information

MDPI

Article

SybilEye: Observer-Assisted Privacy-Preserving Sybil Attack Detection on Mobile Crowdsensing

Junhyeok Yun and Mihui Kim *

School of Computer Engineering & Applied Mathematics, Computer System Institute, Hankyong National University, Anseong 17579, Korea; junhyeok2723@hknu.ac.kr

* Correspondence: mhkim@hknu.ac.kr; Tel.: +82-31-670-5167

Received: 4 February 2020; Accepted: 7 April 2020; Published: 9 April 2020

Abstract: Mobile crowdsensing is a data collection system using widespread mobile devices with various sensors. The data processor cannot manage all mobile devices participating in mobile crowdsensing. A malicious user can conduct a Sybil attack (e.g., achieve a significant influence through extortion or the generation of fake IDs) to receive an incentive or destroy a system. A mobile crowdsensing system should, thus, be able to detect and block a Sybil attack. Existing Sybil attack detection mechanisms for wireless sensor networks cannot apply directly to mobile crowdsensing owing to the privacy issues of the participants and detection overhead. In this paper, we propose an effective privacy-preserving Sybil attack detection mechanism that distributes observer role to the users. To demonstrate the performance of our mechanism, we implement a Wi-Fi-connection-based Sybil attack detection model and show its feasibility by evaluating the detection performance.

Keywords: mobile crowdsensing; sybil attack; privacy preservation; observer-assisted

1. Introduction

Mobile crowdsensing is a type of sensing data collection system that uses widespread mobile devices (e.g., smartphone and smartwatches) as sensors [1]. In general, a mobile crowdsensing system consists of a data processor and users. The data processor provides a platform and generates information using the contributed sensing data. The users request the contribution of sensing data from other users or collect and provide such data to other users [2,3]. The data processor can provide an incentive for data providers to motivate them to contribute sensing data [4]. Such an incentive could be in a monetary (e.g., cash or points) or non-monetary (e.g., information or authority) form [5]. However, malicious users can provide fake sensing data to gather an incentive or destroy the system reliability. A malicious user can take other user IDs and act like a multitude of users to improve the efficiency of the attack. This type of security attack is called a 'Sybil attack' [6]. Thus, the data processor should be able to verify whether the data provider is the valid owner of the ID and drop the fake sensing data contributed to by malicious users.

When the IDs extorted by a Sybil node provide sensing data, the location of each ID appears in a single location. Based on this characteristic, the service provider can detect and block a Sybil attack by monitoring the user location. This type of Sybil node detection mechanism is called an 'Observation.' However, in mobile crowdsensing, the data processor cannot manage all user locations because the mobile crowdsensing system consists of innumerable mobile devices. Moreover, monitoring the locations of all data providers can cause location privacy problems. Thus, an existing observation mechanism cannot be directly applied to a mobile crowdsensing system.

In this paper, we propose an observer-assisted Sybil attack detection mechanism for a mobile crowdsensing system. Observers capture Wi-Fi packets using a monitor mode supporting NIC (Network Interface Controller). With the captured packets, the observer detects a malicious user

using Wi-Fi connection/disconnection and RSS (Received Signal Strength) data. If the observer detects malicious users, the observer reports the user information to the data processor. The data processor requests incentive transaction data within a specific previous period to the data provider to verify whether the reported data provider is the valid owner of the ID. By adopting observer-assisted Sybil attack detection, the data provider can preserve the location information because only the observer near the data provider can monitor the data provider's location. Moreover, the data processor can minimize the Sybil attack detection overhead because the observers reduce the number of validation candidate IDs. We validate our mechanism through an experimental test in which a Wi-Fi-connection-based Sybil attack detection model was implemented.

The rest of the paper is organized as follows. In the related studies section, we describe the mobile crowdsensing, Sybil attack and RSS-based localization. For the proposed system, we show the structure and flow of the proposed system and the Sybil attack detection flow of the proposed Sybil attack detection mechanisms. During the experiment, we implemented the Wi-Fi-connection-based Sybil attack detection model and showed its feasibility by evaluating the detection performance.

2. Related Studies

2.1. Mobile Crowdsensing

Crowdsensing is an information collection system proposed to solve the problems in WSNs (wireless sensor networks) including high sensor installation costs and poor scalability. When first proposed, it had a low feasibility because an insufficient number of mobile sensor devices were available. However, these days, mobile devices with built-in sensors are widespread. A crowdsensing system that collects data from mobile devices (e.g., smartphones, tablet PCs, smartwatches) is called mobile crowdsensing [1].

A crowdsensing system consists of a data processor and users. Users can be classified as data providers and data users (see Figure 1). The data processor generates information using the contributed sensing data and manages user the information and incentive transaction records. The data provider collects the sensing data and provides them to the data processor. As a reward for its contribution, the data provider receives an incentive from the data processor. A data user applies the generated information. In some applications, data users can be incentive providers.

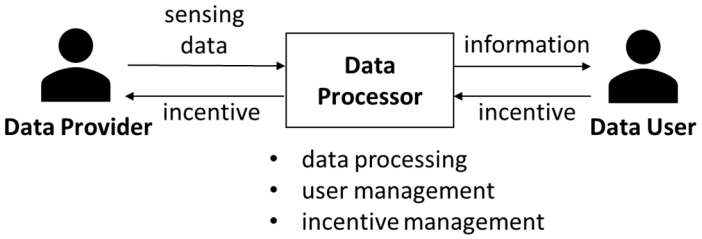

Figure 1. Crowdsensing system structure.

With a mobile crowdsensing system, we can collect the sensing data from a broader range using more sensor devices than in a WSN [7]. However, because the data processor cannot manage all participating sensor devices, the reliability of the sensing data may be compromised. Owing to the characteristics of mobile crowdsensing, malicious users can more easily provide fake sensing data and conduct a Sybil attack to improve the attack efficiency. Thus, the data processor should be able to detect and prevent a Sybil attack. However, it is practically impossible in mobile crowdsensing for the data processor to monitor all participating devices and detect the Sybil nodes. A Sybil attack detection mechanism with a high detection performance and low detection overhead is necessary for mobile crowdsensing.

