Rabin Fingerprint-Based Provenance Compression Scheme for Wireless Sensor Network

Yu Yang, Zhiming Zhang(✉), and Wei Yang

Jiangxi Normal University, Nanchang 330022, China
zzm_9650@163.com

Abstract. Provenance is considered an effective mechanism to evaluate the reliability of data. To avoid the linear growth of provenances with the growth of the packet transmission path, this paper proposes a provenance compression scheme based on the Rabin fingerprint (RFP). In the RFP scheme, each node uses its identity ID as a seed to generate a fingerprint, the fingerprint is its provenance. When a node on the transmission path receives a packet, it performs a fingerprint connection operation between its provenance and the provenance stored in the packet to generate a new fixed-length fingerprint provenance, and the new provenance is updated to the package. When the base station receives the packet, it will recover the complete transmission path of the packet based on the provenance. Performance analysis and simulation results show that compared with existing provenance compression schemes, the provenance size of the RFP scheme not only does not increase as the path length becomes longer but also has great advantages in storage overhead and energy consumption.

Keywords: Wireless Sensor Network · Provenance · Rabin Fingerprint · Path Fingerprint

1 Introduction

The Wireless sensor network (WSN) is a distributed and self-organizing network [1]. According to the specific needs of users, many sensor nodes work together to monitor, collect and process environmental information, then transmit various environmental data to end-users through wireless channel transmission [2]. However, in sensitive fields such as health monitoring and military affairs, data is collected by sensor nodes and sent to servers [3]. Therefore, it is necessary to ensure the credibility of the sensor data collected by servers to provide reliable information to applications. Since provenance [4, 5] records the history of the packet transmission and the relevant operations on the packet [6, 7], it is possible to recover the complete path of packet transmission based on the provenance. Therefore, provenance is an effective mechanism for evaluating data credibility [8]. The simplest provenance scheme is to directly record the node ID [9] of all transmission nodes as provenance. Therefore, the size of provenance will increase linearly with the

Y. Cao and X. Shao (Eds.): MONAMI 2022, LNICST 474, pp. 331–344, 2023.
https://doi.org/10.1007/978-3-031-32443-7_24

growth of the packet transmission path. Still, the storage space, energy, and bandwidth of wireless sensor networks are limited, so it is necessary to compress the provenance for transmission.

In recent years, scholars have proposed effective schemes for provenance compression [10–15]. To reduce the size of provenance, the reference [10] proposed the probabilistic packet marking method, in which each node on the transmission path appends its information to the packet with a certain probability. But the shortcoming is that the base station needs to receive enough packets containing provenance information to recover the packet transmission path. To reduce the provenance size vigorously, the reference [12] proposed the method of the embedded bloom filter, which can effectively reduce the provenance size but has the problem of false positives. The reference [13] proposed an arithmetic coding-based provenance compression scheme, but as the network topology becomes complex, it increases the coding and decoding computation. The reference [14] proposed a digital dictionary-based provenance compression scheme, but if the network topology changes frequently, it will decrease the algorithm's efficiency. To improve the dictionary-based provenance method, which is sensitive to topology changes, reference [15] proposed a provenance compression scheme based on path index difference. However, when the network topology is complex, the efficiency of retrieving similar paths is not high. For most of the above schemes, the size of provenance increases with the path length.

This paper proposes a provenance compression scheme based on the Rabin fingerprint (RFP). The basic idea is that the base station first generates a tree with the base station as the root node from the entire wireless sensor network. Then calculates the path fingerprint of each path through the Rabin fingerprint algorithm and saves the path fingerprint and the corresponding path to the path fingerprint table. In the RFP scheme, each node uses its ID as the seed to calculate the Rabin fingerprint, and the fingerprint is its provenance. When a node on the transmission path receives a packet, it performs a fingerprint connection operation between its provenance and the provenance stored in the packet to generate a new fixed-length fingerprint provenance, and the new provenance is updated to the package. When the base station receives the packet sent by the node, it first extracts the provenance of the packet and then checks the path fingerprint table. If the path fingerprint is equal to the provenance, the path corresponding to the path fingerprint is the packet's complete transmission path. The performance analysis and experimental simulation show that compared with the existing schemes, the RF scheme has obvious advantages in terms of storage space and energy consumption as the path length increases.

