Forensic Verification of Health Data From Wearable Devices Using Anonymous Witnesses

Muhammad Siddiqi, Syed Taha Ali, and Vijay Sivaraman

***Abstract*—The use of wearable devices, such as smartwatches, glasses, clothes, and fitness bracelets is increasing at an ever-growing pace. Major corporations and insurance companies have started mandating their use for their employees and clients. Data from such devices have begun to feature in settlement claims and as evidence in courts as well, requiring it to be irrefutable and tamper-proof. Lack of protection for personal data as well as the contextual information such as location tracking and its use by law enforcement agencies is raising serious privacy concerns among the general public and civil liberty advocates. In this article, we propose a novel scheme to secure the wearable sensor's communication through its *crowdsourced logging* by neighboring wearable and smart devices called *witnesses* preserving the contextual information (such as time and location) as well. To ensure witness privacy, gateway and witness devices use the reciprocity property of wireless medium between them to generate pairs of closely matching link signatures, which not only provide the proof of presence for the witnesses in the vicinity but also act as their time-varying pseudonyms. We demonstrate the feasibility and efficacy of our scheme through the prototype implementation using real wireless devices, and via simulation and experimental results.**

***Index Terms*—Body sensor networks, crowdsourcing, forensics, Internet of Things, network security, wearable sensors.**

I. Introduction

THE USE of the wearable technology, such as smartwatches, activity trackers, fitness monitors, and healthcare devices, is becoming commonplace in daily life. Devices, such as the Fitbit, Jawbone, Apple Watch, and Nike Fuelband, have a variety of built-in sensors, which measure users' location and physical activity in real time and relay this data to the cloud where various trends about our health and habits may be identified, tracked, and shared. According to a report by International Data Corporation (IDC) [1], around 113.2 million wearable devices were shipped in 2017 and this number is expected to double to an estimated 222.3 million units by 2021.

Manuscript received July 25, 2019; revised January 5, 2020; accepted March 10, 2020. Date of publication March 24, 2020; date of current version November 12, 2020. This work was supported by the Australian Research Council's Discovery Project under Grant DP150100564. *(Corresponding author: Muhammad Siddiqi.)*

Muhammad Siddiqi is with the Department of Business Information Systems, Australian Institute of Higher Education, Sydney, NSW 2000, Australia (e-mail: m.siddiqi@aih.nsw.edu.au).

Syed Taha Ali is with the School of Electrical Engineering and Computer Science, National University of Science and Technology, Islamabad 44000, Pakistan (e-mail: taha.ali@seecs.edu.pk).

Vijay Sivaraman is with the School of Electrical Engineering and Telecommunications, University of New South Wales, Sydney, NSW 2052, Australia (e-mail: vijay@unsw.edu.au).

Digital Object Identifier 10.1109/JIOT.2020.2982958

Employers and health insurers are actively developing strategies to integrate this new technology into their policies. Retail giant target recently distributed Fitbit trackers to 335 000 U.S. employees and IBM to 40 000 employees [2]. Insurance firms, such as John Hancock Insurance [3], United HealthCare Group [4], and MLC [5], now offer their customers free wearable sensing devices together with financial incentives and discounts on premiums to keep active and meet wellness goals. Self-insured employers, such as Appirio, have saved significantly on insurance premiums after sharing detailed activity records of employees with their insurers [2].

These devices also provide an investigative advantage to law enforcement agencies. Data from wearable devices have begun to feature as evidence in courts. For instance, a Canadian law firm used a client's Fitbit activity records in a personal injury compensation case to successfully prove that she had reduced activity levels even after four years of suffering injury [6]. Likewise data from a man's pacemaker have been used in a court in Ohio to discredit his testimony and charge him with aggravated arson and insurance fraud [7]. During the time he claimed to be asleep, his activity data revealed that he was physically active and, as per a cardiologist's testimony, it was highly improbable that he personally rescued a large number of heavy household items as per his statement.

Indeed, the overwhelming popularity of sensing devices deployed in the home and on the body has prompted some forensics experts to contend that criminal cases will now consist of a strong "digital component" [8]. Efforts are therefore underway to assess the legal position of these devices as admissible alibis in court [9], [10]. The major concern of stakeholders, i.e., doctors, users, insurers, and prosecutors, is the reliability and security of these devices.

Researchers have demonstrated the ease and extent of hacking wearable devices and tampering with their data. For instance, in our previous work, we compromised the Withings Pulse tracker and showed how users could easily backfill activity records (e.g., step count) by tampering with the clock on their mobile phones [11]. Fereidooni *et al.* [12] hacked Fitbit and demonstrated that malicious users could generate false activity records with altered timelines. These, and several similar examples (such as [13]–[16]), reinforce the need to protect the data these devices are collecting about our health and activity.

In prior work, we introduced a novel *crowdsourced logging* technique in which wireless devices in the vicinity can confirm the presence of a wearable device and probabilistically

attest to its data transmissions [17]. This article was motivated by the fact that IoT devices are being deployed within buildings, vehicles, and carried by people in ever-growing numbers, and this increasing density enables these devices to overhear and record wireless transmissions of neighboring devices. Our solution enables opportunistic binding of a device's communications to location and time and allows forensic experts to verify a wearable's whereabouts and readings by querying records generated by neighboring devices (called *witnesses*).

Whereas this approach shows promising results, the increasing adoption of wearable devices raises considerable privacy concerns. These concerns are being articulated by the research community [9], [18] as well as public interest groups and watchdog bodies [19]. In a recent survey of 1000 British workers [20], 67% of respondents expressed the fear that wearables would usher in a "big brother" surveillance culture.

Motti and Caine [21] documented that users are also particularly wary of "surveillance," i.e., the practice whereby activities are recorded by participants themselves, typically using wearable devices. A key concern here is that in group scenarios, the data of other participants may be recorded and shared online without their express knowledge or consent.

To reconcile these conflicting concerns of verification and user privacy, in this article, we introduce the concept of *pseudonymous witnesses*. For this purpose, we leverage another recent innovation in the research literature, which relies on the spatiotemporal characteristics of the wireless link to generate unique and symmetrical *link signatures* between two communicating parties [22]. These signatures augment our crowdsourced logging solution by effectively masking the identity of witness devices in the environment while at the same time, confirming their presence in the environment.

We make the following specific contributions in this article.

1) We describe a scheme that attests to the communications of wearable devices in a pseudonymous and verifiable manner by relying on wireless smart devices in the vicinity.
2) We develop an analytic model for forensics investigators to quantify trust in these witness records and we demonstrate how to tune system design parameters for various witness devices based on their capabilities and environment.
3) We implement our scheme on real wireless devices (MicaZ motes) and demonstrate its feasibility and efficacy with results from a real office environment scenario.

The remainder of the article is organized as follows. Section II presents representative examples to motivate our solution. Section III summarizes the link signature and crowdsourced logging primitives upon which we build our solution. In Section IV, we describe in detail the protocols comprising our solution, followed by an analysis of its security properties. Section V presents the analytical trust model for witness records and tuning of design parameters of our solution followed by Section VI which discusses the experiments and their results. We conclude in Section VII.

II. Background

A. Motivation

In this section, we present certain representative scenarios to motivate our pseudonymous logging scheme. These scenarios, based on real-life instances, indicate the potential for abuse of data generated by wearables as well as the novel challenges faced by investigators working with these devices.

The ecosystem typically consists of the following parties: users wear these devices to monitor health or fitness. These devices typically use a base station device, such as a smartphone or an Internet gateway to communicate their data to servers in the cloud. Here, the data may be accessed by healthcare services and medical practitioners and mined using machine learning algorithms to identify patterns and trends. Insurance providers may also access this data to tailor policies for their clients and check for compliance. Ideally, gateway devices also maintain detailed logs of network activity to assist forensics investigators in the event of an incident.

1) Scenario #1: Alice is murdered in her home at night. According to the statement of her husband Bob, a masked intruder broke into their house and knocked Bob unconscious and then killed Alice. To corroborate his account, Bob shows his head wound and offers the data from his wearable fitness monitor. However, investigators do not find compelling evidence for a break-in and they also discover that Bob has a financial interest in his wife's death in the form of a large insurance payout. They are therefore forced to consider the possibility that Bob may have killed Alice and inflicted the head wound himself and tampered with the data of his wearable device.

This case is inspired by a real-life murder case from Connecticut in 2017 [23]. However, in that instance, it was data collected from the murdered wife's Fitbit exercise tracker which discredited her husband's testimony. The device showed that, during the time the husband claimed they were attacked and then restrained, his wife was walking around the house and was far more active than compared to her husband's account.

However, tampering with wearables to create a desired outcome is not very difficult. Due to their small form factor these devices have restricted processing capabilities and are not able to run comprehensive security protocols. As noted earlier, Fereidooni *et al.* [12] have demonstrated the ease with which malicious users can inject false activity records into their Fitbit data. Wearables also typically depend on paired smartphones for clock synchronization, thereby opening up new attack vectors. We, in our previous work, have described attacks where users can backfill activity records in Withings Pulse trackers [11].

In less technical yet clever ways, users have already been doctoring their activity records by attaching their Fitbits to their pets, to electric fans, to active toddlers, etc., usually with an intent to exaggerate their activity to their peers, to win office competitions, or to secure insurance benefits [24]. These examples raise serious concerns about using data generated by wearable devices as admissible evidence in court proceedings.

2) Scenario #2: Alice suffers a debilitating work injury and, as a result, her physical activity is extremely limited after

the incident. She files a claim for worker compensation but her claim is rejected and she takes her insurer to court. To support her claim, Alice presents activity records from her wearable device to prove her movement has been extremely limited after the incident. However, the insurance company presents alternative records acquired from the servers belonging to Alice's healthcare service which negate her account and display normal activity levels. Both sides accuse each other of tampering with the data. The court needs to ascertain which party is telling the truth.

This scenario draws on the real-life example mentioned in Section I, where a party in Canada won a personal injury compensation claim using her Fitbit activity records to demonstrate continued reduced activity levels years after the accident. Real-life instances also indicate that data tampering in these cases is not uniquely associated with users. Insurers too have the incentive to tamper with client records. In 2016, Australian insurer, CommInsure, was found tampering with the medical record of their clients, in collusion with healthcare providers, in order to dismiss their insurance claims [25].

To motivate anonymization of witnesses, we consider another emerging trend in law enforcement, one in which detectives are now using location traces from the cloud to identify persons of interest in criminal investigations. In a recent murder investigation in Raleigh, North Carolina, police used search warrants to force Google to release Google account details of all mobile users in a particular geographic zone (demarcated by GPS coordinates) at specific points in time [26]. Raleigh police have made similar inquiries in at least four instances in 2017 to investigate the homicide, sexual assault, and arson [27]. For the arson investigation, police requested Google to disclose "anonymized information" about users' accounts along with their timestamped location coordinates. According to detectives, this strategy will help narrow down the list of suspects, for whom they will then demand Google to release user names, details, and, as they wrote, "contextual data points with points of travel outside of the geographical area" for extended periods of time.

Public and civil liberty advocates have aired serious concerns over such practices, noting that these searches covered several acres of area, including several homes and businesses, and revealed private information of several users [27]. There is also the concern that this practice is an imposition on people who may be reluctant to participate in criminal investigations. Indeed there have been many instances in the past where legitimate witnesses did not come forward due to unwillingness to deal with the police, or fear of loss of reputation or loss of life. Examples include an instance where two teenaged boys were murdered in a house party attended by approximately a hundred people [28], and a case where a seven-year-old girl was murdered during a gang fight in front of a crowd that did not testify [29].

These and similar cases have motivated the introduction of the Criminal Evidence (Witness Anonymity) Amendment Bill of 2018 in U.K. parliament, which would allow a judge in an investigation to keep a witness hidden from the jury as well as the defendant [30]. Our solution assists the forensic investigator to determine the ground truth in these new emerging scenarios while assigning potential witnesses with pseudonyms to protect their identity.

B. Prior Work

In this section, we discuss prior work in this domain.