In this paper, we propose a two-step Sybil attack detection mechanism to effectively detect and prevent a Sybil attack. In the first step, observers detect malicious users using physical wireless signal

data and report the detected user's information to the data processor. In the second step, the data processor verifies whether the reported users are the valid owner of the IDs by applying incentive transaction record-based validation. By separating the Sybil attack detection mechanism into two steps, the data processor can minimize the detection overhead.

2.2. Sybil Attack

A Sybil attack is a security attack in which a single user creates or extorts multiple IDs to behave as a multitude of users [6]. A service provider can detect and block a Sybil attack through an observation [8–10], certification [11], trusted device management [6], fee-charging [12] or trustiness scoring [13] (see Table 1).

Table 1. Comparison for Sybil attack detection mechanisms.

	Observation [8–10]	**Certification [11]**	**Trusted Device [6]**	**Fee-Charging [12]**	**Trustiness Scoring [13]**
Privacy Problem	O	O	X	X	X
Account Extortion Tolerant	O	X	X	O	X
Central Overhead	High	High	Low	Low	High
Additional Infrastructure Required	O	X	O	X	O
Detection Delay	Low	Low	Low	High	Low

To implement certification and trusted device management in a mobile crowdsensing system, the data processor should manage user identification information (e.g., phone number or e-mail address). In this instance, the data processor or attacker can take sensitive private information such as real-time location data by combining user-identifying information and sensing data. Moreover, certification process cannot detect the Sybil attack if the attacker extorts certification information. A fee-charge can be implemented by imposing a fee to the data provider and return it with an incentive when the sensing data validated. Fee-charging is relatively free from privacy problem and central processing overhead. However, it could demotivate sensing data contribution and it could not detect Sybil attack before the sensing data collection completes. In addition, if the Sybil attack data are more than the normal data, it is highly likely to fail detection. The trustiness scoring is a method in which the user's trustiness is scored based on historical data, similar to former sensing data provisions. However, such scoring cannot block an attacker who extorts the user with a high trustiness score. An observation is a mechanism for continuously monitoring a user's location and detecting abnormally numerous sensing data contributed from the same location. It could detect Sybil attack rapidly even if the attacker extorts certification information or trusted devices. However, in mobile crowdsensing, the system consists of massive numbers of mobile devices and the location fluidity of each device is higher than in a WSN and massive numbers of mobile devices participate in the sensing data contribution. Thus, it is practically impossible for a data processor to monitor all user locations. Even if the data processor can monitor all user locations, a centralized observation can cause a location privacy problem. Thus, a centralized observation is inadequate for mobile crowdsensing.

In this paper, we propose a privacy-preserving Sybil attack detection mechanism by which some users function as observers instead of data processor. Massive numbers of mobile devices participate in mobile crowdsensing and the number of sensor devices continues to increase with the number of participating users. The observer-assisted Sybil attack detection mechanism enables an observation during mobile crowdsensing by distributing the observer roles. By applying observer-assisted method, it is possible that the proposed mechanism overcome the problems of privacy and central overhead which are the problems of the existing observation mechanisms, while preserving the advantage of observation (i.e., account extortion tolerant or low detection delay). Instead of a central data processor, randomly selected observers among multiple observers perform location monitoring for the data providers, preventing the central data processor from tracking the location information of a specific user. Moreover, the distributed monitoring of multiple observers minimizes central processing overhead

and enables location monitoring without additional infrastructure. Thus, the observer-assisted Sybil attack detection mechanism is appropriate for use in mobile crowdsensing.

2.3. RSS-Based Localization

RSS-based localization is a mechanism that uses the physical wireless signal strength to measure the device location [14]. The RSS value represents the signal attenuation during a wireless data transmission. Widespread wireless communication standards (e.g., Wi-Fi [15] and Bluetooth [16]) include the RSS value. Thus, RSS-based localization can apply to most mobile devices without a new device installation. It requires three or more observer nodes to obtain an accurate location of the target device (see Figure 2). Each observer node communicates with the target device to obtain the RSS data. Using the RSS data, the observer calculates the distance to the target device. The observer node shares its position and distance to the target device with other observer nodes. Circles with a radius of the distance to the target device centered on the position of each observer node are drawn. The point where the three circles meet is the position of the target device.

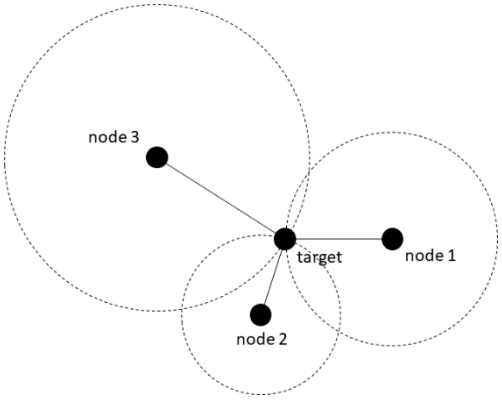

Figure 2. Received Signal Strength (RSS)-based localization with three observer nodes.

The observation could implement in the mobile crowdsensing system with RSS-based localization [8]. However, because of a high user location fluidity, two or more observer nodes are not always present in the same place. Thus, an RSS-based observation is inappropriate for application to mobile crowdsensing.

In this study, the observer scans whether another observer is present in the area. If two or more observers are in the same place, the observer detects Sybil nodes with an RSS-based observation. Otherwise, the observer detects Sybil nodes using Wi-Fi connection/disconnection data. This allows the observer to achieve the best Sybil node detection performance.

3. Proposed System

3.1. System Structure

The proposed system consists of a data processor and users (see Figure 3). Users can classify as data providers and observers. Each user can contribute to the sensor data using only one device. The user ID is the MAC address of a participating mobile device. The data processor stores sensor data and generates information using such data. The observer notices nearby users and detects the Sybil nodes. If the observer detects the Sybil nodes, the observer reports the detected user to the data processor. The data providers collect and provide sensing data to the data processor. As a reward for the sensing data contribution, the data provider obtains incentives from the data processor.