2 Related Work

In recent years, scholars have proposed effective provenance compression schemes.

Chaudhari et al. [10] proposed the PPM (Probabilistic Packet Marking) method. When a packet passes through a node, each node will write its node ID into the packet with a certain probability. When the base station receives enough marked packets, it will obtain the ID of each node on the transmission path to recover the transmission path of the packet. Although the compression rate of this scheme is high, a large number of

packets will overload the whole network, and problems such as a significant error rate in reconstructing the path will be associated.

Alam et al. [11] proposed the PPF (Probabilistic Provenance Flow) method, which uses the IDs of nodes to construct provenance. The IDs of each node were embedded into packets following different algorithms according to a certain probability. The compression effect of this method is better than PPM. However, similarly, the base station has to receive enough packets with marked information to reconstruct the transmission path of the packets.

Sultana et al. [12] proposed the IBF (in packet bloom filter) method. The core of this method is to embed a bloom filter in each packet and write the node's ID into the bloom filter using the Hash function. After receiving the packet, the base station extracts the provenance from the bloom filter and recovers the packet transmission path. However, this scheme has a false positive problem.

Hussain et al. [13] proposed a compression method based on arithmetic coding. This method assigns shorter code words to characters with a high probability of occurrence and otherwise allocates longer code words. The size of its provenance mainly depends on the probability of the packet passing through the node. The greater the probability is, the smaller the provenance is. Although it has a high compression rate, the encoding and decoding of provenance require a lot of computation. With the expansion of wireless sensor network scale and the complexity of network topology, excessive calculation will inevitably lead to the performance decline of the algorithm.

Wang et al. [14] proposed a dictionary-based compression scheme in which each node in the network has a dictionary sequence that stores the transmission paths of the packets. The base station only needs to query the path index value to get the complete transmission path. If the transmission path of packets does not change frequently, the dictionary usage will be high, and the algorithm will be more efficient. However, if the transmission path of packets changes frequently, the dictionary sequence will also change frequently, resulting in the inefficiency of the algorithm.

Xu et al. [15] proposed a provenance scheme based on the difference in path index. The scheme first builds the backbone path along the gradient direction, then de-duplicates the backbone path based on the method of Truncation Hamming Distance, and then builds the dictionary of the backbone path after de-duplication. When a new transmission path appears, the path most similar to it in the dictionary is retrieved, representing the new path as its index difference form. Although this scheme improves the problem that the dictionary-based provenance compression method is sensitive to network topology changes, it is inefficient to retrieve similar paths when the wireless sensor network is extensive in scale.

3 The System Model

3.1 Rabin Fingerprint

The Rabin fingerprint algorithm [16, 17] was proposed by Rabin, a professor at Harvard University in the United States. The basic ideas are as follows:

Assuming that $S(a_1, a_2, \cdots, a_n)$ is a binary string containing n binary bits, given an integer t over a finite field $GF(2^n)$, the corresponding (n-1) degree polynomial can be constructed from the string S:

$$S(t) = a_1 t^{n-1} + a_2 t^{n-2} + \cdots + a_n \tag{1}$$

Given an m-degree polynomial $R(t) = b_1 t^m + b_2 t^{m-1} + \cdots + b_m$, the Rabin fingerprint of the string S is calculated as follows:

$$RF(S) = S(t) \bmod R(t) \tag{2}$$

Let $M = R(t) = b_1 t^m + b_2 t^{m-1} + \cdots + b_m$, then the Rabin fingerprint of the string $S(a_1, a_2, \cdots, a_n)$ can be expressed as:

$$RF(a_1, a_2, \cdots, a_n) = a_1 t^{n-1} + a_2 t^{n-2} + \cdots + a_n \bmod M \tag{3}$$

The Rabin fingerprint connection calculation of nodes n_i and n_j is as follows:

$$\begin{aligned} &RF(n_i n_j) = RF(n_i \| n_j) \bmod M = RF(RF(n_i) \| n_j) \bmod M \\ &= \left[RF\left(RF(n_i) \times t^{l_j} \right) + RF(n_j) \right] \bmod M \end{aligned} \tag{4}$$

where $\|$ denotes the connection operation, l_j denotes the string length of node n_j, and in this way, the Rabin fingerprint connection operation for nodes $n_1, n_2, \cdots, n_m$ is as follows:

$$\begin{aligned} &RF(n_1 n_2 n_3 \cdots n_m) = RF(RF(n_1 n_2 n_3 \cdots n_{m-1}) \| n_m) \bmod M \\ &= \left(RF\left(RF(n_1 n_2 n_3 \cdots n_{m-1}) \times t^{l_m} \right) + RF(n_m) \right) \bmod M \end{aligned} \tag{5}$$

where $\|$ denotes the connection operation, l_m denotes the string length of node n_m.