Forensics investigators rely on provenance solutions (such as event logs or cryptographic mechanisms) to determine the ground truth in their investigations. Provenance may be defined simply as metadata that tracks the origin and evolution of a data item within a system. In terms of wearable devices, this may consist of the time, location, and context within which a data reading has been generated and how that item was then transmitted, stored, and shared.

However, the granularity with which the provenance is gathered depends upon the application and device capabilities. For example, Pohly *et al.* [31] presented a high-fidelity kernel-level provenance system that is capable of providing fine-grained forensic analysis for enterprise-level networks. However, for resource-constrained wearable devices, such detailed operation logs and analysis are hard to achieve and we may have to rely on digital signatures and timestamps. Moreover, it may not be possible to express provenance in binary in certain cases such as large multihop sensor networks where it might make more sense to give confidence in sensor data in the form of a probability value [32] or a trust score [33], [34].

Braun *et al.* [35] have convincingly argued that the security of this provenance data is a vital concern with its own distinct threat model. This concern is particularly highlighted when we consider that data from wearables are now considered admissible evidence in courtroom proceedings [9].

In this context, researchers have designed tools to demonstrate the feasibility of misleading forensics investigations. Even unskilled users can easily create false alibis on computers [36] or smartphones [37] using automation tools that generate fake activity patterns on user devices as well as online on social media platforms. On the other hand, Castiglione *et al.* [38] have shown that digital evidence may be selectively and securely deleted from a computer with little effort. As we mentioned earlier in this article, many wearable devices have been hacked [39]. Fereidooni *et al.* [12] have described how activity traces may be forged on a Fitbit tracker.

Secure logging solutions typically provide security properties such as confidentiality, integrity, and auditability during the storage and transmission of sensitive data. A comprehensive review of these techniques may be found in [40]. However, the vast majority of these techniques only focuses on individual security properties and specific use cases, and entail power and compute requirements that are far too expensive for resource-constrained wearable devices.

We are aware of only two contributions in the literature for holistic and secure logging solutions aimed at wearable devices. First, De La Piedra *et al.* [41] described a system in which messages from multiple wearable devices are linked together by a gateway or base station by hash chaining their relative timestamps to form a "threaded authentication tree" (an extension of the Merkle tree). A central server higher up in the hierarchy collects and binds together messages collected

from multiple gateways. This scheme claims to provide authentication, confidentiality, and data integrity. However, it does not provide protection in the case where the gateway or central server is malicious or if different parties collude to tamper with the data.

A second relevant scheme is a lightweight logging solution we proposed, which relies on witness devices in the vicinity to ensure a range of security properties [17]. We discuss this solution (and its shortcomings) in detail in Section III.

1) Data Provenance Solutions: Provenance solutions in the literature also typically focus on individual security properties. For instance, Hasan *et al.* [42] presented WORAL, a secure location-stamping solution for mobile nodes that also relies on witness devices in the vicinity to generate tamper-evident location proofs. Users and witnesses register their true identity with location authorities but communicate anonymously with each other using cryptographic identities. The scheme provides the location proof for the users running a mobile app in their smartphones from the volunteered co-located witnesses running the same app in their smartphones. The scheme requires a location authority in the area that supports the service and communicates with the service provider. Our scheme not only attests to the location of the user but also verifies the physiological data, such as heart rate and blood pressure, and any other contextual data being produced by the sensory network in the area and does so in a simpler manner. Our scheme is not limited to only human worn or carried devices but utilizes stationary smart devices in the vicinity for additional proofs regarding the location and data verification.

Shebaro *et al.* [43] described a path-verification solution for multihop sensor networks where network nodes insert identifying information in Bloom filters appended to each data packet that they forward in the network. This approach, extended by Sultana *et al.* [44], [45], can identify malicious nodes in the network but relies on a trusted infrastructure and does not defend against colluding nodes.

In our previous work, we describe a provenance solution specifically for wearable sensing devices which relies on the spatiotemporal characteristics of the wireless radio channel to securely fingerprint communications between two devices [22]. We discuss this solution (and its shortcomings) in detail in Section III.

Our scheme, described in Section IV, provides a broad range of security properties, including data integrity, chronological ordering, localization, auditability, protection against retroactive data tampering, and verification, and protects witness identity in a lightweight manner ideally suited for resource-constrained devices.

III. Approach

Our solution utilizes two building blocks: 1) the *link signature* primitive and 2) the *crowdsourced logging* primitive.

A. Link Signature Primitive

We define a link signature as a unique and closely matching bitstring generated by two communicating parties based on the reciprocity property of the wireless link [46]–[48]. The wireless channel between two devices is symmetric in nature and highly sensitive to the orientation and movement of the devices and objects in the environment. If two communicating parties, Alice and Bob, were to independently measure characteristics of the wireless channel between them, they can each derive a bitstring which uniquely identifies their communication session. As research has demonstrated, the radio signal decorelates rapidly with distance and this bitstring cannot be deduced by an adversary who is situated at a distance greater than one radio wavelength from Alice and Bob (for the 2.4-GHz band, this equates to approximately 13 cm) [49]. This phenomenon has also been investigated for body-worn wearable devices in various studies and has been proposed as a lightweight alternative to cryptographic key exchange protocols [49]–[51].

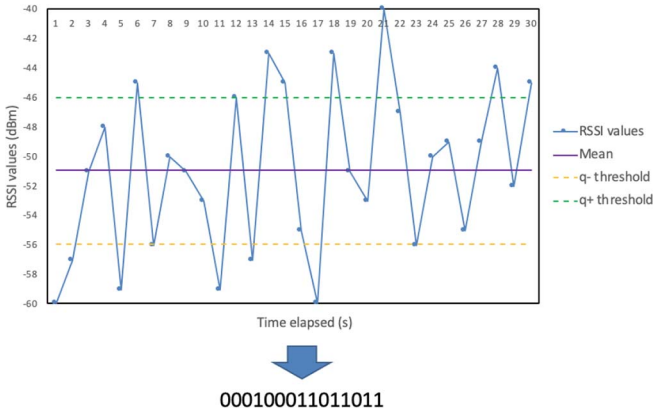

Fig. 1. Example of the level crossing quantization technique.

This process consists of three key steps.

1) *Channel Sampling:* Both communicating parties sample the channel over a period of time to measure channel characteristics. Received signal strength (RSS) is the most convenient and popular feature used for this purpose in the literature. However, other characteristics, such as the magnitude and phase of the wireless channel have been used as well. This process results in a series of raw values on both sides.
2) *Quantization:* Both parties then apply a quantization process to convert the RSS measurements into a bit string. Various quantization mechanisms have been investigated for this purpose, including level crossing and ranking techniques [22]. The choice of the quantization technique depends on application constraints with a typical tradeoff between generating bits faster versus a larger rate of mismatching bits on both sides. An example of digitization using a level crossing quantization technique is given in Fig. 1. Raw RSSI values are first plotted for a time window and then we compute the mean and two thresholds, $q+$ and $q-$, which depend on the standard deviation of the values. These are marked in the figure using solid and dotted lines, respectively. Using a window size of 5 in this example, RSSI values lying above the $q+$ threshold are encoded as 1 whereas the ones below $q-$ are encoded as 0.

3) *Reconciliation:* Some bits may differ at both endpoints due to channel degradation or thermal effects. To harmonize the strings generated by both parties, an interactive information reconciliation protocol is used, which identifies and then discards or corrects the bits which differ at both ends.
4) *Privacy Amplification:* To further improve the secrecy of the bitstring, both parties decide to discard certain bits or undertake a transformation operation that enhances the key entropy. This process further helps to mask any information about the string that might have leaked to adversaries during the reconciliation process.

The previous work investigating this technique for body-worn devices has proposed using filtering techniques (e.g., the Savitzky–Golay filter) to minimize noise and asymmetric components in the channel measurements at both ends and improve the correlation of RSS values [52]. This allows both parties to dispense with the information reconciliation process, avoiding significant implementation and processing costs.

The majority of prior work uses this technique to generate shared secret keys between two devices [47], [53]. However, Ali *et al.* [54] have demonstrated that this shared bitstring can be effectively used to determine data provenance. The matching bitstring, or "link signature," at both ends uniquely associates a data session with a wireless link in a way that is verifiable by third parties at a later stage. Forensic investigators can use this signature to determine with confidence that specific data were indeed communicated over a specific wireless link between two parties at a particular point in time.

We leverage this particular application of link signatures in our solution. For wearable devices, this enables experts to verify data offload points (e.g., gateway devices in the home, office, gym, etc.), and thereby approximate the device's location and the subject's activities, and allow for the contextualization of data collected by the device. In case of an incident, this information can be used to verify the data trail and identify erroneous factors.

B. Crowdsourced Logging Primitive

In our previous work [17], we have explored the idea of crowdsourcing the security of data logs by the use of *witnesses*, i.e., smart devices in the vicinity of a wearable device which can later attest to its communication with the gateway. Due to the broadcast nature of the wireless medium, these witnesses overhear communications between the wearable device and the gateway and record fingerprints of this conversation in a lightweight and space-efficient manner using Bloom filters (shown in Fig. 2).

A *Bloom filter* is a probabilistic data structure that allows for compact storage of data items and efficient membership enquiries [55]. The filter is a bit array of predefined size with all bits initialized to 0. A data item to be stored in the filter is first passed through a set of hash functions whose outputs are uniformly distributed over the length of the filter. The outputs of the hash functions are used to set the corresponding bits of the Bloom filter to 1. To resolve a membership enquiry, the same hash functions are run on the item to locate the corresponding bits of the Bloom filter to check their value. If all the corresponding bits are 1, it indicates that the item was likely stored in the Bloom filter. However, if even a single 0 bit features in these locations, it results in a "no membership" outcome. A Bloom filter yields false positives with a given probability, depending on the size of the filter, the number of entries, and the number of hash functions used. However, a Bloom filter does not produce false negatives.

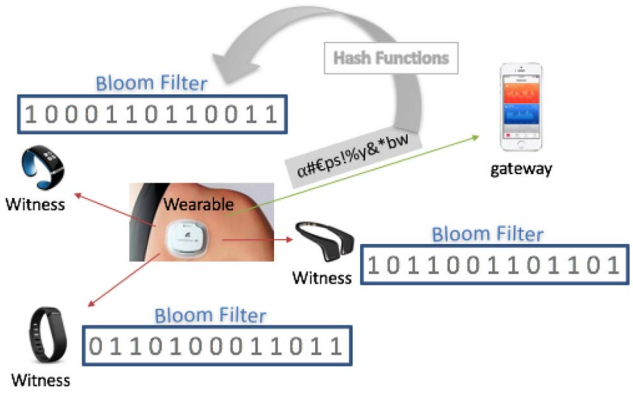

Fig. 2. Crowdsourcing of secure logging.

In our previous crowdsourced logging solution, these witness devices populate their Bloom filters with conversations they overhear and, at preset intervals, digitally sign and submit the filters to a central medical server where they are stored in a timestamped log. This log is replicated in multiple locations to prevent retroactive tampering with the data. A forensics investigator can later verify with high confidence individual data entries in the log by querying the Bloom filters of various witnesses to see if they overheard those communications at that point in time.

IV. Our Solution

The network consists of the following entities.

1) *Wearable devices*, such as fitness trackers and health monitors which record users' data.
2) *A healthcare server* in the cloud which stores data recorded by the wearable device and makes it accessible to concerned parties, such as medical personnel, etc.
3) *Smart devices* in the vicinity, such as smartphones, smoke alarms, other wearables, etc., which serve as witnesses.
4) *A gateway device* which is typically a WiFi access point (AP) that provides local area networking and Internet access for these various devices in a star topology, and maintains detailed logs of all data communications and network events.
5) *An anonymizing service* in the cloud which authenticates statements recorded by witness devices, anonymizes them, and makes them available to forensics investigators.

The solution comprises two protocols: 1) a *pseudonymous testimonial protocol* and 2) a *verification protocol*. We describe these next.