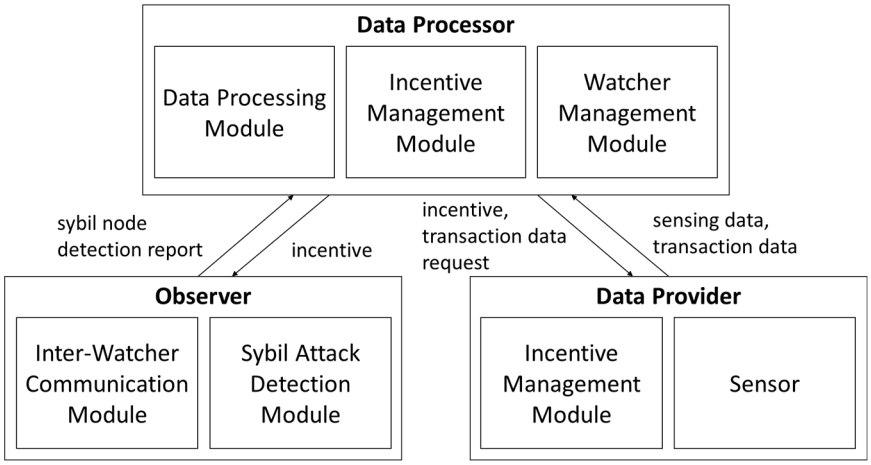

Figure 3. Proposed system structure.

The data processor includes a data processing module, an incentive management module and an observer management module. The data processing module stores the sensing data and generates information by aggregating and processing such data. The incentive management module manages the users' own incentive quantity and handles incentive transactions. The data processor uses incentive transaction records for an incentive transaction record-based user validation. The observer management module stores the observers' information and communicates with the observers to obtain the Sybil node detection results. Users can be an observer by providing personally identifiable information (e.g., phone numbers and national identification numbers) to the data processor. The data processor manages the information of observers with the observer management module. If the generated information is different from real-world information, the data processor can add an observer of an invalid task to the blacklist.

The observer modules include an inter-observer communication module and Sybil attack detection module. The Sybil attack detection module captures nearby Wi-Fi packets using a monitor mode supported NIC used to collect the Wi-Fi connection/disconnection and RSS data. The inter-observer communication module periodically broadcasts the observer location and collected RSS data. In addition, it listens for other the broadcasts of other observers and chooses the Sybil node detection method depending on the number of observers in the area. If there are no other nearby observers, the observer conducts a Wi-Fi association/disassociation data-based Sybil node detection. Otherwise, the observer conducts an RSS-based observation.

The data provider includes a sensor device and an incentive management module. The data provider collects sensing data with the sensor device and sends the data to the data processor. The incentive management module manages the incentive transaction records and for the incentive transaction record-based user validation, the incentive transaction records are applied.

3.2. Attack Model

The proposed system limits only one device per user to participate in the sensing data contribution. The user ID is the MAC address of the device. Thus, the attacker should extort other user IDs and replace the NIC's MAC address with these IDs to conduct a Sybil attack. The attacker uses a four-step automation to improve the efficiency of a Sybil attack: disable the NIC, change the MAC address, enable the NIC and contribute to the sensing data.

With the proposed system, the observer uses physical wireless signal information (i.e., RSS and Wi-Fi connection/disconnection data) to detect the Sybil node. The data processor uses incentive transaction records to verify if the data provider is the valid owner of the ID. If the observer detects a Sybil attack first, the data processor does not have to verify all users. As a result, the data processor can reduce the amount of wasted resources that occur through Sybil node detection.

3.3. Sybil Attack Detection

The Sybil attack detection mechanism proposed herein consists of two stages (see Figures 4 and 5). In Figure 4a, the observer captures nearby Wi-Fi packets and (Figure 4b) broadcasts its location and RSS data to the devices in the area. (Figure 4c) The observer then checks whether other observers are present in the area. If another observer is nearby, (Figure 4d) its Sybil node is detected using an RSS-based observation. Otherwise, (Figure 4e) the Sybil node is detected using Wi-Fi connection/disconnection data. After the detection has finished, (Figure 4f) the detection results are reported to the data processor. (Figure 4g) If a malicious user exists, the data processor conducts an incentive transaction records-based user verification.

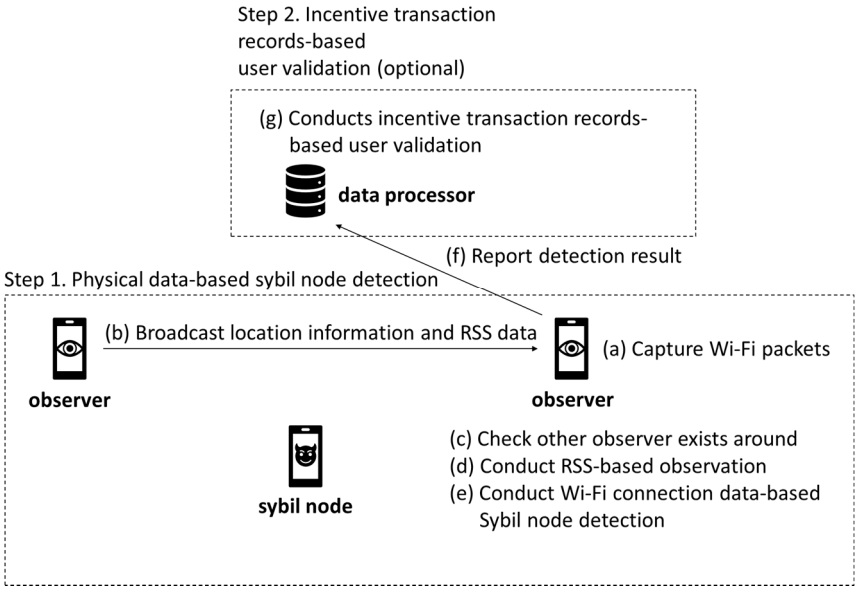

Figure 4. Sybil attack detection flow: (**a**) Observer Wi-Fi packets monitoring; (**b**) Observer information broadcasting; (**c**) Observer existence checking; (**d**) RSS-based observation; (**e**) Wi-Fi connection data-based detection; (**f**) Detection result reporting; (**g**) Incentive transaction records-based user validation.