3.2 The Network Model

The whole sensor network consists of n common nodes and a base station. G(N,L) denotes the topology of the wireless sensor network model, where N denotes the set of all nodes in the network, and L denotes the set of edges for all nodes in the network.

$N = \{n_i, i = 1, 2, \ldots, n\}$, $L = \left\{ l_{ij}, i = 1, 2, \ldots, n;\ j = 1, 2, \ldots, n \right\}$.

Each node n_i is assigned a unique identification ID before deployment, and the ID is used as a seed to compute the Rabin fingerprint, which is the provenance that the node attaches to the packet. Once deployed, the nodes will no longer change their positions. All nodes are formed into a tree with the base station as the root, as shown in Fig. 1. If some nodes die due to energy exhaustion and the path of the wireless sensor network changes, the topology of the wireless sensor network will be automatically updated.

When the sensor node senses the data, it will send it to the base station. In the data transmission process, each node will attach its Rabin fingerprint as provenance to the packet and pass it to the next node. When the base station receives the packet, it will recover the complete transmission path of the data based on the provenance in the packet.

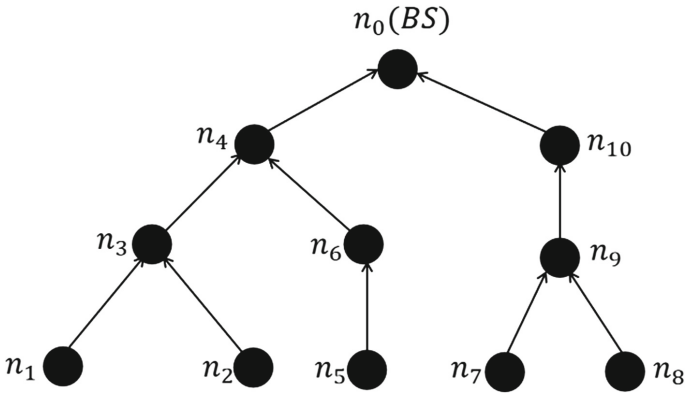

Fig. 1. Network tree topology.

4 Rabin Fingerprint-based Provenance Compression Scheme for Wireless Sensor Network

This paper proposes a Rabin fingerprint-based provenance compression scheme for wireless sensor networks (RFP). The RFP is divided into three steps: path fingerprint calculation by the base station, provenance coding, and provenance decoding.

4.1 Compute Path Fingerprint

When all nodes form a tree structure with the base station as the root node, the base station calculates the corresponding path fingerprint for each path according to algorithm 1. Assuming a path consists of nodes $n_1, n_2, \cdots, n_m$, the base station first calculates the fingerprint $RF(n_1), RF(n_2), \cdots, RF(n_m)$ of all nodes on the path according to the formula (3). The base station performs connection operation between $RF(n_1)$ and $RF(n_2)$ according to formula (4) to obtain the new fingerprint $RF(n_1n_2)$. In this way, the path fingerprint $RF(n_1n_2 \cdots n_m)$ of the path is finally calculated according to Formula (5).

In Fig. 1, assume that the ID of node n_1 is 1, the ID of node n_3 is 3 and the ID of node n_4 is 4. Set $M = 11$ and calculate the fingerprint $RF(n_1) = 1$, $RF(n_3) = 3$, $RF(n_4) = 4$ respectively according to Formula (3). Then $RF(n_1)$ and $RF(n_3)$ are connected according to Formula (4) to obtain the new fingerprint $RF(n_1n_3) = 8$. Then $RF(n_1n_3)$ and $RF(n_4)$ are connected according to Formula (5) to obtain the new fingerprint $RF(n_1n_3n_4) = 0$. Finally, the path fingerprint of the path $(n_1n_3n_4n_0)$ is calculated as $RF(n_1n_3n_4n_0) = 0$. In this way, the path fingerprints of other paths can be calculated according to algorithm 1, and finally, the path fingerprints corresponding to all paths can be stored in Table 1.