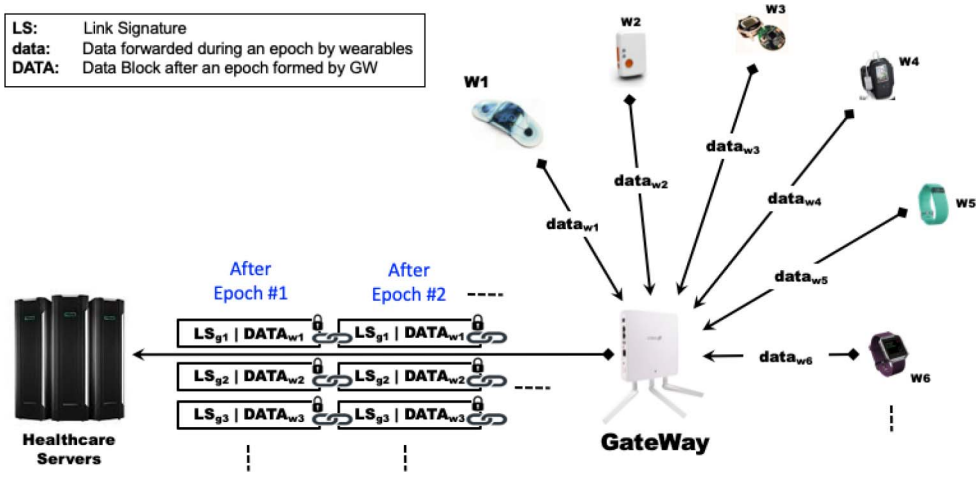

Fig. 3. Creation of gateway log containing data and link signatures.

A. Pseudonymous Testimonial Protocol

This protocol runs on the gateway device and smart devices in the vicinity to enable corroboration of the communications of wearable devices. The process can be summarized as follows: the gateway maintains a detailed and timestamped log of all data communications over the network. Simultaneously, to enable an independent check on this log, smart devices record fingerprints of packets they overhear belonging to the target wearable device. These data logs and fingerprints are signed and stored separately in two different locations. In the event of an incident, a forensics investigator can retrieve these records and compare them to verify the data communications of the wearables device. The process flow closely follows that of our crowdsourced logging primitive described earlier (detailed in [17]) and is modified to include link signatures which enable pseudonymity for witnesses as well as generate the proof of their presence in the vicinity. We describe the process next.

1) The gateway maintains a packet-level record of communications with all devices, including the wearable device. This record is referred to as the *gateway log*. At periodic intervals or *epochs*, this recorded data are bundled together into a batch or *block* and forwarded to the healthcare server.
2) To prevent retroactive tampering, our solution adapts techniques from the literature on the secure timestamping and blockchain technology. Each block is digitally signed by the gateway device, thereby ensuring source authentication. The individual blocks of the gateway log are also chained together (as depicted in Fig. 3), such that each new block contains the *header* of the previous block, i.e., a hash value computed over the previous block, thereby preserving the integrity of the entire log. Furthermore, the log may be replicated in various locations controlled by other stakeholders, e.g., insurance service, cloud backup server, etc., thereby making it harder for an adversary to tamper with the log undetected.
3) The previous block header computed by the gateway device also serves as a unique identifier for the current epoch (an *epoch identifier*) which enables all network devices to synchronize without reliance on a dedicated timestamping service. At the start of every epoch, the gateway device broadcasts this header value to all devices in the network.
4) By the virtue of sharing the wireless broadcast domain, smart devices in the vicinity overhear communications between the wearable and the gateway device and, as witnesses, maintain an independent record against which the gateway log may be corroborated. However, to reduce the memory and communication overheads, instead of logging entire packets, these devices only record a fingerprint of each packet by inserting it into a Bloom filter. We refer to this as the *witness statement* (shown in Fig. 4). The parameters of the Bloom filter are chosen by the witness device *a priori* depending upon multiple factors which we discuss in Section V.
5) In parallel with this process, the gateway and all connected devices generate link signatures for their shared wireless link in the course of their routine communications (as discussed in Section III-A). This results in unique, closely matching bitstrings at both endpoints (i.e., at the gateway and the individual devices) which are very difficult to forge.
6) At the conclusion of each epoch, the gateway compiles the current block and sends it to the healthcare server. The block also includes link signatures computed by the gateway for other devices it communicated within the epoch (shown in Fig. 3). These signatures serve two purposes: first, forensics investigators can use these to later ascertain that the witness devices were indeed physically present in the vicinity during this period; and second, they uniquely identify the wireless link and they vary over time, thereby serving as effective pseudonyms that mask the real identity of the witness.
7) Smart devices in the vicinity independently prepare their *witness testimony* consisting of the identity of the gateway, an epoch identifier, their link signature with the gateway and their witness statement (packet format is shown in Fig. 5). Each witness digitally signs its witness record and dispatches it over an encrypted link to

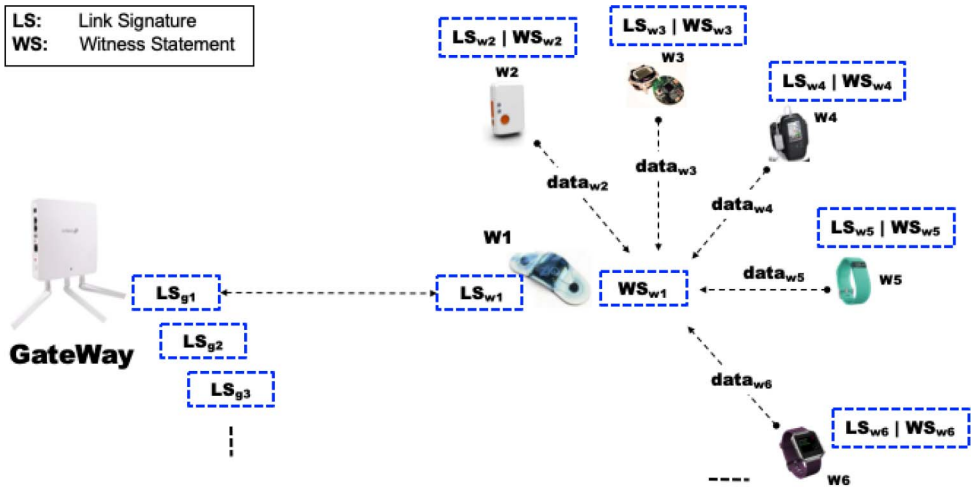

Fig. 4. Creation of witness testimonies (witness statements and link signatures).

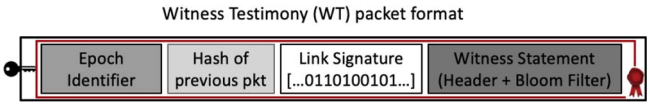

Fig. 5. Packet format of witness testimony.

the anonymizing service, as shown in Fig. 6. The signatures ensure nonrepudiation while the encryption keeps the identity of the witness confidential from the gateway device and the forensics investigators.

8) The anonymizing service decrypts the records, verifies the validity of the digital signatures, and then strips off the signatures, prior to making the records available to forensics investigators. This sanitization step effectively conceals the identity of the witness device.

B. Forensics Verification Protocol

In the event of an incident, a forensic investigator may later use these witness statements to independently verify the data recorded by the wearable device and/or the integrity of the gateway log in a privacy-preserving manner. The architecture is shown in Fig. 7 and the process is detailed as follows.

1) The forensic investigator identifies the epoch for the particular data items he wishes to verify. This is done by examining the gateway log maintained on the healthcare server and extracting the relevant epoch identifier.
2) From the gateway log, the investigator also obtains the link signatures computed by the gateway device for witness devices in the vicinity. These serve as identifiers and proofs of location for the witnesses.
3) The investigator then approaches the anonymizing service and requests all witness statements recorded for the specific gateway device and particular epoch identifier. He compares the link signatures for these records against those recorded by the gateway device to ascertain that the witnesses were physically present at the time of the incident.
4) The investigator verifies individual data items recorded by the wearable or gateway device by querying the witness Bloom filters.
5) Using the witness statements, the investigator may also compute a numerical measure of confidence or "trust" in the data items as we describe in (7) in Section V. He can also verify the chronological ordering of the data with the help of various epoch lengths adopted by different witnesses, as detailed in Section V.

C. Security Properties and Exposition

To discuss the security properties of our solution, we first revisit the scenarios presented in Section II, and explore how our solution assists forensics investigators. Next, we discuss additional useful properties and possible enhancements.

We earlier considered the hypothetical case (*scenario #2*) where Bob may have murdered his wife for insurance benefits and later tampered with the activity records of his wearable device to deceive the authorities. In *scenario #2*, Alice and her insurer produced two conflicting activity records while contesting a personal injury compensation claim. If our crowdsourced logging solution were to be deployed, in both instances, the witness statements would serve as extra reference points from which investigators could derive confidence in the evidence collected from the wearable.

Apart from corroborating wearable communications, this scheme also provides a form of *location proof* [56]. For instance, if Bob were to claim that he was not at home at the time of the crime, but was instead working late at his office, the investigator could verify his account by confirming that his wearable's link signatures match with those that have been forwarded by the office gateway device. Furthermore, the data readings are verified by other witness devices in the office, which lends further credence to Bob's testimony.

The previous work has demonstrated how wearable devices may be hacked to backfill healthcare data [11], but to subvert our scheme and cheat the investigators, an attacker would also have to compromise the gateway and all the witness devices and falsify their statements. Furthermore, this must be done

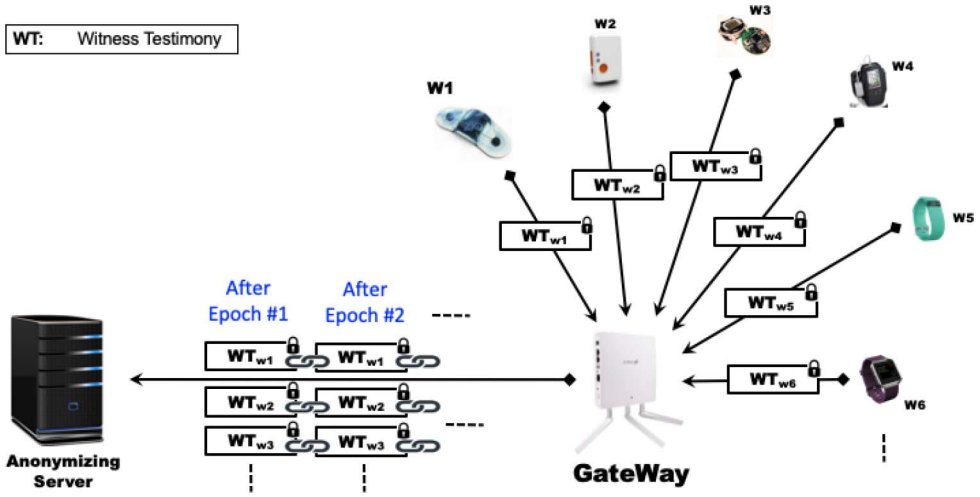

Fig. 6. Witness testimonies forwarded to anonymizing service after each epoch.

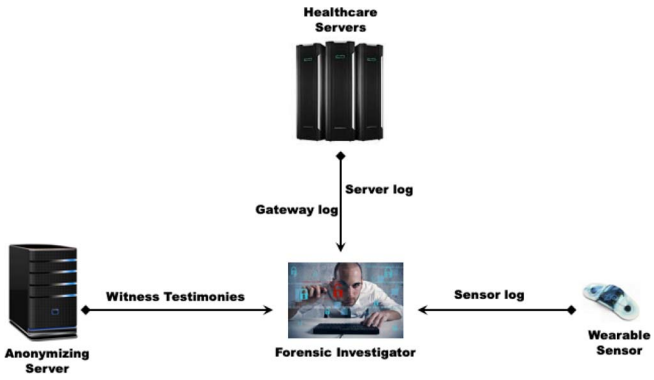

Fig. 7. Forensic verification.

within the narrow time window of the epoch. When an epoch concludes, the witness statements are signed by the respective witnesses and dispatched to the anonymizing service. Digital signatures on epoch-level data blocks and witness records ensure *nonrepudiation* and *integrity* of forensic data, while hash chaining these items preserves *chronological ordering*.

Distributing trust in this manner to neighboring devices and third parties thereby enables effective *accountability* of major stakeholders. The scheme is *tamper evident* in that if any party later alters the data, the alteration can be detected with very high probability. Furthermore, in such cases, it may assist forensic investigators in determining liabilities. For example, in the earlier scenario where Alice accused her insurance company and healthcare service of tampering with her fitness records, investigators can use the witness statements to determine the correct account.