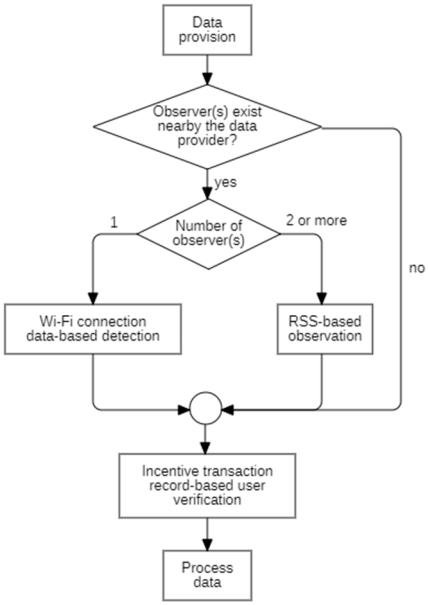

Figure 5. Sybil attack detection mechanism selection.

The proposed mechanism uses three different Sybil attack detection methods (see Figure 5), namely, Wi-Fi connection data-based detection, an RSS-based observation and an incentive transaction record-based user verification. The method applied by the observers and the data processor depends on the existence of the observer near the data provider. If there are no observers nearby when the data provider provides the sensing data, the data processor starts a direct incentive transaction record-based user verification. Otherwise, if one or more observer is present, they find another nearby observer. If there are no other nearby observers (and only one observer is present around the data provider), the observer conducts a Wi-Fi connection data-based detection. Otherwise, an RSS-based observation is conducted with other observers. The observers report the detection results to the data processor and the data processor conducts an incentive transaction-record based user verification only to the reported users.

To implement physical data-based Sybil attack detection, an observer should able to monitor a physical wireless signal. Thus, we assume that the Sybil node sends fake sensing data through a monitorable wireless network (e.g., Wi-Fi or LoRa) and not an unmonitorable wireless network (e.g., WCDMA or LTE).

3.3.1. Wi-Fi Connection Data-Based Detection

A malicious user spoofs the NIC's MAC address with another user's ID. During this process, the one device disconnects from the network, immediately followed by another device connecting to the network, which occurs in a repeated pattern. The observer captures nearby Wi-Fi packets and finds the pattern to detect a Sybil attack (see Figure 6). The observer recognizes a new device connection with an association, authentication packets and a device disconnection with a disassociation and de-authentication packets.

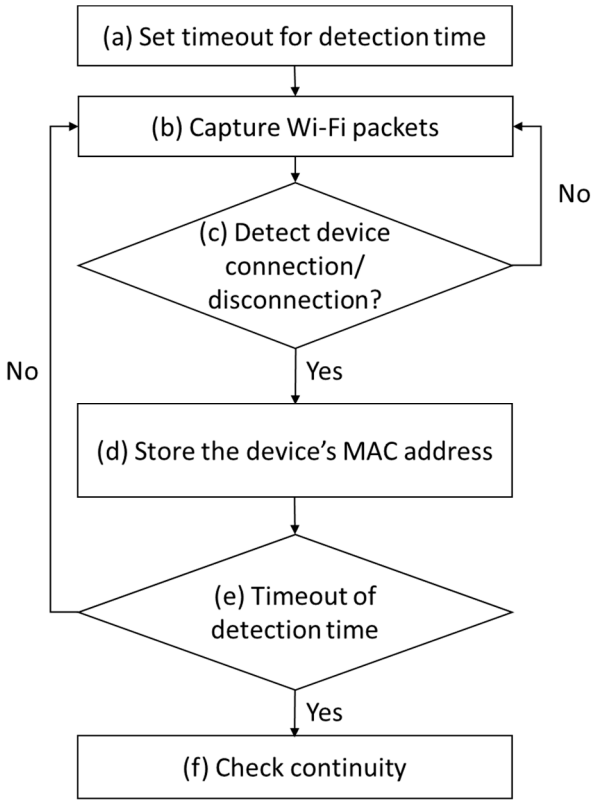

Figure 6. Wi-Fi connection data-based detection flow. (**a**) The observer sets the detection time and (**b**) captures nearby Wi-Fi packets. (**c**) When the device connection/disconnection is detected, (**d**) the MAC address of the device is stored. (**e**) After the detection time, (**f**) the continuity of the stored devices is checked. Such continuity indicates that the devices satisfy the pattern that appears when Sybil attacks are conducted (i.e., a new device connection after a device disconnection).

Connections 2, 4 and 5 satisfy the continuity (see Table 2). For connection 2, the new device bb:bb:bb connects to the network after aa:aa:aa has disconnected. For connection 4, another new device dd:dd:dd connects to the network after bb:bb:bb has disconnected. In addition, aa:aa:aa, bb:bb:bb and dd:dd:dd are sequentially connected and disconnected. This pattern appears when one user repeatedly changes the NIC's MAC address. Connection 1 does not have a former disconnected device, and connection 3 does not have continuity with other connections. Thus, the observer determines the devices aa:aa:aa, bb:bb:bb, dd:dd:dd and ee:ee:ee are Sybil nodes.

Table 2. Example of continuity.

No.	Disconnected	Connected
1		aa:aa:aa
2	aa:aa:aa	bb:bb:bb
3	cc:cc:cc	dd:dd:dd
4	bb:bb:bb	dd:dd:dd
5	dd:dd:dd	ee:ee:ee

The proposed system uses a tree structure to calculate the continuity depth (see Figure 7). Each node records the device addresses and the empty nodes are pointers indicating where the address of a new device is recorded. In Figure 7a, device A has connected and in Figure 7b, device B has connected. An empty node has changed to node B and an empty node has been generated as a sibling node of node B. In Figure 7c, device A has disconnected. When a device disconnects, an empty child node is added to the disconnected device. In Figure 7d, device C has connected. All empty nodes have changed to node C and empty nodes are generated as the sibling node of node C. In (d), the depth of the tree has continuity. In Figure 7, the continuity depth is 2, and devices A and C satisfy this continuity.