Table 1. Path fingerprint table.

Path coding	Complete path	Path fingerprint
1	(n_1, n_3, n_4, n_0)	0
2	(n_2, n_3, n_4, n_0)	3
3	(n_5, n_6, n_4, n_0)	5
4	(n_7, n_9, n_{10}, n_0)	10
5	(n_8, n_9, n_{10}, n_0)	2

Algorithm 1. Path fingerprint calculation algorithm

Input：A path with m nodes $n_1,n_2,\cdots,n_m$

Output: Path fingerprint $RF(n_1,n_2,\cdots,n_m)$

For(i=2;i≤m;i++) do

$RF(n_1n_2\cdots n_i)=RF(RF(n_1n_2\cdots n_{i-1})\|n_i)$

End for

Return $RF(n_1n_2\cdots n_m)$

4.2 Provenance Coding

(1) **Source node provenance coding.**

When a source node n_s wants to send sensor data to the base station, after generating the packet, it will create two fields on the packet to store the provenance. The format of the fields is shown in Table 2. The fingerprint field stores the path fingerprint of the current packet transmission path, and the length field denotes the path length. The source node n_s takes its ID as the seed and generates a fingerprint $RF(n_s)$ according to Formula (3). It then stores $RF(n_s)$ in the packets' fingerprint field and sets the field's value to 1. Finally, it sends the packet containing the provenance to the next node.

Table 2. Provenance of source node n_s.

fingerprint	length
$RF(n_s)$	1

(2) **Forwarding node provenance encoding.**

When a forwarding node n_i receives a packet from the previous node n_j, node n_i first extracts the original provenance fingerprint RF(Z) from the packet. It then uses its ID as the seed to calculate the fingerprint $RF(n_i)$. Then perform a fingerprint connection operation between $RF(n_i)$ and RF(Z) to obtain $RF(n_1 n_2 \cdots n_i) = RF(RF(Z)||n_i)$ mode M, take it as the new provenance, and update the fingerprint field. And then update the length field and set length = length + 1. Finally, send the updated packet to the next node. The provenance of forwarding node n_i is shown in Table 3:

Table 3. Provenance of forwarding node n_i.

Fingerprint	Length
$RF(n_1 n_2 \cdots n_i)$	length + 1

4.3 Provenance Decoding

When the base station receives the packet sent by a node, it first extracts the provenance from the packet. It then checks Table 1 to find the path fingerprint equal to the provenance, the path corresponding to the path fingerprint is the packet's complete transmission path. The specific provenance decoding algorithm is shown in Algorithm 2, where k denotes the total number of paths in the path fingerprint table.

Algorithm 2. Provenance Decoding Algorithm

```
Input：A packet P(t)
Output：The set of complete paths corresponding to the packet TP(i)
Get RF(t) from P(t)
For(i=1;i≤k;i++）do
    If (RF(t)==RF(i)) then
        Return TP(i)
End if
End for
```

4.4 Example

To understand the RFP scheme more clearly, this section gives an example to illustrate how to encode and decode provenance in RFP scheme. In Fig. 1, assume that the identity

IDs of the nodes n_1, n_3, n_4 in the path (n_1, n_3, n_4, n_0) are 1, 3, 4. Set M = 11. If the source node n_1 wants to send the packet to the base station, it first uses its ID as the seed and calculates the fingerprint RF(n_1)= 1 according to formula (3). Then store RF(n_1) in the fingerprint field of the packet, set the value of the length field is 1, and then send the packet to the next node n_3. Table 4 shows the provenance generated by source node n_1.

Table 4. Provenance of source node n_1.

Fingerprint	Length
RF(n_1)= 1	1

When node n_3 receives a packet from node n_1, it first extracts the provenance fingerprint RF(n_1) from the packet, and then uses its ID as the seed to calculate the fingerprint RF(n_3). Then perform a fingerprint connection operation between RF(n_3) and RF(n_1) to obtain RF(n_1n_3)= 8, take it as the new provenance and update the fingerprint field of the packet to 8. And then update the value of the length field to 2. Then send the updated packet to node n_4, and the provenance generated by the forwarding node n_3 is shown in Table 5. In this way, the provenance generated by the forwarding node n_4 is shown in Table 6.