Furthermore, our scheme significantly improves on the shortcomings of the individual primitives we deploy (described in Section III). The link signature primitive secures data provenance across a single hop but notably cannot protect against the case where both communicating parties collude to tamper with the provenance record. On the other hand, the crowd-sourced logging primitive does not preserve the privacy of the witnesses, nor does it provide any guarantee that the witnesses are indeed genuine.

In our case, however, collusion may be detected by using the witness record. Likewise the hash chained log resists retroactive data tampering. The link signature imparts investigators with confidence that witnesses were indeed physically present on the scene during the epoch. Furthermore, the signature acts as a credible pseudonym for witness devices that effectively dissociates witness statements across epochs, thereby providing the important property of *unlinkability*.

Our scheme relies on the presence of wireless witness devices in the vicinity of a wireless transaction. Our scheme would not work if there is no such device around to log the transaction, however, a single witness would suffice to verify data. While there is already a high penetration of such devices in urban society with an even higher growth rate [57], the lack of smart devices in rural areas is quickly diminishing as well. Governments are pushing to deploy telemedicine or mHealth solutions for rural healthcare [58] due to their potential socio-economic benefits [59]. Smart farming [60] is another promising application of the modern technology that will give rise to the density of smart devices in rural areas. For instance, Australian researchers, with government help, carried out a number of smart farming projects and reported considerable benefits in productivity, quality, and management [60].

Here, we briefly discuss the practicability of our solution. Our scheme is lightweight as it exploits the passive radio scanning activity of wireless devices. It does not require frequent use of expensive cryptographic techniques (we provide energy estimates in Section V). There is no reliance on specialized hardware, such as the GPS unit and tracking technology that one might typically deploy in a solution that provides location proofs.

The regarding network architecture, most of these entities we describe already exist in typical home deployments of wearables and smart devices. Our solution expands the role of the other entities to log messages they overhear. This may be easily accomplished using software patches on the gateway and smart devices. Our solution adds only one new entity to

the network, i.e., the anonymizing service, whose role is to safeguard the privacy of witnesses.

The anonymizing service we describe is inspired by a similar trend toward outsourcing security, evidenced most recently in proposals, such as blockchain notaries (SilentNotary [61], Stampd [62], etc.), key directories (such as KeyBase [63]), and cryptocurrency tumblers [64]. Moreover, we anticipate the feasibility of such a service in the wake of recent EU General Data Protection Regulation (GDPR) which prescribes strict guidelines on the collection and handling of user data. This service could be operated by third parties in compliance with regulations by relevant industry or watchdog bodies.

An obvious question that arises at this point is with regard to the trustworthiness of this anonymizing service. The service may tamper with the witness statements on its own or in collusion with other parties to support a false narrative. While there is no guarantee that this service is honest, there are cryptographic mechanisms that can be deployed to reduce the probability of fraud. For instance, threshold cryptography may be used to distribute the decryption of witness statements among multiple parties on the assumption that at least one of these parties will be honest. The only change necessitated, in this case, would be to switch to a suitable cryptosystem, such as the Paillier scheme [65] or elliptic curve digital signature algorithm (ECDSA).

Another interesting strategy is *randomized partial checking* which splits processing into two or more stages undertaken using mixnets, which may then be publicly audited for fairness [66]. This strategy has been deployed for auditing purposes in various well-known electronic voting protocols, including Scantegrity [67], Prêt à Voter [68], and Wombat [69]. In this situation, the decryption process consists of two or more mixes connected in series, such that the outputs of one mix serve as the inputs of the next mix. The ciphertext, i.e., the witness statements, would have multiple layers of public-key encryption, each corresponding to the credentials of a particular mix. As the ciphertext passes through the mixnet, each mix strips away a layer of encryption and permutes the inputs before passing them on to the next mix. The process can be audited with very high confidence by an observer who randomly selects certain outputs of the mixes which are then revealed by the mix operator and verified. It is important that the maximum number of outputs be checked in this way while ensuring that no end-to-end path through the mixnet is revealed.

When compared to the crowdsourced logging scheme described in [17], our current solution preserves witnesses' privacy at the cost of additional processing and introduction of an anonymizing service. The process of link signature generation between the gateway and the witness devices hardly adds an extra cost as the RSS values are mostly sampled when routine data are transmitted. (We adapt the ultralightweight link signature generation strategy described in [52].) Furthermore, encrypting a witness record with the anonymizing service's public key costs 52 mJ of energy (for MicaZ motes, as discussed later in Section V-C3), which is an infrequent operation that occurs only once each epoch. Moreover, the crowdsourced logging scheme does not preserve the privacy of the witnesses. Any third party can examine the witness logs and infer certain information about the witnesses, including their identities, their connectivity, location, movements, etc. We have sought to address this shortcoming in this article. The anonymizing service conceals witness identities and patterns while the link signatures provide hard guarantees regarding the authenticity and integrity of the witness logs. We believe people are more likely to buy a smart device that lets them support a forensic investigation anonymously than one that may implicate them in an active investigation. This hunch is supported by real-life examples that we mention in Section II-A where two teenage boys were murdered in a house party attended by approximately a hundred people [28], and the case where a-seven-year old girl was murdered during a gang fight in front of a crowd [29]. In both cases, no witness was willing to testify.

Regarding the motivation and deployment for our solution, we believe an ecosystem such as we have described in this article is already emerging. We have noted in Section I that large companies are incentivizing wearables for their employees and that some insurance companies are also encouraging this technology. Indeed, John Hancock, one of the oldest and largest insurers in the U.S. has now stopped offering traditional life insurance and has switched entirely to interactive policies that mandate wearable devices [70]. Likewise these devices are starting to feature prominently in criminal investigations. Given how vulnerable these devices are, with increasing integration into our lives, it is only a matter of time before pressure builds for innovative solutions to secure these devices. It is entirely possible that employers and insurers may bundle such a solution as part of their policies in the near future.

V. Trust Model and System Design Parameters

In this section, we develop an analytical model to compute trust in the presence of witnesses as well as their statements. We further discuss various design parameters, namely, the length of link signatures, their bit agreement, size of Bloom filter for witness statements, energy cost to generate witness statements, granularity of time with which an occurrence (reading) can be verified, and how these parameters are affected by epoch length. We also discuss the parameters' various tradeoffs helpful in selecting optimum values of these parameters in devices of varying capabilities and resources.

A. Computation of Trust in Witness Presence

Here, we develop an analytical model to quantify the confidence in the presence of a witness in the environment where the incident happened. As discussed earlier in Section IV that the witnesses generate link signatures with the gateway to prove their presence in the vicinity. It is well documented in [52], [53], and [71] also shown in Section VI that these link signatures cannot be forged by an eavesdropper as they are generated based on unique spatiotemporal variations in the wireless channel between two parties (discussed in Section III). The confidence in such a proof of presence can only be undermined by the possibility that an attacker might successfully replicate the signatures by chance by choosing a

random bitstring with a similar or better match than that of the genuine witness, the probability of which we call "replication probability" g that can be expressed as follows:

$$g = \frac{1}{2^l} \sum_{i=s}^{l} \binom{l}{i}; \qquad 0 \leq s \leq l. \tag{1}$$

Here, l is the length of the link signature and the term 2^l represents the number of all possible selections of a random bitstring while the term $\sum_{i=s}^{l} \binom{l}{i}$ is the number of random selection of strings that have the same or better match than that of the genuine witness. The parameter s here represents the number of matching bits between the link signatures of the gateway and the genuine witness. We note that, with a randomly chosen bitstring, a 50% bit match is more likely (which is the lowest match considered) and that the probability of match decreases symmetrically around 50% bit match, e.g., the probabilities of a 30% and 70% bit match are same. To shift the reference, we double the probability while we only consider the match of more than 50% bits. Equation (1) can be rewritten as follows:

$$g = \frac{1}{2^{l-1}} \sum_{i=s}^{l} \binom{l}{i}; \quad l/2 \leq s \leq l. \tag{2}$$

Also, since $\binom{l}{i} = \binom{l}{l-i}$, with the change of limits, (2) becomes

$$g = \frac{1}{2^{l-1}} \sum_{i=0}^{l-s} \binom{l}{i}; \quad l/2 \leq s \leq l. \tag{3}$$

Probability that an attacker cannot replicate the link signatures successfully can be translated as the trust in the witness presence and can be represented as follows:

$$\tau_p = 1 - g = 1 - \frac{1}{2^{l-1}} \sum_{i=0}^{l-s} \binom{l}{i}; \quad l/2 \leq s \leq l. \tag{4}$$

Based on the above trust function, Fig. 8 shows an example of how trust changes with the varying number of bits in agreement for a given length of link signature (101 b in this example). We note that the value of the trust approaches 0.99999 (five 9s, the desired value) at the bit agreement of approximately 73%.

In order to further explore the behavior of the trust function [given in (4)] and to validate it, we calculate the trust at the boundary values of the parameter s (the number of matching bits). For $s = l$, i.e., when all the bits are matching, the value of trust reduces to

$$\tau_p = 1 - \frac{1}{2^{l-1}}; \quad \text{when} \quad s = l. \tag{5}$$

The above relation shows that the value of trust increases with the length of the link signatures and approaches 0.99999 (five 9s, the desired value) at the link signature length of 18.

However, at $s = l/2$, i.e., when half the bits match, the value of trust becomes

$$\tau_p = 1 - \frac{1}{2^{l-1}} \sum_{i=0}^{l/2} \binom{l}{i}; \quad \text{when} \quad s = l/2.$$

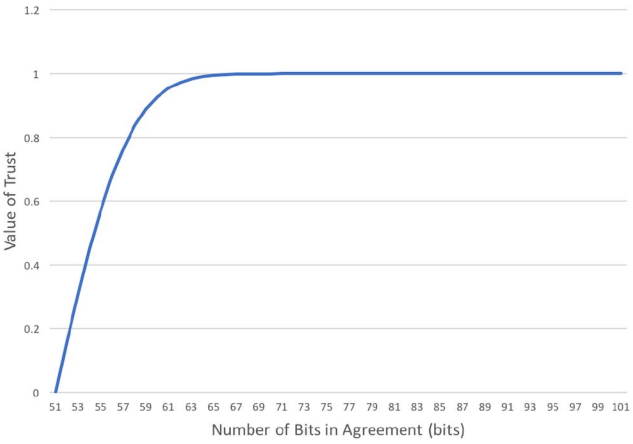

Fig. 8. Value of trust in witness presence having 101-b link signature with varying number of bits in agreement.

Since the sequence of binomial coefficients of $\sum_{i=0}^{l} \binom{l}{i}$ is symmetric and we know that $\sum_{i=0}^{l} \binom{l}{i} = 2^l$, hence $\sum_{i=0}^{l/2} \binom{l}{i} \approx (2^l/2) = 2^{l-1}$, that leads to

$$\tau_p \approx 0; \quad \text{when} \quad s = l/2$$

which conforms to the intuitive expectation as 50% is the lowest match.

B. Computation of Trust in Witness Statement

Data verification through Bloom filters comes with a probability of false positive, which affects our trust in the witness statements. This probability of false verification f in Bloom filters, well documented in [72], can be computed from the following relation:

$$f \approx e^{-\frac{m(\ln 2)^2}{n}} \tag{6}$$

where n is the number of packets inserted in the Bloom filter, while m is the size of the Bloom filter in bits. The witnesses include these parameters in their testimony headers to be used by investigators to compute probability of false positive. Trust in the witness statement can be represented as follows:

$$\tau_d = 1 - f \approx e^{-\frac{m(\ln 2)^2}{n}}. \tag{7}$$

C. System Design Parameters Tuning

In Section IV, we have described that our scheme uses a system-level *epoch*, which is a set time interval. After each such interval, the gateway forms a data block and hash chains it to the previous block to create the gateway log. Similarly, the link signatures between the gateway and the witnesses, and the witness statements by the witnesses are also generated every epoch and forwarded to the servers. In this section, we discuss how different system parameters depend upon the epoch size and how it might impact the witnesses.

