I Know Where You are: Proofs of Presence Resilient to Malicious Provers

Markus Miettinen
Technische Universität Darmstadt

N. Asokan
Aalto University and University of Helsinki

Farinaz Koushanfar
Rice University

Thien Duc Nguyen
Technische Universität Darmstadt

Jon Rios
Technische Universität Darmstadt

Ahmad-Reza Sadeghi
Technische Universität Darmstadt

Majid Sobhani
Technische Universität Darmstadt

Sudha Yellapantula
Rice University

ABSTRACT

In the recent years, new services and businesses leveraging location-based services (LBS) are rapidly emerging. On the other hand this has raised the incentive of users to cheat about their locations to the service providers for personal benefits. Context-based proofs-of-presence (PoPs) have been proposed to enable verification of users' location claims. However, as we show in this paper, they are vulnerable to *context guessing attacks*. To make PoPs resilient to malicious provers we propose two complementary approaches for making context-based PoPs: one approach focuses on *surprisal filtering* based on estimating the entropy of particular PoPs in order to detect context measurements vulnerable to such attacks. The other approach is based on utilizing longitudinal observations of ambient modalities like noise level and ambient luminosity. It is capable of extracting more entropy from the context to construct PoPs that are hard to guess by an attacker even in situations in which other context sensor modalities fail to provide reliable PoPs.

1. INTRODUCTION

Contemporary mobile devices are capable of utilizing a range of positioning technologies such as GPS or network triangulation to find their locations. Therefore, new applications and services leveraging the mobile device's locationing abilities are rapidly emerging. For instance, Facebook and other online social networks (OSNs) extensively utilize location "check-ins" of users to enhance their services; Foursquare [8] uses the location information to connect users to local businesses like shops or restaurants; a number of business owners offer concrete benefits such as free vouchers, special discount, and even cash value to the most active registrants visiting their shops or restaurants.

The LBS business model is built upon the premise of trustworthiness of mobile users. However, as the LBS and businesses are on the rise, so are the clients' incentives to engage in *location cheating* for their personal benefit. Misbehaving users may obtain unjustified benefits at particular venues by repeatedly making false location check-ins. "Fake location" applications that aid the cheating clients are already available for popular smartphone platforms.

A drawback of currently deployed positioning technologies is that they rely on the mobile client to perform the positioning operation. It is difficult for external entities to determine whether the location claimed by a client device is in fact correct. Therefore, there is a need for *location proofs*: methods for verifying the correctness of location claims that clients present to the LBS. In peer-to-peer scenarios, mobile devices may require proofs of co-presence from other devices such that they can control their visibility. For example, a device might want to reveal its presence in a particular location only to those peer devices that are present in the same location [10]. Therefore, the devices need to be able to verify that a location claim made by a peer device indicating proximity is indeed genuine. The peer device is required to present a *proof of co-presence* to establish the validity of their claim. In both cases, we model the situation as follows: a *prover* aims to provide a *proof-of-presence (PoP)* to a *verifier* that they are in the same context, i.e., present in the same proximate environment.

Prior work has suggested two main classes of solutions for constructing PoPs: *beaconing* and *context-based PoPs*. The former class of proofs is based on active beaconing of information by the verifier into its immediate vicinity. The potential provers are then required to capture this information using their on-board sensors (e.g., WiFi or Bluetooth). The beaconed information is utilized by the prover either directly as the proof or in a proof-of-knowledge protocol with the verifier. The underlying presumption of this approach is that only a device actually co-present with the verifier is able to accurately capture the information beaconed by the verifier. In peer-to-peer scenarios, beaconing information into the context has the drawback that the verifier has to reveal its presence in the context. However, revealing presence may be undesirable because of its adverse impact on the verifier's privacy. For example, if beaconing is realized using a WiFi or Bluetooth channel, the verifying device needs to actively emit the beacon information and thus expose its own MAC address.

In this paper, we focus on context-based PoPs. These are based on simultaneous sensing of contextual data by both the verifier and the prover. A number of such methodologies have been proposed [5, 10, 14–16]. In this setting the prover and verifier concurrently sample their incident context via sensors. The supposition is that the transient contextual fluctuations cannot be exactly sensed or predicted by an attacker outside the context. These measurements are either directly used to generate a common key (e.g., [16]), or the prover sends its measurements to the verifier who compares them to its own measurements. Because of the sensing and synchronization jitters, the measurements often contain noise. If the (noisy) measurements are similar enough, this constitutes a PoP. In the following, we denote such proofs as *context-based proofs of presence*.