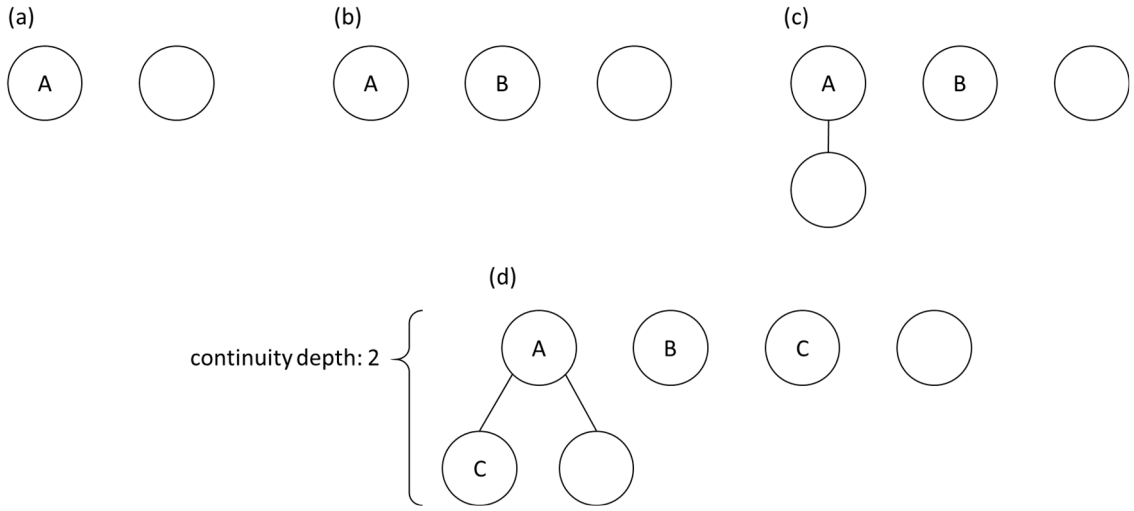

Figure 7. Continuity depth calculation: (**a**) Continuity graph after device A is connected; (**b**) Continuity graph after device B is connected; (**c**) Continuity graph after device A is disconnected; (**d**) Continuity graph after device C is connected.

In a location with a large floating population, Wi-Fi connection data-based detection may judge a valid user as a malicious user. The possibility that two or more users accidentally satisfy the continuity increases as the floating population increases. However, an incentive transaction-record based user validation can decrease the false alarm rate by a significant amount.

3.3.2. RSS-Based Observation

If other observers are in the area, they can detect a Sybil attack by sharing their location and RSS data (see Figure 8). As shown in Figure 8a, the distance from an observer to the data provider is calculated using RSS data. In Figure 8b, circles are drawn using as radius as distance to the data provider centered at each observer's location. In Figure 8c, the points where the circles meet are candidate locations of the data provider. If two observers are nearby, the observers can obtain two candidate locations—and if three or more observers are present, the observers can obtain an accurate candidate location. The observers can determine that a data provider is a Sybil node when two or more data providers send data from the same location.

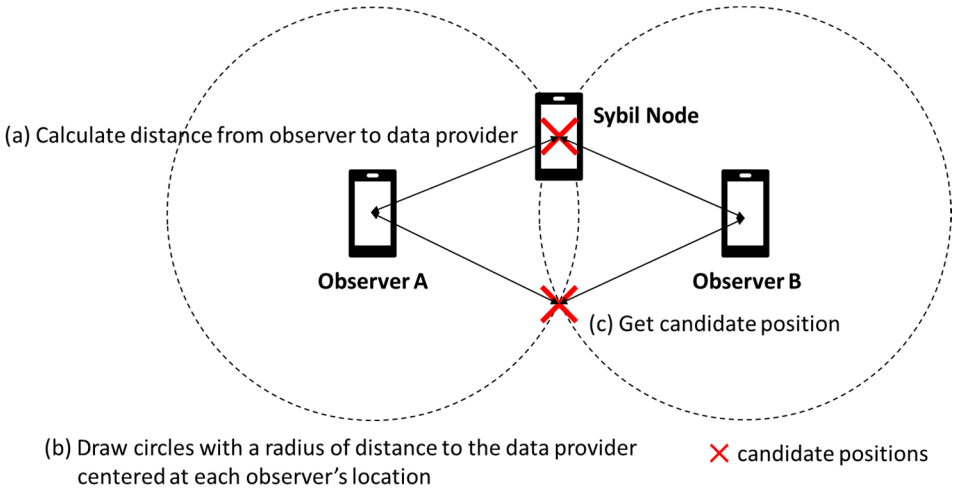

Figure 8. RSS-based observation with two observer nodes: (**a**) Calculating distance to data provider; (**b**) Calculating available positions of data provider; (**c**) Get common candidate positions of data provider.

When there are only two observers in the same location, the false alarm rate can appear to be high because there are two candidate locations. However, this is not a significant problem because the data processor conducts an incentive transaction record-based user validation for the reported users. An incentive transaction-record based user validation can decrease the false alarm rate to a meaningful level.

3.3.3. Incentive Transaction-Record Based User Verification

The data processor can use the incentive transaction records to verify that the data provider is a valid owner of an ID (see Figure 9). As shown in Figure 9a, the data processor obtains the data provider's incentive transaction records from the incentive management module and as shown in Figure 9b, a random duration is set. As shown in Figure 9c, the data processor requests incentive transaction records during a set duration from the data provider and as indicated in Figure 9d the data provider provides incentive transaction records to the data processor. Finally, as shown Figure 9e, the data processor compares the incentive transaction records from the incentive management module and the data provider. If the data provider is not a valid owner of the ID, the data will not match.

When applying incentive transaction records of a past duration, user verification does not require any user-identifying information to be requested. Moreover, because the data processor sets the duration at random, a malicious user should monitor the victim's incentive transaction for a long duration. Even if an attacker succeeds at extorting all incentive transaction records of the victim, the data processor can notice that the victim has been extorted by the attacker because the information generated using the victim's sensing data differs from the real-world information. As a result, the attack cost increases, and profiting becomes difficult.