1) Link Signature Parameters: Link signature parameters—length and bit agreement—are the sole contributors in establishing the trust in the witness presence, as evident from (4) and plot in Fig. 8. For a given bit agreement, the trust in the witness presence τ_p increases with the increase of link

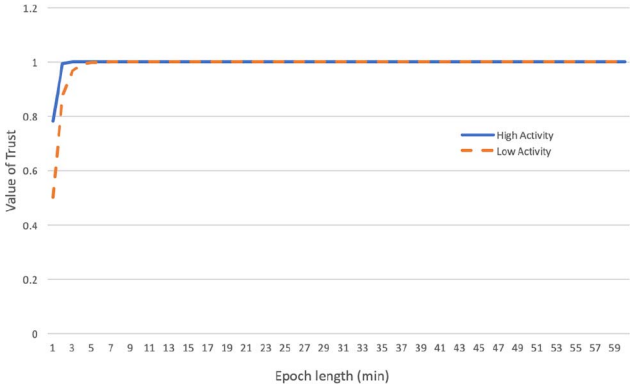

Fig. 9. Value of trust in witness presence in a high and low activity environments against epoch length.

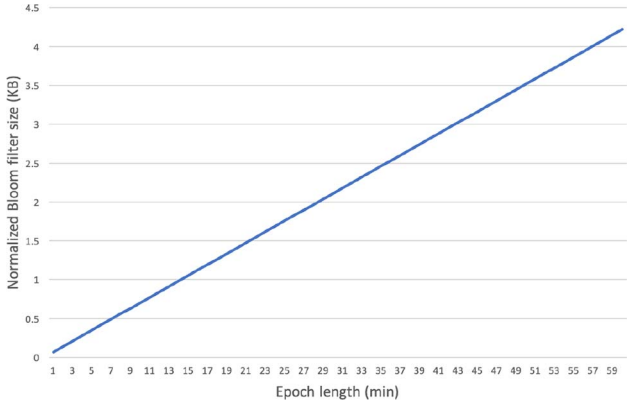

Fig. 10. Bloom filter size versus epoch length at $r = 1$ packet/s and $f = 1\%$.

signature length. That said, a higher trust in witness presence requires a longer link signature, however, for a longer link signature, sufficient time (a longer epoch) is needed. In a representative work on wearable devices by Ali *et al.* [52], it is shown that the length of the link signature and the bit agreement depends upon the dynamics of the environment. Ali *et al.* [52] performed experiments to generate link signatures between a gateway and a wearable device with low and high physical activity (mobility) of the subject wearing the device. They listed the values of performance metrics, such as bit agreement and secret bit rate of link signatures for those scenarios. They also demonstrated the effect of filtering and the variation of activity threshold θ on the performance metrics.

Let us consider a low physical activity environment such as a hospital ward. From [52], we know the performance metrics of link signature generation for such an environment. In a low activity environment (for activity threshold $\theta = 2$), there is a 100% bit agreement with a secret bit rate of 0.036 b/s. In Fig. 9, the broken line shows the trust in the witness presence τ_p in a low activity environment against the varying epoch length. We note that a minimum nine minutes of epoch length is required for trust to reach five 9s and beyond in the low activity environment.

Similarly, for a high activity environment such as a gym, there is 99.88% bit agreement with an increased secret bit rate of 0.101 b/s (for activity threshold $\theta = 2$) [52], as a slight movement causes variation in the channel between two parties that results in higher bit rate. In Fig. 9, the solid line shows the trust in the witness presence τ_p in high activity environment against varying epoch length, which shows that for high activity environment, only 3 min of epoch length is required for trust value to reach five 9s.

2) Bloom Filter Size: The size of the Bloom filter is set considering two main factors: 1) the anticipated number of inserted items and 2) the targeted probability of false positive. The size of the Bloom filter, as per standard calculation [72], is given in the following expression:

$$m \approx -\frac{n \ln(f)}{(\ln 2)^2}. \tag{8}$$

Here, m is the size of the Bloom filter in bits, while n is the number of items inserted and f is the probability of false positive.

The witnesses set the Bloom filter and epoch length *a priori* at the start of the epoch, so they estimate the number of items to be inserted from the average data transmission they overhear. Let t be the epoch length and r be the rate at which a witness is overhearing packets, such that $n = rt$. Equation (8) can be rewritten as follows:

$$m \approx -\frac{rt \ln(f)}{(\ln 2)^2}. \tag{9}$$

Normalizing the filter size by the packet reception rate r (packet/s), the above equation can be expressed as

$$\mu = \frac{m}{r} \approx -\frac{\ln(f)}{(\ln 2)^2} t \tag{10}$$

which is a linear expression for a targeted probability of false positive f.

Fig. 10 plots the normalized Bloom filter size μ (in kB) for a targeted probability of false positive of 1%. A given reception rate r is multiplied by the normalized size to get the required Bloom filter size. A larger Bloom filter size may be undesirable for certain wearable devices due to memory limitations—a concern quickly being diminished due to advancement in technology offering cheap and compact storage.

3) Energy Cost: Energy cost is the most crucial parameter when it comes to wearable devices due to their size limitations, which is the primary reason why a complete cryptographic security protocols suite cannot be implemented on these devices, which in turn makes them vulnerable to security threats. Here, we only provide the energy estimates for the witness devices, as the gateway devices are usually rich in resources and energy consumption for such a lightweight protocol is not a major concern for them. As detailed in Section IV, our solution is lightweight as most of its operations either use the ongoing routine procedures run by wearable devices or implement light cryptographic operations only scarcely. For link signature generation, the RSS values are mostly sampled when routine data are transmitted by the wearable device to the gateway and its acknowledgment is received by the device.

The rest of the protocol has the following energy costs for the witnesses: 1) logging overheard packets in the witness statement (Bloom filter); 2) digitally signing the witness testimony [link signatures + witness statement]; 3) encrypting the witness testimony with the public key of anonymizing server; and 4) transmitting the witness testimony to anonymizing server via gateway. For witness statements, the ongoing radio scanning by these wireless devices allows them to witness packets generated by the neighboring devices. To insert these packets into Bloom filter, they are hashed with an appropriate hash function. For MicaZ motes—that we use in our experiment in Section VI—the cost of SHA-1 hash for one packet is 154 μJ [73], however, the hardware implementation of the hash function reduces the cost dramatically. For example, Kaps and Sunar [74] implemented SHA-1 hash for RFID tags and wireless sensors that only consumed 21.65-nJ energy (26.73 μW at 500 kHz in 405 cycles). To digitally sign the witness testimony, the ECDSA with a public-key size of 160 b in MicaZ mote costs 52 mJ of energy [75]. Public-key encryption of witness testimony consumes the same energy as a digital signature, which is 52 mJ [75]. As far as the transmission of witness record is concerned, transmitting 1 b costs 0.6 μJ of energy in MicaZ mote [75].

Let us revisit these costs to see which one of these affects the total cost if the length of the epoch is varied. In other words which cost is linear in time and does not depend upon the epoch length and *vice versa*. We note that the logging cost is linear as it only depends upon the rate at which it receives the packets. Moreover, since the size of the Bloom filter is proportional to the epoch length, as given in (9), the cost of transmitting a witness statement is also linear in time and does not depend upon the epoch length. The only significant costs that are affected by the epoch length are the ones for digital signature and encryption of witness testimonies, operations that happen once every epoch.

Let us take a scenario where a witness device is receiving packets at a rate of 1 packet/s. It uses three hash functions to log these packets into a Bloom filter. The size of the Bloom filter depends upon the epoch length it chooses. Let us calculate the total energy costs over a period of 1 h from logging the witness data to its transmission by varying the epoch length from 1 min to 1 h. Fig. 11 shows the total energy cost of implementing the witness testimonial protocol on a MicaZ mote. The solid line graph shows the total cost using the existing SHA-1 implementation while the dotted line graph uses the hardware implementation cost of SHA-1 given in [74]. We note that the logging and transmission costs are constant over various epoch lengths being 564.75 and 10.43 mJ, respectively, for two implementations while the digital signature and encryption costs reduce by selecting longer epoch lengths.

4) Time Granularity: The time interval in which a sensor reading is verified is determined by the epoch length. A witness creates witness testimony (that contains a Bloom filter) every epoch, hash chains it with the previous epoch, and forwards to the servers. While the testimonies are timestamped and an epoch-level chronological ordering is preserved due to hash chaining, the ordering of individual packets within an epoch is lost when inserted into the Bloom filter. A forensic investigator can verify the individual packets in a Bloom filter and can verify an event happing during that epoch, however, the time granularity of the event (sensor reading) depends upon the epoch length. The shorter the epoch length is the more fine-grained timing information is.

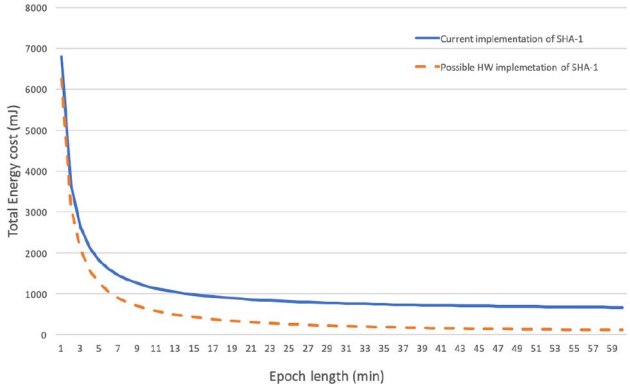

Fig. 11. Total energy cost over the period of 1 h in MicaZ mote.

5) Epoch Length: The above discussion reveals that all the system parameters, namely, link signature length, the bit agreement and hence the trust in witness presence, Bloom filter size, energy consumption, and time granularity depends upon the selection of the epoch length. If we look at the graphs in Figs. 9–11, we note that a lower epoch length is limited by two factors: 1) lower trust in witness presence and 2) higher energy cost. We note that for a low and high activity environments, a witness device needs to have a minimum of 9 and 3 min of epoch lengths, respectively, to prove their presence in the vicinity. However, a resource-constrained device might not be able to sign and transmit witness statements that frequently. Moreover, larger epoch lengths are limited by the memory size used by the Bloom filter. We also note that many IoT devices in the vicinity, such as smoke sensors and smartphones can easily meet the energy demand for using the shortest allowable epochs that could provide more time granularity to the provenance data.

Based on the above discussion, we propose to keep a system-level epoch of fixed duration (e.g., 1 h), however, different devices can divide the system epoch into *subepochs* and can decide *a priori* the appropriate subepoch length based on their energy and memory resources, mobility profile, and data rate in the environment providing different levels of time granularity that could help the forensic investigator reconstruct and verify the events with greater precision.

VI. Experiments and Simulations

A. Prototype Experiment

We implemented our scheme using IEEE 802.15.4 compliant 2.4-GHz MicaZ wireless motes. Our experiment involved a human subject wearing a MicaZ mote on his right arm emulating a fitness monitor transmitting at the rate of 1 packet/s. The subject walked within an office environment with cubicles (shown in Fig. 12) for approximately 15 min. A stationary gateway in the middle of the office hall not only received and logged the wearable sensor data but also generated the

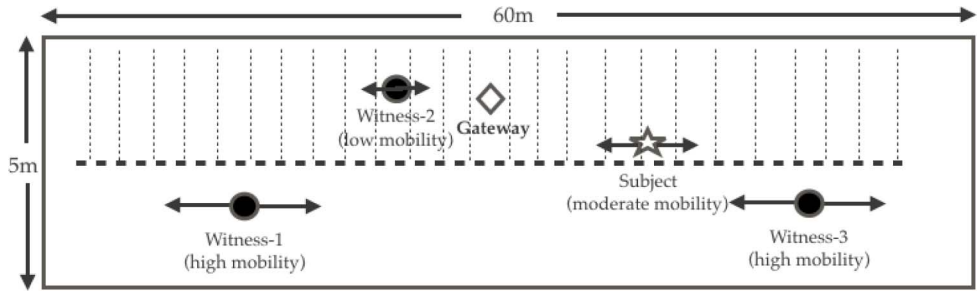

Fig. 12. Layout of the prototype experimental setup.