Earlier work on context-based PoPs do not consider the possibility of *context-guessing attacks*, either because these are out of scope [15, 16], or, they assume that the used context modalities provide sufficient entropy so that attackers are not able to fabricate context-based proofs, [10].

Our goal and conttributions: In this paper, we empirically analyze such attacks against commonly used context sensor modalities such as Bluetooth and WiFi, thus demonstrating that for reliable PoPs, the entropy of individual context observations needs to be taken into account also.

We address context guessing attacks by proposing two complementary approaches: First we show how by using *surprisal filtering* we can make sure that only PoPs with sufficient entropy are admitted as valid PoPs. The approach is based on applying data mining methods for profiling the context and estimating the occurrence probability of particular context parameter combinations in them. Second we make use of longitudinal ambient context observations to extract inherent randomness from the context that contains sufficient entropy to make context guessing attacks in most cases impractical. Earlier approaches utilize ambient context data (e.g., [5, 15]), in which only momentary snapshots of the context are considered. In contrast, we monitor the context and short-term changes in the context's physical parameters over a longer time period and utilize these changes in the context as a means to extract sufficient entropy to construct a reliable PoP.

We make the following contributions:

- We empirically analyze the feasibility of *context-guessing attacks* on context modalities that have earlier been used for co-location verification (Bluetooth and WiFi) and show that these modalities are in fact vulnerable to attacks in which a malicious prover fabricates context-based PoPs to cheat about its location.

- We propose two countermeasures to mitigate context-guessing attacks: *surprisal filtering* which is based on profiling and estimating the entropy associated with individual PoPs, and, the use of longitudinal observations of ambient physical properties of the context. We show based on empirical data that surprisal filtering provides an efficient method for identifying potentially too weak PoPs and demonstrate how to use longitudinal context data in such cases to extract sufficient entropy from the context to construct a reliable context-based proof-of-presence.

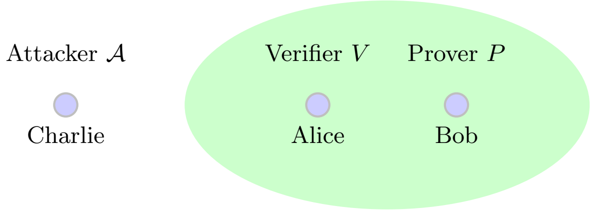

Figure 1: Scenario 1: Peer co-presence

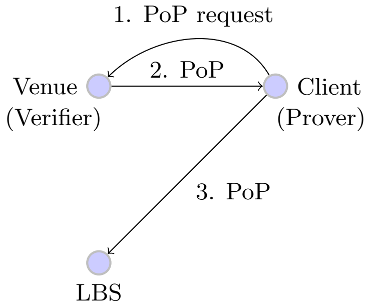

Figure 2: Scenario 2: Presence at a venue

2. PROBLEM SETTING

Context-based proofs-of-presence play a role in proofs of (co-)presence between peers and proofs of presence at a venue with regard to a Location-Based Service (LBS). Unlike some earlier works that consider the comparison of context information for creating a pairing between the principals (e.g., [16]), we assume that there already exists a security association between the parties. Therefore, we are not considering the threat of man-in-the-middle or eavesdropping attacks in this discussion. We focus on the problem of one party lying about its own context to the other party.

The first scenario is depicted in Fig. 1: Alice, Bob and Charlie are friends in an Online Social Network (OSN). Alice is willing to automatically share her status information or engage in communications, like instant chat with her OSN friends who are present in the same context as she is (i.e., Bob), but not with others (e.g., Charlie). Therefore, Bob needs to prove co-presence to Alice in order to get connected to her.

In the other scenario depicted in Fig. 2, a client of an LBS wants to prove to the LBS its presence at a venue (e.g., a restaurant or a shop) in order to obtain benefits like rebates or gift cards given out to loyal customers of the venue. The LBS cannot rely on unilateral presence claims of the client, since the client has an incentive to cheat in order to obtain the above-mentioned benefits. Independent proof of the client's presence provided by the venue is therefore required.

2.1 Threat model and Assumptions

In both scenarios, the attacker $\mathcal{A}$ is a malicious prover, who *fabricates* a PoP in order to cheat the verifier V into believing that $\mathcal{A}$ is in the same context as V.