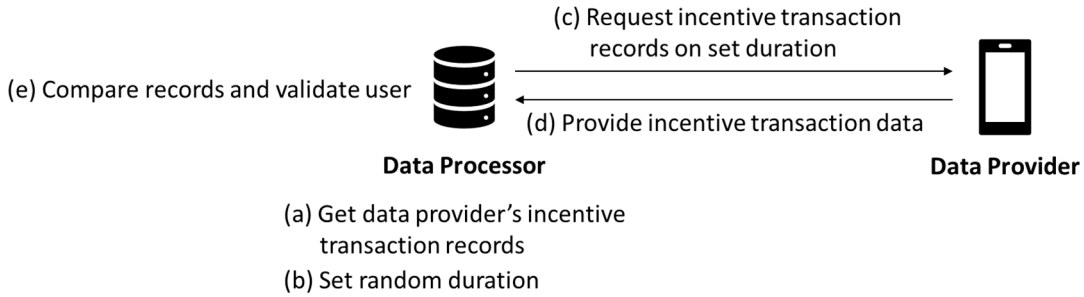

Figure 9. Incentive transaction-record based user verification.

4. Experiment

We implemented a Wi-Fi-connection-based Sybil attack detection model to analyze the performance of the proposed mechanism. The performance of the Wi-Fi-connection-based Sybil attack detection model can appear to differ depending on detection time and continuity depth required. We analyzed the detection and false alarm rates depending on the change in requested continuity depth and detection time to find the optimized value. The false alarm rate is the ratio of nodes that the detection model incorrectly determines as Sybil nodes. We set the detection time to 2, 3 and 5 min and the required continuity depth to 2, 3 and 5, respectively. A high detection time of greater than 5 min and a continuity depth of over 5 can cause a high false alarm rate and a low detection rate. We use CYW43455 [17] NIC, which is included in Raspberry Pi 3B+ and TShark [18], for capturing nearby Wi-Fi packets. We generated the continuity depth calculation code in Python and collected 400,000 Wi-Fi packets of data from Pyeongtaek station, Republic of Korea, to evaluate the model performance. We collected packets at 2 PM on a Sunday to evaluate the model performance in a crowded environment. We simulated the Sybil attack using an automated code written in Python. The automated code conducts a four-step Sybil attack (i.e., disable the NIC, replace the MAC address, enable the NIC, and provide fake sensing data) repeatedly. We implemented incentive transaction record-based user validation to demonstrate that the proposed mechanism could detect the Sybil attack with lower central overhead than existing centralized Sybil attack detection mechanisms (i.e., certification [11] and trustiness scoring [13]). As comparison mechanisms, we implemented certification mechanism that the data provider signs and transmits the sensing data, and the data processor validates the signature. We also implemented trustiness scoring that the data processor calculates behavior-based trustiness score for each sensing data provision. All mechanisms are implemented in Python.

The model requiring a continuity depth of 2 for 2 min demonstrates the highest detection rate of 99.2% (Figure 10). By contrast, the model requiring a continuity depth of 5 for 2 min achieves the lowest detection rate of 93%. Except for the model requiring 5 continuity nodes for 2 min, the detection rate and detection time are inversely proportional. In an area with a large floating population, some normal nodes satisfy a high continuity depth by chance. A longer detection time can increase the possibility of a wrong detection. With the model requiring a continuity depth of 10 for 2 min, the detection rate drops sharply to 93%. We interpret this case as a result of a higher continuity depth required compared to a shorter detection time. In the experiment, the automated Sybil attack code occasionally fails to provide 10 or more fake sensing data within 2 min. In addition, the model requiring a continuity depth of 10 in 2 min determines the existence of a Sybil attack using 9 IDs under a normal situation, which may be the reason for the increased detection rate. The model requiring a continuity depth of 2 for 5 min has the highest false alarm rate of 16.9% (see Figure 11). The model requiring a continuity depth of 5 for 2 min has the lowest false alarm rate of 1.0%. The false alarm rate and continuity depth required are inversely proportional, and the false alarm rate and detection time are proportional. A false alarm occurs when a device enters a wireless signal range right after another device has left the range accidentally. Thus, a false alarm mostly occurs in a location with a large floating population.

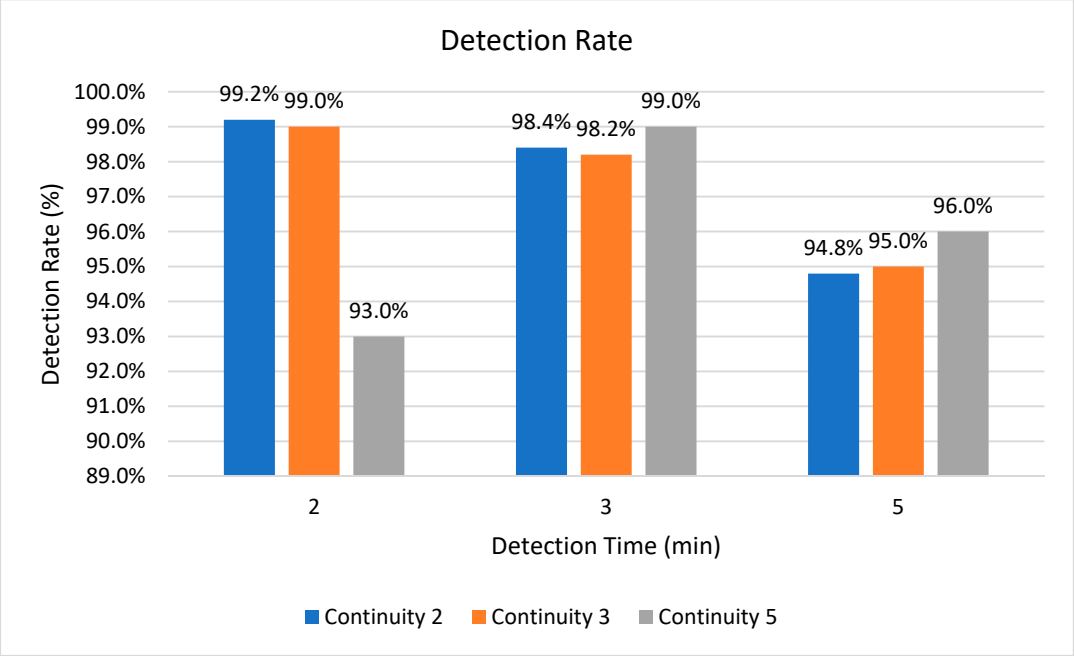

Figure 10. Detection rate.