TABLE I
PERFORMANCE PARAMETERS AND TRUST IN THE WITNESS TESTIMONIES IN THE PROTOTYPE EXPERIMENT

	Sensor (medium mobility)	Witness #1 (high mobility)	Witness #2 (low mobility)	Witness #3 (high mobility)
Match with gateway's link signature	99.34%	100%	97.01%	100%
Link signature length (bits)	151	255	67	262
Secret bit rate	0.1678	0.2833	0.0744	0.2911
$\overset{\tau_p}{(1-g)}$	1	1	1	1
$\overset{\tau_d}{(1-f)}$		0.985	0.984	0.986

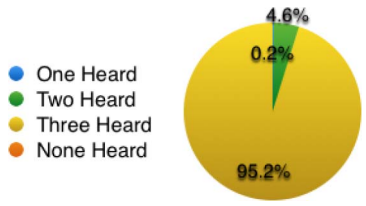

Fig. 13. Percentage of packets verified by witnesses.

TABLE II
BIT AGREEMENT AMONG THE LINK SIGNATURES OF WIRELESS WEARABLE DEVICES

	Sensor LS	Witness#1 LS	Witness#2 LS	Witness#3 LS
Sensor LS	100%	51.51%	51.85%	52.76%
Witness#1 LS	51.51%	100%	51.85%	52.69%
Witness#2 LS	51.85%	51.85%	100%	51.03%
Witness#3 LS	52.76%	52.69%	51.03%	100%

link signature with the wearable device. Three such wearable devices in the vicinity witnessed the fitness tracker's communication with the gateway and generated the witness statements containing the fingerprint of the communication. They also generated the link signatures for that epoch with the gateway to prove their presence in the vicinity.

The subject wearing the sensor device usually sats in his cubicle and occasionally walked a bit in the office that we classify as having medium mobility. One of the witnesses (witness #2) mostly sat in his chair and moved three times only slightly during the entire experiment and is classified as having low mobility. However, the rest of the two witnesses had high mobility and walked in the office for the entirety of the experiment.

The sensor wearable device and the three witness devices generated with the gateway, the link signatures of varying length, and bit agreement depending upon their mobility as shown in Table I. The channel between the gateway and the wearable devices was sampled at both ends for RSS values. At the gateway end, the sampling was done when the packet was received while at the wearable device end, it was done when it received acknowledgments of the packets from the gateway. The witnesses generated their witness statements as well based on what they overheard from the sensor communication. They used a 1-kB Bloom filter each with five Murmur3 hash functions. We calculated the trust in the presence of each witness τ_p, as well as in their witness statements τ_d, as given in Table I.

We verified the sensor data (packets logged by the gateway) by checking the witness statements of all three witnesses. We found that none of the packets remained unheard and all the packets verified by at least one witness, as depicted in Fig. 13.

In our prototype experiment, we considered the malicious behavior of the witnesses as well in which they tried to generate the sensor's link signatures in an attempt to fake more devices with the same link signatures undermining the uniqueness of the link signatures as we claimed earlier. They did so by sampling the RSS values of the acknowledgment packets from the gateway to the sensor that they witnessed. However, we found out that these fake link signatures had a match of around 50% bits (50.08%, 50.32%, and 51.19%, respectively) with the gateway-sensor link signatures as compared to their genuine 99.34% match.

In Table II, we have listed the bit agreement among the link signatures of wireless wearable devices that they generated with the gateway. It is evident from the table that their signatures are completely independent from each other with nearly 50% bit agreement between any two link signatures created during the prototype experiment.

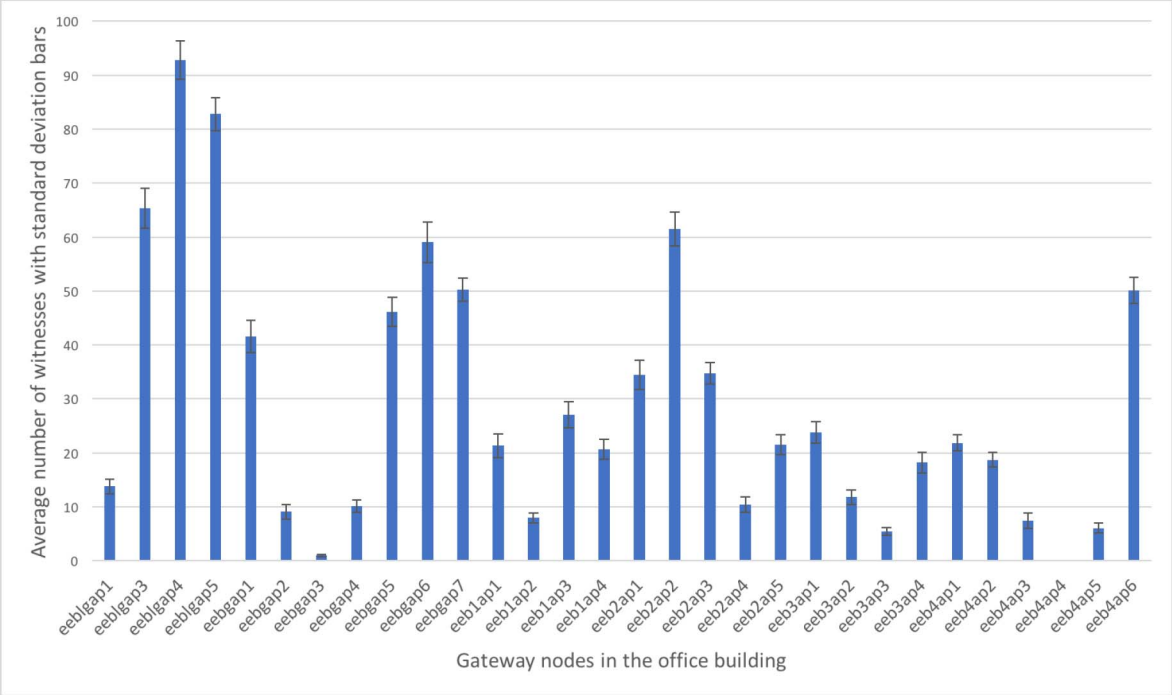

Fig. 14. Average number of witnesses and standard deviation during the epoch 4 P.M.–5 P.M.

B. Experiment for Office Environment

We revisit *scenario #2* presented in Section II where a company employee Alice injures herself while in office, says she falls from one of the office stairs while no eyewitnesses happened to be around. As discussed earlier, Alice claims a huge compensation and uses data from her activity tracker in court to prove her lack of activity after the incident. The forensic investigator needs to confirm whether the incident happened in the office during the claimed time (say between 4 P.M. and 5 P.M.) or not. We simulate the situation in a multistorey university building full of students and staff whom we model as representing the company employees. Their wireless gadgets, such as smartphones and laptops represent smart wearable devices. The building has six storeys, namely, lower ground, ground, and 1–4 and is equipped with 30 Cisco wireless APs mounted at different levels for full coverage of service. With the help of root access to the APs, we managed to get the statistics of all the wireless connections with the APs for a single day.

With the help of this real trace data, we simulate our scheme using Python language. We make the assumption that the clients connected to a certain APs are within the broadcast domain of each other and can act as the witnesses to each other's communications. We divide the day (24 h) into 24 epochs of 1 h each. The wireless clients and the APs (gateways) implement the *anonymous testimonial protocol* discussed in Section IV. The clients and their respective APs generate link signatures for each epoch and forward to a central database at its conclusion. Moreover, the clients manage witness statements (Bloom filters) for each epoch, which they populate with the transmissions they overhear from neighboring devices and forward them to a central database at the conclusion of each epoch. In our simulation, we assume that all the clients (wearable devices) transmit one packet every 10 s (0.1 packet/s). The size of Bloom filters (witness statements) is set 1 kB for each client with three Murmur3 hash functions used to insert overheard packets.

From trace data that we got from the APs, we worked out the average number of witnesses Alice could have if she were connected to a given AP during epoch 4 P.M.–5 P.M., which is shown in plots given in Fig. 14 together with their standard deviations.

It is apparent from the plots that the witnesses are not uniformly distributed across the APs due to the apparent reasons of the clustering of students in certain areas (lecture rooms, labs, etc.). The witnesses on average can go as high as 92 if Alice is connected to the AP "eeblgap4." The APs such as "eeb4ap4" that has no witnesses and "eebgap3" with one witness are located in parts of the building not being used by the students or staff during the epoch. However, given Alice is connected to any random AP, the probability of Alice having more than five witnesses is more than 93%. In our simulation, we randomly chose a client to emulate Alice's fitness tracker, which happened to be connected to the gateway "eeb2ap4" for the epoch 4 P.M.–5 P.M. The probability of having a given number of witnesses for a packet transmitted by Alice's fitness tracker is given by the graph in Fig. 15. The average number of witnesses per packet is 10 with a maximum of 14 witnesses.

The job of the forensic expert is to verify Alice's fitness tracker data (proving a fall from the stairs) from the witness statements of neighboring smart devices. We implement *forensic verification protocol* discussed in Section IV for this purpose. In summary, we gather the packets Alice generated during the epoch 4 P.M.–5 P.M. as well as the link signatures that Alice's gateway eeb2ap4 had generated with all the devices connected to it during the epoch. In order to

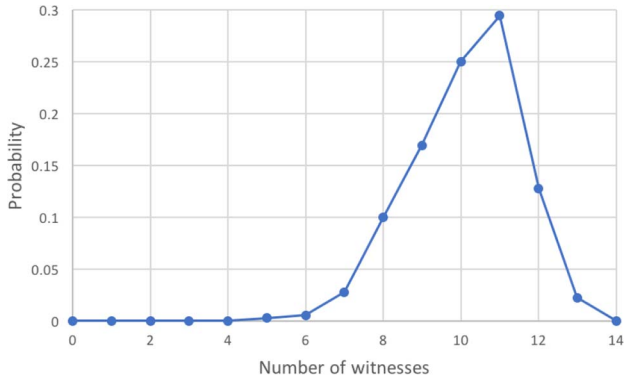

Fig. 15. Probability of the number of witnesses during the epoch 4 P.M.–5 P.M. given Alice is connected to eeb2ap4 gateway.

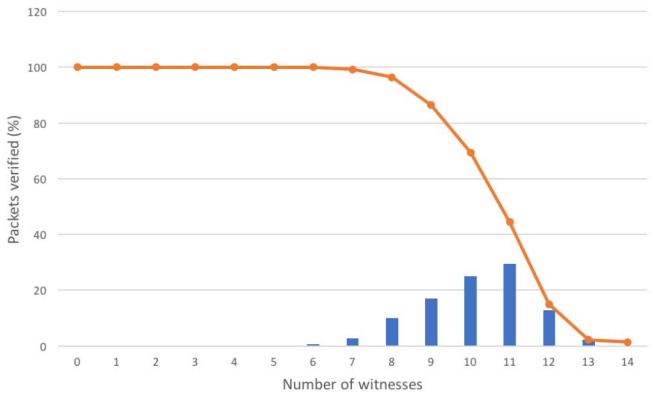

Fig. 16. Packets verified by witnesses during the epoch 4 P.M.–5 P.M. (column chart: PDF and line chart: CCDF).

TABLE III
FINDINGS BY FORENSIC INVESTIGATOR

Witness #	Link Sig. Match (%)	Packets Verified (%)	τ_p (%)	τ_d (%)
1	95.313	65.278	1	97.180
2	94.531	97.222	1	90.914
3	94.531	95.556	1	91.219
4	96.094	90.833	1	92.365
5	96.094	26.667	1	99.961
6	97.656	63.333	1	97.342
7	96.094	82.222	1	94.275
8	97.656	44.167	1	99.356
9	92.969	34.444	1	99.875
10	96.094	90.833	1	92.237
11	94.531	65.833	1	96.649
12	96.875	68.889	1	96.560
13	96.875	97.222	1	90.914
14	94.531	90.278	1	92.043

identify the potential witnesses, against each link signature posted by the gateway during the epoch, we searched for the best match from the list of all the link signatures posted by each individual device in the building for that epoch. We then verified Alice data from the witness statements associated with the matching link signatures of the witnesses in the previous step. Table III lists the findings during the forensic investigation, including the trust in the presence τ_p, trust in the witness statement τ_d, and the percentage of packets verified by each individual witness of total 14 identified by the protocol. The line chart in Fig. 16 (complementary CDF plot) shows the percentage of packets verified by the least number of witnesses while the column chart (PDF plot) shows the packets verified by the exact number of witnesses. It is apparent from the graphs that at least seven witnesses verified all the data and 29.4% packets found the most (11) witnesses.