Table 5. Provenance of forwarding node n_3.

Fingerprint	Length
RF(n_1n_3)= 8	2

Table 6. Provenance of forwarding node n_4.

Fingerprint	Length
RF($n_1n_3n_4$)= 0	3

After the base station receives the packet sent by node n_4, to recover the complete transmission path of the packet, it first extracts the provenance RF($n_1n_3n_4$)= 0 of the packet. It then checks Table 1 to find the path fingerprint equal to RF($n_1n_3n_4$). The path corresponding to fingerprint 0 is (n_1,n_3,n_4,n_0), so path (n_1,n_3,n_4,n_0) is the complete transmission path of this packet according to algorithm 2.

5 Performance Analysis

In this paper, we will compare RFP with BFP [12], OP [18], and MP (Message Authentication code-based provenance) from the aspects of storage overhead and energy consumption. BFP scheme is based on bloom filter, which writes provenance of nodes into

bloom filter using Hash function. In the OP scheme, node performs an orthogonal code addition operation between its provenance and the original provenance of the packet to form new provenance and a new fixed-length message identification code chain. In the MP scheme, each node n_i appends its identity tag ID_i and message authentication code directly to the packet. Since this paper assumes that the base station's computing, storage, and communication capabilities are not limited, the storage overhead and energy consumption of the base station are not discussed here.

5.1 Storage Overhead Analysis

Suppose a transmission path passes through N nodes from the source node to the base station. In the BFP scheme, the storage overhead required for the provenance of a packet is $-N \times \ln(P_{fp})/(\ln 2)^2$Bytes, where P_{fp} is the probability of false positivity of the bloom filter. In the MP scheme, the storage overhead required for the provenance of a packet is $N \times 6$Bytes. In the OP scheme, the length of a node identity tag is 4 Bytes, and the storage overhead required for the provenance of a packet is 23 Bytes.

In the RFP scheme, the provenance of a data package consists of two fields. In other words, the provenance of a data package can be represented as fingerprint, length, where fingerprint denotes the path fingerprint value. According to the fingerprint connection operation property, the result is a fixed-length fingerprint value if multiple fingerprint values are connected. If a 4 Bytes ID generates a 4 Bytes fingerprint value, the fingerprint length is still 4 Bytes. And length indicates that the path length is 1 Byte. Therefore, in the RFP scheme, the storage overhead required for the provenance of a packet is 5 Bytes.

In conclusion, both BFP and MP schemes are related to the transmission path length. As the path length increases, the storage overhead increases. Although the storage overhead of the OP scheme is a fixed value independent of the path length, it is much higher than that of the RFP scheme. The storage overheads of the RFP scheme, BFP scheme, MP scheme, and OP scheme are shown in Table 7.

Table 7. Comparison of storage overhead.

Scheme	Storage Overhead/Bytes
BFP	$-N \times \ln(P_{fp})/(\ln 2)^2$
MP	$N \times 6$
OP	23
RFP	5

5.2 Energy Consumption Analysis

In the RFP scheme, the energy consumption of each node is mainly the receiving and sending of packets. Suppose a transmission path passes through N nodes from the source

node to the base station. In the BFP and MP schemes, the storage overhead required for the provenance of a packet is $-N \times \ln(P_{fp})/(\ln 2)^2$ Bytes and $N \times 6$ Bytes, respectively. And the corresponding energy consumption increases proportionately to $-N \times \ln(P_{fp})/(\ln 2)^2$ and $N \times 6$ Bytes, respectively. In the OP scheme, the storage overhead of the provenance is a fixed value of 23 Bytes, but its energy consumption increases slowly as the transmission path increases. Compared with other schemes, the RFP scheme has a relatively small storage overhead of the provenance. Therefore, with the increase in path length, the energy consumption of the RFP scheme is much less than that of the other three schemes.

6 The Simulation Results

This paper simulates and evaluates the performance of RFP schemes in terms of average provenance size, energy consumption, and validation error rate. The simulation experiment environment is carried out on the OMNeT++ platform with 100 nodes randomly distributed at 500 m × 500 m square area. Before the node is deployed, each node is assigned a number from 0 to 99 as the unique identity ID, and the node numbered 0 is the base station. The communication range of each node is 150 m, the node will not move after deployment, and the base station is deployed in the center of the region. Randomly select some network nodes as data source nodes and others as intermediate forwarding nodes. The data source node sends a packet to the base station by multi-hop every 1 s. For each parameter, take the average of 100 simulations.