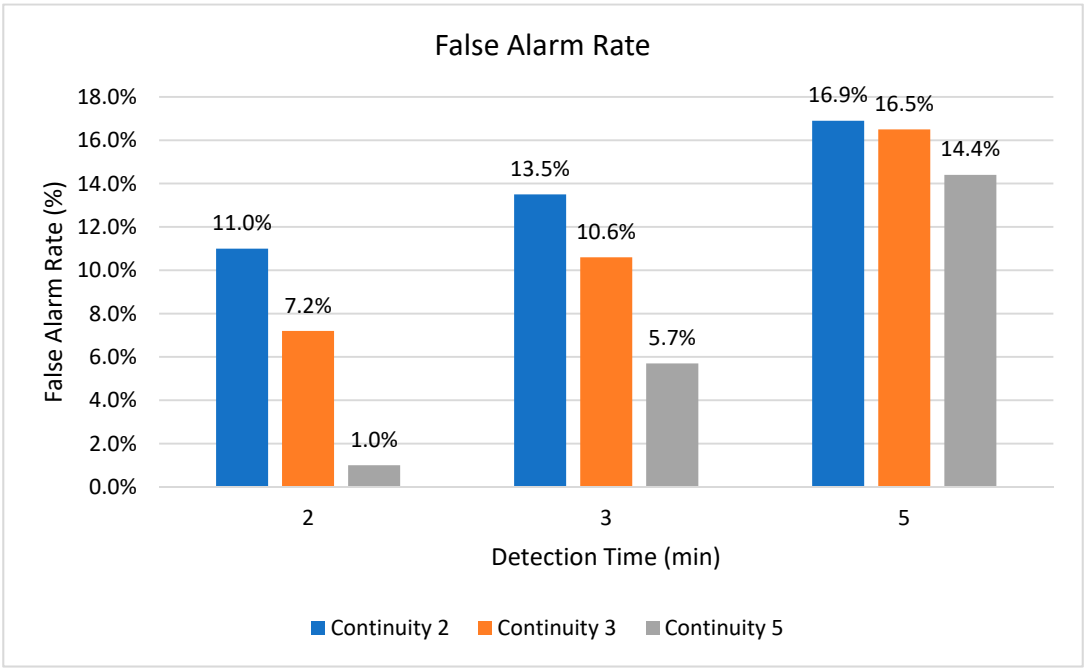

Figure 11. False alarm rate.

The detection rate is the most crucial factor in evaluating the performance of the detection model. If the detection rates of two or more models are similar, the performance of the model is determined based on the false alarm rate and the detection time. Based on these criteria, the detection model requiring a continuity depth of 3 for 2 min is the most efficient.

As shown in Figure 12, the incentive transaction record-based user validation proposed in this paper can detect Sybil attack with lower processing overhead than other existing central detection mechanisms. For the provision of 1000 sensing data, the certification mechanism took 4.29 s to validate the sensing data. In the case of trustiness scoring, the processing speed under 200 data provision was similar to that of the proposed mechanism. However, as the number of sensing data provisions increases, the processing speed of trustiness scoring mechanism also increases rapidly. This can be analyzed because trustiness scoring mechanism calculates the reliability compared to other previously provided sensing data, rather than checking only specific sensing data. In the case of the proposed mechanism, it showed a processing time of 1.95 s for the provision of 1000 sensing data, and thus it was possible to detect a Sybil attack faster than the certification and trustiness scoring mechanisms. The results of these experiments show that the proposed mechanism can detect Sybil attack for providing a large number of sensing data faster than the existing centralized detection mechanisms. Moreover, the data processor in the proposed mechanism validates only suspicious sensing data detected by the first stage detection (i.e., Wi-Fi connection data-based detection or RSS-based observation), and thus additional overhead reduction can be expected.

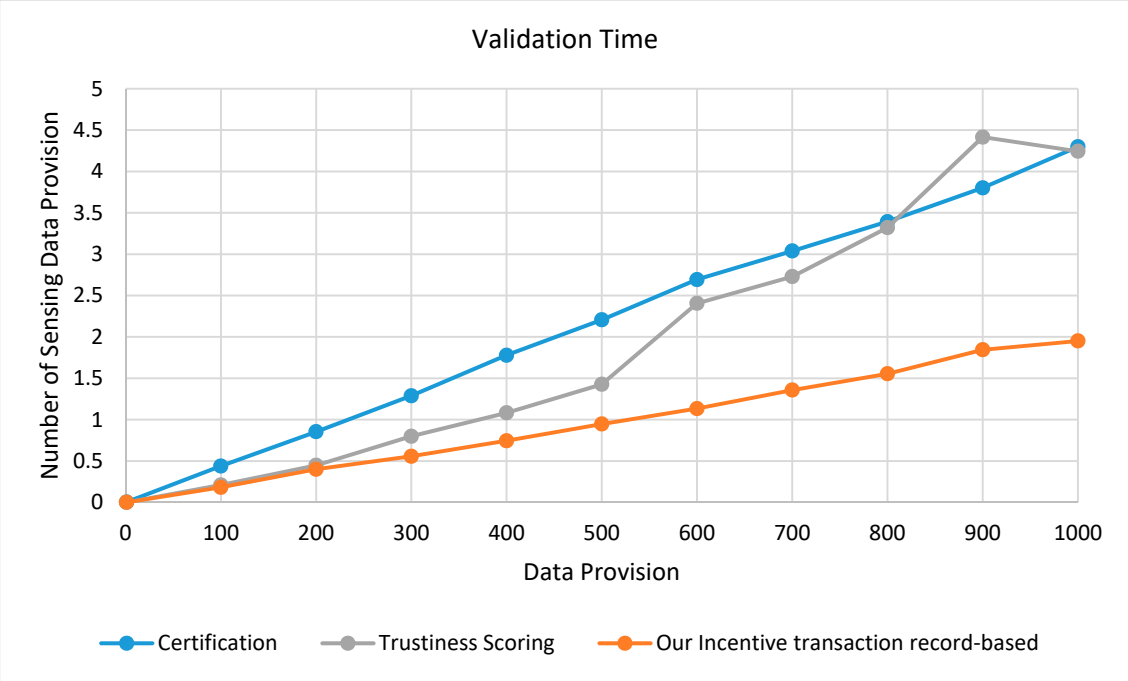

Figure 12. Validation time.