VII. CONCLUSION

Due to the increased popularity and usage of wearable technology by individuals, data from these devices have started to facilitate healthcare providers, insurance companies, and law enforcement agencies. To ensure the correctness of data and its admissibility as evidence in courts, novel solutions are required to secure the data against tampering by various stakeholders, such as patients/users, doctors, insurers, and prosecutors. In this article, we have proposed a crowdsourced secure logging scheme in which the smart devices in the vicinity of the wearable sensor record its communication and ensure its contextual correctness in a lightweight manner, acting as witnesses to the sensor's transaction. To prove the witness presence in the vicinity and to ensure its privacy, our scheme has used nonforgeable link signatures between the gateway and witness devices based on the unique wireless channel between them. We implemented the prototype solution of our scheme using real wireless devices in real-life environments. Furthermore, we simulated our scheme using real data from wireless devices in a university building emulating an office environment with results that validated the feasibility and efficacy of our scheme.

In future work, we intend to undertake research in the following directions.

First, there is a potential of further optimization for our crowdsourced logging scheme. The cost of digital signatures could be amortized by using an aggregate signature scheme like the one proposed by Ali *et al.* [76] that uses a Merkle hash tree. Witnesses can only sign the root of the statements' tree after its formation and an individual witness statement could be authenticated with the help of the authentication path to the root. Such a solution has the potential to significantly reduce the energy costs of our scheme.

Second, to add further context and enhance the security guarantees, our scheme could be integrated with other secure localization schemes such as the one described by Saroiu and Wolman [56]. Those witnesses equipped with GPS could insert a more precise location proof in addition to the loose form of localization our scheme provides. We will also examine the possibilities of integrating our scheme with biometric authentication methods [77].

Third, we intend to make law enforcement agencies accountable by notifying the mobile users whose data are accessed by the police without consent. This is inspired by the idea of "making decryption accountable" presented by Ryan [78], who proposed a scheme whereby a decrypting agent cannot undertake a decryption operation without leaving

a trace in the log. We intend to explore ways in which an access event log could be maintained for any use of such provenance data to curb the abuse of users' privacy.

We believe the above-mentioned research directions could considerably improve our scheme, increase its effectiveness, and lead to greater adoption and better utilization of wearable technology.

REFERENCES

[1] *IDC Forecasts Shipments of Wearable Devices to Nearly Double by 2021 asSmart Watches and New Product Categories Gain Traction*, IDC, Framingham, MA, USA. Accessed: Dec. 18, 2017. [Online]. Available: https://www.idc.com/getdoc.jsp?containerId=prUS43408517

[2] C. Farr. (Apr. 2016). *How Fitbit Became the Next Big Thing in Corporate Wellness*. Accessed: Dec. 18, 2017. [Online]. Available: https://www.fastcompany.com/3058462/how-fitbit-became-the-next-big-thing-in-corporate-wellness

[3] *John Hancock Introduces a Whole New Approach to Life Insurance in the U.S. That Rewards Customers for Healthy Living*, JH Insurance, Boston, MA, USA, Apr. 2015. Accessed: Jun. 13, 2016. [Online]. Available: http://www.johnhancock.com/about/news_details.php?fn=apr0815-text&yr=2015

[4] *UnitedHealthcare and Qualcomm Collaborate to Launch New Wellness Program That Links Financial Incentives With the Use of Wearable Devices*, UnitedHealthcare, Minnetonka, MN, USA, Mar. 2016. Accessed: Jun. 13, 2016. [Online]. Available: http://www.unitedhealthgroup.com/Newsroom/Articles/Feed/UnitedHealthcare/2016/0301QualcommUnitedHealthcareMotion.aspx?r=1

[5] I Big CloudAnalytics. (Mar. 2016). *National Australia Bank's "MLC On Track" Program Leverages Big Cloud Analytics' Predictive Analytics From Wearable Devices and Internet of Things Data*. Accessed: Jun. 13, 2016. [Online]. Available: https://goo.gl/fvvTAv

[6] K. Crawford. (Nov. 2014). *When Fitbit Is the Expert Witness*. Accessed: Dec. 14, 2017. [Online]. Available: https://www.theatlantic.com/technology/archive/2014/11/when-fitbit-is-the-expert-witness/382936/

[7] A. Watts. (Feb. 2017). *Pacemaker Could Hold Key in Arson Case*. Accessed: Dec. 18, 2017. [Online]. Available: http://edition.cnn.com/2017/02/08/us/pacemaker-arson—trnd/index.html

[8] T. Connor. (Apr. 2017). *Fitbit Murder Case: Richard Dabate Pleads Not Guilty in Wifes Death*. Accessed: Dec. 14, 2017. [Online]. Available: https://www.nbcnews.com/news/us-news/fitbit-murder-case-richard-dabate-pleads-not-guilty-wife-s-n752526

[9] N. Chauriye, "Wearable devices as admissible evidence: Technology is killing our opportunity to lie," *Catholic Univ. J. Law Technol.*, vol. 24, no. 2, p. 9, 2016.

[10] K. E. Vinez. (2017). *The Admissibility of Data Collected From Wearable Devices*. Accessed: Dec. 14, 2017. [Online]. Available: https://www2.stetson.edu/advocacy-journal/wp-content/uploads/2017/06/Vinez_-_Wearables.pdf

[11] M. Siddiqi, V. Sivaraman, and S. Jha, "Timestamp integrity in wearable healthcare devices," in *Proc. IEEE Int. Conf. Adv. Netw. Telecommun. Syst. (ANTS)*, Banglore, India, Nov. 2016, pp. 1–6.

[12] H. Fereidooni *et al.*, "Breaking fitness records without moving: Reverse engineering and spoofing fitbit," 2017. [Online]. Available: arXiv:1706.09165.

[13] J. Radcliffe, "Hacking medical devices for fun and insulin: Breaking the human scada system," in *Proc. Black Hat Conf. Presentation Slides*, vol. 2011, 2011.

[14] C. Li, M. Zhang, A. Raghunathan, and N. K. Jha, "Attacking and defending a diabetes therapy system," in *Security and Privacy for Implantable Medical Devices*. New York, NY, USA: Springer, 2014, pp. 175–193.

[15] L. Reverberi and D. Oswald, "Breaking (and fixing) a widely used continuous glucose monitoring system," in *Proc. 11th Workshop Offensive Technol.*, 2017, p. 18.

[16] Q. Zhang and Z. Liang, "Security analysis of bluetooth low energy based smart wristbands," in *Proc. 2nd Int. Conf. Frontiers Sens. Technol. (ICFST)*, Shenzhen, China, 2017, pp. 421–425.

[17] M. Siddiqi, S. T. Ali, and V. Sivaraman, "Secure opportunistic contextual logging for wearable healthcare sensing devices," *IEEE Trans. Depend. Secure Comput.*, early access, Jul. 9, 2019, doi: 10.1109/TDSC.2019.2927674

[18] A. Raij, A. Ghosh, S. Kumar, and M. Srivastava, "Privacy risks emerging from the adoption of innocuous wearable sensors in the mobile environment," in *Proc. SIGCHI Conf. Human Factors Comput. Syst.*, 2011, pp. 11–20.

[19] C. McGoogan. (Nov. 2016). *Fitness Trackers Breaking Privacy Laws, Says Watchdog*. Accessed: Dec. 18, 2017. [Online]. Available: http://www.telegraph.co.uk/technology/2016/11/03/fitness-trackers-breaking-privacy-laws-says-watchdog/

[20] L. Hender. (Jun. 2017). *Surveillance, Safety and Stress: Concerns Over 'Big Brother' Wearables*. Accessed: Dec. 18, 2017. [Online]. Available: https://www.officegenie.co.uk/blog/20170606-surveillance-safety-stress-concerns-over-big-brother-wearables

[21] V. G. Motti and K. Caine, "Users' privacy concerns about wearables," in *International Conference on Financial Cryptography and Data Security*. Heidelberg, Germany: Springer, 2015, pp. 231–244.

[22] S. T. Ali, V. Sivaraman, D. Ostry, G. Tsudik, and S. Jha, "Securing first-hop data provenance for bodyworn devices using wireless link fingerprints," *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 12, pp. 2193–2204, Dec. 2014.

[23] C. Hauser. (Apr. 2017). *In Connecticut Murder Case, a Fitbit Is a Silent Witness*. Accessed: Dec. 14, 2017. [Online]. Available: https://www.nytimes.com/2017/04/27/nyregion/in-connecticut-murder-case-a-fitbit-is-a-silent-witness.html

[24] J. Wieczner. (Jun. 2016). *Fitbit Users Are Finding Creative Ways to Cheat*. Accessed: Dec. 18, 2017. [Online]. Available: http://fortune.com/2016/06/10/fitbit-hack-cheat/

[25] R. Fogarty. (Mar. 2016). *CommInsure: Who's Who in the Commonwealth Bank's Life Insurance Scandal?* Accessed: Jun. 13, 2016. [Online]. Available: http://www.abc.net.au/news/2016-03-07/comminsure-scandal-whos-who-four-corners/7226576

[26] Warrant. (Jan. 2018). *Search Warrant for Google Accounts in Murder Case*. Accessed: Aug. 21, 2018. [Online]. Available: https://www.documentcloud.org/documents/4388571-20170308-homicide-warrant.html

[27] T. Dukes. (Mar. 2018). *To Find Suspects, Police Quietly Turn to Google*. Accessed: Aug. 21, 2018. [Online]. Available: https://www.wral.com/Raleigh-police-search-google-location-history/17377435/

[28] R. Kimitch. (Jan. 2015). *Why Eliminating Witness Fears Is Key to Solving Homicides*. Accessed: Aug. 21, 2018. [Online]. Available: https://www.dailynews.com/2015/01/25/why-eliminating-witness-fears-is-key-to-solving-homicides/

[29] D. Koscieniewski. (Jul. 2007). *A Little Girl Shot, and a Crowd That Didn't See*. Accessed: Aug. 21, 2018. [Online]. Available: https://www.nytimes.com/2007/07/09/nyregion/09taj.html

[30] D. Bannister. (Apr. 2018). *New Protection for Witnesses*. Accessed: Aug. 21, 2018. [Online]. Available: https://thenassauguardian.com/2018/04/19/new-protection-for-witnesses/

[31] D. J. Pohly, S. McLaughlin, P. McDaniel, and K. Butler, "Hi-Fi: Collecting high-fidelity whole-system provenance," in *Proc. 28th Annu. Comput. Security Appl. Conf.*, 2012, pp. 259–268.

[32] B. Przydatek, D. Song, and A. Perrig, "SIA: Secure information aggregation in sensor networks," in *Proc. 1st Int. Conf. Embedded Netw. Sens. Syst.*, 2003, pp. 255–265.

[33] H.-S. Lim, Y.-S. Moon, and E. Bertino, "Provenance-based trustworthiness assessment in sensor networks," in *Proc. 7th Int. Workshop Data Manag. Sens. Netw.*, 2010, pp. 2–7.

[34] H.-S. Lim, G. Ghinita, E. Bertino, and M. Kantarcioglu, "A game-theoretic approach for high-assurance of data trustworthiness in sensor networks," in *Proc. IEEE 28th Int. Conf. Data Eng. (ICDE)*, Washington, DC, USA, 2012, pp. 1192–1203.

[35] U. Braun, A. Shinnar, and M. I. Seltzer, "Securing provenance," in *Proc. 3rd Conf. Hot Topics Security (HotSec)*, 2008, pp. 1–5.