Figure 2 depicts the average provenance size of the RFP scheme, OP scheme, BFP scheme, and MP scheme under different path lengths. Assuming that the base station receives m packets sent by a source node, the average provenance size (APS) refers to the average length of the provenance for m packets. That is,

$$\mathrm{APS} = \frac{\sum_{i=1}^{m} \mathrm{PR}_i}{m} \tag{6}$$

where m denotes the number of packets received from a source node, and PR_i represents the length of the provenance for the ith packet. In the MP scheme, each node directly adds its ID to the packet as the provenance when forwarding the packet, so the average provenance size of the MP scheme increases linearly with the length of the transmission path. The average size of provenance in the BFP scheme is relatively flat compared with the MP scheme. Because in the BFP scheme, each node stores its provenance in the bloom filter of packets. Therefore, although the BFP scheme's provenance size is also related to the transmission path length, it is less obvious than in the MP scheme. In the OP scheme, when each node receives the packet, it makes an orthogonal operation between its provenance and the provenance in the packet to obtain a new provenance with a fixed-length value. Therefore, the average provenance size of the OP scheme is a constant value independent of the path length. In the RFP scheme, each node uses its ID as the seed to generate a fingerprint as provenance and performs a fingerprint connection calculation with the provenance fingerprint stored in packets to generate a new fixed-length fingerprint provenance. Therefore, the average provenance size of the RFP scheme is also a constant value. In the case of 100 nodes in the simulation experiment, no matter

how the transmission path length increases, the average provenance size remains almost unchanged, accounting for about 5Bytes.

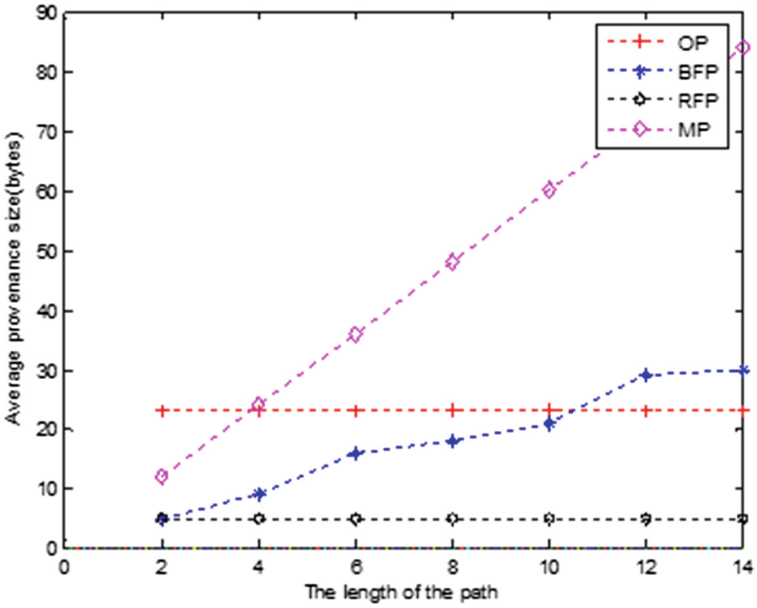

Fig. 2. Average provenance size under different path lengths.

In the RFP scheme, the energy consumption of each node is mainly on receiving and sending packets. So the total energy consumption (TEC) of all nodes in the network is,

$$TEC = \sum_{i=1}^{n} L^{*}(M_i * e_r + N_i * e_s) \tag{7}$$

where n denotes the total number of nodes in the network, L represents the packet length, and M_i denotes the number of packets forwarded by node i. N_i represents the number of packets sent by node i, e_r represents the energy consumed for receiving 1bit data, and e_s represents the energy consumed for transmitting 1bit data. Figure 3 describes the total energy consumption of the RFP scheme, OP scheme, BFP scheme, and MP scheme under different path lengths. It can be seen that when the path length exceeds five hops, the total energy consumption of the OP scheme, BFP scheme, and MP scheme is greater than that of the RFP scheme. And with the increase in the path length, the energy-saving advantage of the RFP scheme is more obvious.