5. Conclusions

In this paper, we propose an observer-assisted Sybil attack detection mechanism for mobile crowdsensing and show the feasibility of the system by implementing a Wi-Fi connection data-based Sybil attack detection module. The mobile crowdsensing system anonymously collects sensing data from the mobile devices of users, which have a high location fluidity. To implement a traditional observation mechanism for mobile crowdsensing, a central data processor should be installed in monitoring devices throughout a location where a data provider is present. However, this entails high installation and maintenance costs. In addition, data providers may be hesitant to reveal their location data to a data processor. Thus, traditional Sybil attack detection mechanisms developed for a WSN are difficult to adopt for mobile crowdsensing. The proposed system can preserve the privacy of participants by distributing an observer role to the users. In such, the central data processor cannot access the accurate location data of the data provider. Moreover, the data processor can minimize overhead occurring through Sybil-node detection. If the central data processor conducts incentive

transaction record-based user validations for all data providers, the processing overhead doubles, as compared with a system without a Sybil attack detection mechanism. Observers can reduce the validation overhead of the central data processor to a meaningful level.

Author Contributions: J.Y. and M.K. completed this work. J.Y. developed the proposed system and implemented and experimented the prototype of the proposed system. M.K. organized for designing and developing the proposed system in this work and guided this whole work as a corresponding author. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) [No. 2018R1A2B6009620].

Conflicts of Interest: The authors declare that there is no conflict of interest regarding the publication of this paper.

References

1. Ganti, R.; Ye, F.; Lei, H. Mobile crowdsensing: Current state and future challenges. *IEEE Commun. Mag.* **2017**, *49*, 32–39. [CrossRef]
2. Matarazzo, T.J.; Santi, P.; Pakzad, S.N.; Carter, K.; Ratti, C.; Moaveni, B.; Osgood, C.; Jacob, N. Crowdsensing Framework for Monitoring Bridge Vibrations Using Moving Smartphones. *Proc. IEEE* **2018**, *106*, 577–593. [CrossRef]
3. Ludwig, T.; Reuter, C.; Siebigteroth, T.; Pipek, V. CrowdMonitor: Mobile Crowd Sensing for Assessing Physical and Digital Activities of Citizens during Emergencies. In Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems—CHI '15, Seoul, Korea, 18–23 April 2015; pp. 4083–4092.
4. Zhang, X.; Yang, Z.; Sun, W.; Liu, Y.; Tang, S.; Xing, K.; Mao, X. Incentives for Mobile Crowd Sensing: A Survey. *IEEE Commun. Surv. Tutor.* **2016**, *18*, 54–67. [CrossRef]
5. Jaims, L.G.; Vergara-Laurens, I.J.; Raji, A. A Survey of Incentive Techniques for Mobile Crowd Sensing. *IEEE Internet Things J.* **2015**, *2*, 370–380. [CrossRef]
6. Newsome, J.; Shi, E.; Song, D.; Perrig, A. The Sybil Attack in Sensor Networks: Analysis & Defenses. In Proceedings of the Third International Symposium on Information Processing in Sensor Networks, Berkeley, CA, USA, 26–27 April 2014; pp. 259–268.
7. Guo, B.; Yu, Z.; Zhou, X.; Zhang, D. From Participatory Sensing to Mobile Crowd Sensing. In Proceedings of the IEEE International Conference on Pervasive Computing and Communication Workshops, San Diego, CA, USA, 18–22 March 2013; pp. 593–598.
8. Piro, C.; Shields, C.; Levine, B.N. Detecting the Sybil Attack in Mobile Ad hoc Networks. In Proceedings of the 2006 Securecomm and Workshops, Baltimore, MD, USA, 28 August 2006; pp. 1–11.
9. Patel, S.T.; Mistry, N.H. A Review: Sybil Attack Detection Techniques in WSN. In Proceedings of the 2017 International Conference on Electronics and Communication Systems, Coimbatore, India, 19–20 October 2017; pp. 184–188.
10. Jan, M.A.; Nanda, P.; He, X.; Liu, R.P. A Sybil Attack Detection Scheme for a Forest Wildfire Monitoring Application. *FGCS* **2016**, *80*, 613–626. [CrossRef]
11. Douceur, J. The Sybil Attack. In *Intl Wkshp on Peer-to-Peer Systems*; Springer: Berlin/Heidelberg, Germany, 2002; pp. 251–260.
12. Margolin, N.B.; Levine, B.N. Quantifying Resistance to the Sybil Attack. In Proceedings of the Financial Cryptography and Data Security; Tsudik, G., Ed.; Springer: Berlin/Heidelberg, Germany, 2008; pp. 1–15.
13. Singh, R.; Singh, J.; Singh, R. A Novel Sybil Attack Detection Technique for Wireless Sensor Networks. *Adv. Comput. Sci. Technol.* **2017**, *10*, 185–202.
14. Vaghefi, R.M.; Gholami, M.R.; Storm, E.G. RSS-based sensor localization with unknown transmit power. In Proceedings of the 2011 IEEE International Conference on Acoustics, Speech, and Signal Processing, Pargue, Czech Republic, 22–27 May 2011; pp. 2480–2483.
15. IEEE 802.11-2016 Specification. Available online: https://standards.ieee.org/content/ieee-standards/en/standard/802_11-2016.html (accessed on 23 October 2019).
16. IEEE 802.15.1-2005 Specification. Available online: https://standards.ieee.org/standard/802_15_1-2005.html (accessed on 23 October 2019).

17. CYW43455 Datasheet. Available online: https://www.cypress.com/file/358916/download (accessed on 30 October 2019).
18. TShark Documentation. Available online: https://www.wireshark.org/docs/man-pages/tshark.html (accessed on 30 October 2019).