152–159. doi:10.1145/1127777.1127803
26. Vitter JS, Howard PG, Howard PG, Vitter JS (1994) Arithmetic coding for data compression. In: Information processing and management, pp 749–763
27. Wang C, Hussain SR, Bertino E (2016) Dictionary-based secure provenance compression for wireless sensor networks. IEEE Trans Parallel Distrib Syst 27(2):405–418. doi:10.1109/TPDS.2015.2402156
28. Witten IH, Neal RM, Cleary JG (1987) Arithmetic coding for data compression. ACM Commun 30(6):520–540. doi:10.1145/214762.214771
29. Yunhao L (2013) Introduction to internet of things, 2nd edn. Science, Beijing
30. Zhang Q, Zhou X, Yang F, Li X (2007) Contact-based traceback in wireless sensor networks. In: 2007 international conference on wireless communications, networking and mobile computing, pp 2487–2490. doi:10.1109/WICOM.2007.619
31. Zhou W, Fei Q, Narayan A, Haeberlen A, Loo BT, Sherr M (2011) Secure network provenance. In: Proceedings of the twenty-third ACM symposium on operating systems principles, SOSP '11. ACM, New York, NY, USA, pp 295–310. doi:10.1145/2043556.2043584
32. Zhou W, Sherr M, Tao T, Li X, Loo BT, Mao Y (2010) Efficient querying and maintenance of network provenance at internet-scale. In: Proceedings of the 2010 ACM SIGMOD international conference on management of data, SIGMOD '10. ACM, New York, NY, USA, pp 615–626. doi:10.1145/1807167.1807234
33. Ziv J, Lempel A (1978) Compression of individual sequences via variable-rate coding. IEEE Trans Inf Theory 24(5):530–536. doi:10.1109/TIT.1978.1055934