[36] A. De Santis, A. Castiglione, G. Cattaneo, G. De Maio, and M. Ianulardo, "Automated construction of a false digital alibi," in *Proc. Int. Cross Domain Conf. Availability Rel. Security Business Enterprise Health Inf. Syst.*, 2011, pp. 359–373.

[37] P. Albano, A. Castiglione, G. Cattaneo, G. De Maio, and A. De Santis, "On the construction of a false digital alibi on the android OS," in *Proc. IEEE 3rd Int. Conf. Intell. Netw. Collaborative Syst. (INCoS)*, Fukuoka, Japan, 2011, pp. 685–690.

[38] A. Castiglione, G. Cattaneo, G. De Maio, and A. De Santis, "Automatic, selective and secure deletion of digital evidence," in *Proc. IEEE Int. Conf. Broadband Wireless Comput. Commun. Appl. (BWCCA)*, Barcelona, Spain, 2011, pp. 392–398.

[39] J. Liu and W. Sun, "Smart attacks against intelligent wearables in people-centric Internet of Things," *IEEE Commun. Mag.*, vol. 54, no. 12, pp. 44–49, Dec. 2016.

[40] R. Accorsi, "Safe-keeping digital evidence with secure logging protocols: State of the art and challenges," in *Proc. 5th Int. Conf. IT Security Incident Manag. IT Forensics (IMF)*, Stuttgart, Germany, 2009, pp. 94–110.

[41] A. De La Piedra, A. Braeken, A. Touhafi, and K. Wouters, "Secure event logging in sensor networks," *Comput. Math. Appl.*, vol. 65, no. 5, pp. 762–773, 2013.

[42] R. Hasan, R. Khan, S. Zawoad, and M. M. Haque, "WORAL: A witness oriented secure location provenance framework for mobile devices," *IEEE Trans. Emerg. Topics Comput.*, vol. 4, no. 1, pp. 128–141, Jan.–Mar. 2016.

[43] B. Shebaro, S. Sultana, S. R. Gopavaram, and E. Bertino, "Demonstrating a lightweight data provenance for sensor networks," in *Proc. ACM Conf. Comput. Commun. Security*, 2012, pp. 1022–1024.

[44] S. Sultana, G. Ghinita, E. Bertino, and M. Shehab, "A lightweight secure provenance scheme for wireless sensor networks," in *IEEE 18th Int. Conf. Parallel Distrib. Syst. (ICPADS)*, Singapore, 2012, pp. 101–108.

[45] S. Sultana, G. Ghinita, E. Bertino, and M. Shehab, "A lightweight secure scheme for detecting provenance forgery and packet dropattacks in wireless sensor networks," *IEEE Trans. Depend. Secure Comput.*, vol. 12, no. 3, pp. 256–269, May/Jun. 2015.

[46] Z. Li, W. Xu, R. Miller, and W. Trappe, "Securing wireless systems via lower layer enforcements," in *Proc. 5th ACM Workshop Wireless Security*, 2006, pp. 33–42.

[47] S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik, "Radio-telepathy: Extracting a secret key from an unauthenticated wireless channel," in *Proc. 14th ACM Int. Conf. Mobile Comput. Netw.*, 2008, pp. 128–139.

[48] M. H. Chinaei, V. Sivaraman, and D. Ostry, "An experimental study of secret key generation for passive Wi-Fi wearable devices," in *Proc. IEEE 18th Int. Symp. World Wireless Mobile Multimedia Netw. (WoWMoM)*, Macau, China, 2017, pp. 1–9.

[49] S. T. Ali, V. Sivaraman, and D. Ostry, "Zero reconciliation secret key generation for body-worn health monitoring devices," in *Proc. 5th ACM Conf. Security Privacy Wireless Mobile Netw.*, 2012, pp. 39–50.

[50] L. W. Hanlen, D. Smith, J. A. Zhang, and D. Lewis, "Key-sharing via channel randomness in narrowband body area networks: Is everyday movement sufficient?" in *Proc. 4th Int. Conf. Body Area Netw.*, 2009, p. 17.

[51] L. Yao, S. T. Ali, V. Sivaraman, and D. Ostry, "Improving secret key generation performance for on-body devices," in *Proc. 6th Int. Conf. Body Area Netw.*, 2011, pp. 19–22.

[52] S. T. Ali, V. Sivaraman, and D. Ostry, "Eliminating reconciliation cost in secret key generation for body-worn health monitoring devices," *IEEE Trans. Mobile Comput.*, vol. 13, no. 12, pp. 2763–2776, Dec. 2014.

[53] S. Jana, S. N. Premnath, M. Clark, S. K. Kasera, N. Patwari, and S. V. Krishnamurthy, "On the effectiveness of secret key extraction from wireless signal strength in real environments," in *Proc. 15th Annu. Int. Conf. Mobile Comput. Netw.*, 2009, pp. 321–332.

[54] S. T. Ali, V. Sivaraman, D. Ostry, and S. Jha, "Securing data provenance in body area networks using lightweight wireless link fingerprints," in *Proc. 3rd Int. Workshop Trustworthy Embedded Devices*, 2013, pp. 65–72.

[55] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Commun. ACM*, vol. 13, pp. 422–426, Jul. 1970.

[56] S. Saroiu and A. Wolman, "Enabling new mobile applications with location proofs," in *Proc. 10th Workshop Mobile Comput. Syst. Appl.*, 2009, p. 1–6.

[57] C. Stamford. (Sep. 2014). *Gartner Says a Typical Family Home Could Contain More Than 500 Smart Devices by 2022*. Accessed: Dec. 30, 2019. [Online]. Available: https://www.gartner.com/en/newsroom/press-releases/2014-09-08-gartner-says-a-typical-family-home-could-contain-more-than-500-smart-devices-by-2022

[58] *Telehealth Framework and Implementation Strategy 2016–2021*, NSW-Health, North Sydney NSW, Australia, 2016. Accessed: Dec. 30, 2019. [Online]. Available: https://www.health.nsw.gov.au/telehealth/Publications/NSW-telehealth-framework.pdf

[59] J. J. Moffatt and D. S. Eley, "The reported benefits of Telehealth for rural Australians," *Australian Health Rev.*, vol. 34, no. 3, pp. 276–281, 2010.

[60] C. Griffith *et al.*, *Smart Farming: Leveraging the Impact of Broadband and the Digital Economy*, CSIRO, Canberra ACT, Australia, 2013.

[61] *Blockchain Notary*, SilentNotary, Belize City, Belize, 2018. Accessed: Nov. 17, 2018. [Online]. Available: https://silentnotary.com

[62] *Blockchain Notary*, Stampd, Los Angeles, CA, USA, 2018. Accessed: Nov. 17, 2018. [Online]. Available: https://stampd.io

[63] *Key Directory*, KeyBase, New York, NY, USA, 2018. Accessed: Nov. 17, 2018. [Online]. Available: https://keybase.io

[64] U. W. Chohan. (Nov. 30, 2017). *The Cryptocurrency Tumblers: Risks, Legality and Oversight, Discussion Paper Series: Notes on the 21st Century*. [Online]. Available: https://ssrn.com/abstract=3080361

[65] C. Jost, H. Lam, A. Maximov, and B. J. Smeets, "Encryption performance improvements of the paillier cryptosystem," *IACR Cryptol. ePrint Archive*, vol. 2015, p. 864, Sep. 2015.

[66] M. Jakobsson, A. Juels, and R. L. Rivest, "Making mix nets robust for electronic voting by randomized partial checking," in *Proc. USENIX Security Symp.*, San Francisco, CA, USA, 2002, pp. 339–353.

[67] D. Chaum *et al.*, "Scantegrity: End-to-end voter-verifiable optical- scan voting," *IEEE Security Privacy*, vol. 6, no. 3, pp. 40–46, May/Jun. 2008.

[68] P. Y. Ryan, D. Bismark, J. Heather, S. Schneider, and Z. Xia, "Prêtà voter: A voter-verifiable voting system," *IEEE Trans. Inf. Forensics Security*, vol. 4, no. 4, pp. 662–673, Dec. 2009.

[69] J. Ben-Nun *et al.*, "A new implementation of a dual (paper and cryptographic) voting system," in *Proc. 5th Int. Conf. Electron. Voting*, 2012, pp. 315–329.

[70] S. Barlyn, (Sep. 2018). *Strap on the Fitbit: John Hancock to Sell Only Interactive Life Insurance*. Accessed: Nov. 12, 2018. [Online]. Available: https://www.reuters.com/article/us-manulife-financi-john-hancock-lifeins/strap-on-the-fitbit-john-hancock-to-sell-only-interactive-life-insurance-idUSKCN1LZ1WL

[71] N. Patwari, J. Croft, S. Jana, and S. K. Kasera, "High-rate uncorrelated bit extraction for shared secret key generation from channel measurements," *IEEE Trans. Mobile Comput.*, vol. 9, no. 1, pp. 17–30, Jan. 2010.

[72] S. Tarkoma, C. E. Rothenberg, and E. Lagerspetz, "Theory and practice of bloom filters for distributed systems," *IEEE Commun. Surveys Tuts.*, vol. 14, no. 1, pp. 131–155, 1st Quart., 2012.

[73] C.-C. Chang, S. Muftic, and D. J. Nagel, "Measurement of energy costs of security in wireless sensor nodes," in *Proc. IEEE 16th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Honolulu, HI, USA, 2007, pp. 95–102.

[74] J.-P. Kaps and B. Sunar, "Energy comparison of AES and SHA-1 for ubiquitous computing," in *International Conference on Embedded and Ubiquitous Computing*. Heidelberg, Germany: Springer, 2006, pp. 372–381.

[75] G. De Meulenaer, F. Gosset, F.-X. Standaert, and O. Pereira, "On the energy cost of communication and cryptography in wireless sensor networks," in *Proc. IEEE Int. Conf. Wireless Mobile Comput. Netw. Commun. (WIMOB'08)*, Avignon, France, 2008, pp. 580–585.

[76] S. T. Ali, V. Sivaraman, D. Ostry, "Authentication of lossy data in body-sensor networks for cloud-based healthcare monitoring," *Future Gener. Comput. Syst.*, vol. 35, pp. 80–90, Jun. 2014.

[77] J. Blasco, T. M. Chen, J. Tapiador, and P. Peris-Lopez, "A survey of wearable biometric recognition systems," *ACM Comput. Surveys (CSUR)*, vol. 49, no. 3, p. 43, 2016.

[78] M. D. Ryan, "Making decryption accountable," in *Cambridge International Workshop on Security Protocols*. Cham, Switzerland: Springer, 2017, pp. 93–98.

Muhammad Siddiqi received the degree in electronics from Quaid-i-Azam University, Islamabad, Pakistan, in 2004, and the master's degree in telecommunications and the Ph.D. degree in electrical engineering from the University of New South Wales, Sydney, NSW, Australia, in 2012 and 2019, respectively.

He worked with Nanjing R&D Centre, ZTE Corporation, Shenzhen, China, from 2005 to 2009. He is currently working as a Lecturer with the Australian Institute of Higher Education, Sydney. His research interests include body area networks, Internet of Things, and network security.

Syed Taha Ali received the B.Sc. (Eng.) degree from the GIK Institute of Engineering Sciences and Technology, Swabi, Pakistan, in 2002, and the M.S. and Ph.D. degrees in electrical engineering from the University of New South Wales, Sydney, NSW, Australia, in 2006 and 2012, respectively.

He was a Postdoctoral Research Fellow with the University of New South Wales in 2013, and Newcastle University, Tyne, U.K., from 2014 to 2016. He is currently an Assistant Professor with the National University of Science and Technology, Islamabad, Pakistan. His research interests include body area networks, software-defined networking, network security, and cryptocurrencies.

Vijay Sivaraman received the B.Tech. degree from the Indian Institute of Technology Delhi, New Delhi, India, in 1994, the M.S. degree from North Carolina State University, Raleigh, NC, USA, in 1996, and the Ph.D. degree from the University of California at Los Angeles, Los Angeles, CA, USA, in 2000.

He has worked with Bell-Labs, Murray Hill, NJ, USA, and a Silicon Valley startup manufacturing optical switch-routers. He is currently a Professor with the University of New South Wales, Sydney, NSW, Australia. His research interests include software-defined networking and IoT technologies.