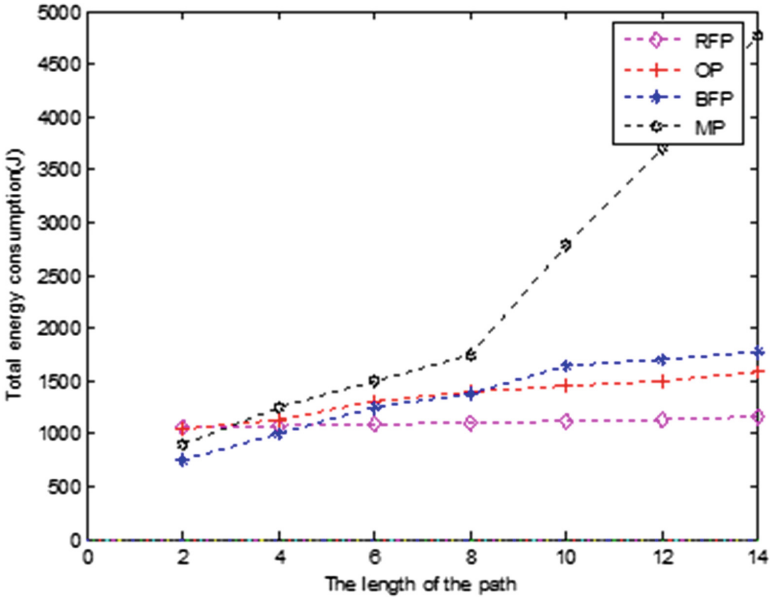

Fig. 3. Total energy consumption at different path lengths.

When the base station receives a packet sent by the source node, it will execute algorithm 2 to recover the complete transmission path of the packet. But due to changes in the network topology or packets that may be damaged in actual transmission, it will not recover the complete transmission path. Assuming that the base station receives a total of m packets, of which d have incorrect provenance, the validation error rate (VER) is,

$$\mathrm{VER} = \frac{\mathrm{d}}{\mathrm{m}} \times 100\% \tag{8}$$

Figure 4 describes the verification error rates of the RFP scheme, the OP scheme, and the BFP scheme under different path lengths. As seen in Fig. 4, when the path length is less than eight hops, there is little difference in the verification error rate of the three schemes. However, when the path length is longer than eight hops, the verification error rate of the BFP scheme is significantly higher than that of the RFP and OP scheme because of the false positive problem of the bloom filter.

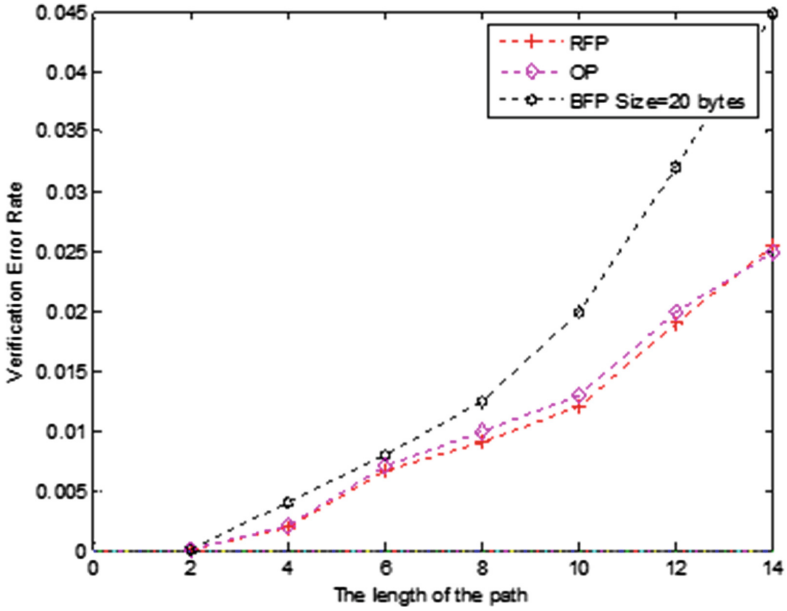

Fig. 4. Verification error rate under different path lengths.

7 Conclusion

This paper proposes an effective provenance compression scheme based on the Rabin fingerprint. In the RFP scheme, each node takes its identity ID as the seed to calculate the Rabin fingerprint as provenance. When a node n_i on the transmission path receives the packet, it performs a fingerprint connection operation with the provenance fingerprint stored in the packet to generate a new fixed-length fingerprint provenance. When the base station receives a packet sent by the node, it extracts its provenance fingerprint and then queries the path fingerprint table. If the fingerprint exists in the table, the base station can recover the complete transmission path of the packet. The RFP scheme only needs one packet to recover the transmission path, and the size of the provenance is independent of the length of the path. Performance analysis and simulation results show that compared with the existing provenance compression schemes, the RFP scheme has obvious advantages in storage overhead and energy consumption with increased path length.

References

1. Pottie, G.J., Kaiser, W.J.: Wireless Integrated Network Sensors(WINS): Principles and Practice. Commun. ACM **43**(5), 51–53 (2000)
2. Liu, V., Zhao, Y.: Wireless sensor networks for internet of things: a systematic review and classification. Info. Technol. J. **12**(16), 3581–3583 (2013)
3. Lal, S., Prathap, J.: An energy-efficient lightweight security protocol for optimal resource provenance in wireless sensor networks. Turkish J. Elec. Eng. Comp. Sci. **28**(6), 3208–3218 (2020)

4. Buneman, P., Khanna, S., Tan, W.C., et al.: Why and where: a characterization of data provenance. In: International Conference on Database Theory, pp. 316–330 (2001)
5. Ramachandran, A., Bhandankar, K., Tariq, M.B., et al.: Packets with Provenance. School of Computer Science Technical Reports, Georgia Institute of Technology (2008)
6. Dogan, G.: A survey of provenance in wireless sensor net- works. Adhoc&sensor wireless networks **30**(1/2), 21–35 (2016)
7. Xu, Q., Wang, C.: Stepwise refinement provenance scheme for wireless sensor networks. IEEE Internet Things J. **9**(13), 11126–11140 (2022)
8. Wang, C., Hussain, S.R.: Dictionary based secure provenance compression for wireless sensor networks. IEEE Trans. Parallel and Distrib. Sys. **27**(2), 405–416 (2016)
9. Hasan, R., Sion, R., Winslett, M.: The case of the fake Picasso: preventing history forgery with secure provenance. In: Proceedings of the 7th Conference on File and Storage Technologies, San Francisco, Feb 24-27, 2009, pp. 1–14. USENIX Association, Berkeley (2009)
10. Chaudhari, K.P., Turukmane, A.V.: Dynamic probabilistic packet marking. Mobile Communication and Power Engineering, 381–384 (2013)
11. Alam, S.M.I., Fahmy, S.: Energy-efficient provenance transmission in large-scale wireless sensor networks. IEEE International Symposium on a World of Wireless, 1–6 (2011)
12. Sultana, S., Ghinita, G., Bertino, E., et al.: A lightweight secure scheme for detecting provenance forgery and packet drop attacks in wireless sensor networks. IEEE Trans. Dependable Secure Comput. **12**(3), 256–269 (2015)
13. Hussain, S.R., Wang, C., Sultana, S.: Secure data provenance compression using arithmetic coding in wireless sensor networks. In: Performance Computing & Communications Conference, pp. 1–10 (2014)
14. Wang, C., Hussain, S., Bertino, E.: Dictionary based secure provenance compression for wireless sensor networks. IEEE Trans. Parallel Distrib. Syst. **27**(2), 405–418 (2015)
15. Xu, Q., Zhang, X., Wang, C.: Provenance compression using packet-path-index differences in wireless sensor networks. In: International Conference on Mobile Ad-Hoc and Sensor Networks (MSN), pp. 200–205 (2019)
16. Sun, J., Chen, H., et al.: Redundant network traffic elimination with GPU accelerated rabin fingerprinting. IEEE Trans. Parallel Distrib. Syst. **27**(7), 2130–2142 (2016)
17. Lu, P., Randall, O., McDonald, E.: An empirical study of rabin fingerprinting parameters. In: IEEE International Conference on Big Data (Big Data), pp. 3686–3691 (2019)
18. Zhang, Z., Deng, J.: A secure and effective method for provenance in Wireless Sensor Networks. Computer science and exploration **13**(4), 608–619 (2019)