In Scenario 1 $\mathcal{A}$ is a malicious user, e.g., Charlie in Fig. 1, who engages in a cyberstalking attack and wants to reveal a

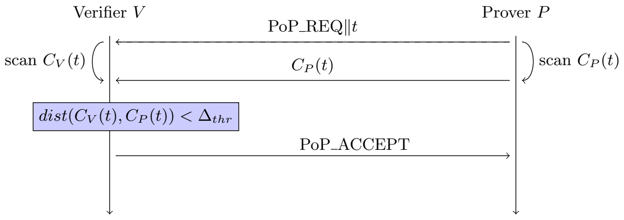

Figure 3: Context-based proof-of-presence

target user's (e.g., Alice in Fig. 1) location without actually being co-located with her. To do this, Charlie fabricates location claims for places which he knows Alice is known to visit, and waits for which location Alice responds to his location claim, thereby learning Alice's whereabouts. Alice acts as the verifier V and Charlie as the malicious prover $\mathcal{A}$.

For Scenario 2, $\mathcal{A}$ is a malicious client of the LBS, who wants to do fake location check-ins at a venue for obtaining benefits even though he has not visited the venue in reality. The venue acts as the verifier V. The malicious client $\mathcal{A}$ fabricates PoPs and presents them to the venue V. If V falsely accepts $\mathcal{A}$'s PoP as genuine, it will issue a PoP to $\mathcal{A}$, which $\mathcal{A}$ can then use to falsely convince the LBS to believe that $\mathcal{A}$ has visited the venue in question.

In particular, $\mathcal{A}$ is likely to target such contexts that it can monitor over an extended period of time in order to obtain infromation that is useful for fabricating PoPs. Therefore we have to assume that for any context X, the attacker $\mathcal{A}$ has acquired a rich context profile that it can utilize in maximizing its chances of fabricating a PoP that would be accepted by V.

2.2 Problem Definition

The basic mechanism for providing context-based PoPs which is applicable in the above scenarios is shown in Fig. 3. Both the verifier V and prover P record a context measurement $C_V(t)$ and $C_P(t)$ at time point t. The prover P then sends its context measurement $C_P(t)$ to the verifier V, who compares it with its own context measurement $C_V(t)$ and determines, whether $C_P(t)$ is similar enough to accept it as a proof-of-presence of P. The rationale behind such PoPs is that devices in the same context will observe roughly the same contextual events and environmental conditions and therefore their context measurements will be more similar than context measurements of devices that are not in the same context.

To mitigate the risk of V erroneously accepting PoPs that the attacker $\mathcal{A}$ has fabricated, V needs to be able to evaluate the risk that a particular PoP could be fabricated in that context. We therefore need a way to determine the entropy of V's context measurements C_V, i.e., how difficult it would be for $\mathcal{A}$ to fabricate a valid $C_{\mathcal{A}}$. In addition, to address such cases in which the context measurement of V would be too easily guessed by $\mathcal{A}$, we need to augment the context measurements used in the PoPs with modalities that contain sufficient entropy against this guessing attack.

3. CONTEXT GUESSING

A malicious prover $\mathcal{A}$ may try to make the verifier V believe that he is in the V's context X, even though he is located somewhere else, for example to make false location chek-ins at a venue so that he will obtain unjustified benefits from the venue. One way for $\mathcal{A}$ to achieve this is to launch a *context guessing attack*. $\mathcal{A}$ *fabricates* a context measurement $C_{\mathcal{A}}$ and presents it as a context-based PoP to V. If the fabricated measurement is similar enough to V's measurement C_V, V will falsely accept it.

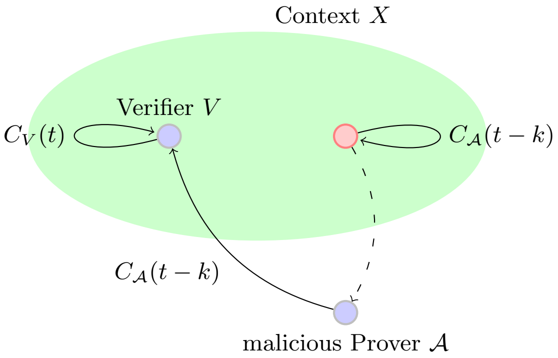

Figure 4: Overview of the context guessing attack

For example, when executing the attack at timepoint t, as shown in Fig. 4, $\mathcal{A}$ can replay an old context measurement $C_{\mathcal{A}}(t-k)$ it obtained while visiting context X at an earlier timepoint $t-k$, claiming it to be his current context measurement. The attack will succeed, if $C_{\mathcal{A}}(t-k)$ is similar enough to the verifier's measurement $C_V(t)$. Alternatively, $\mathcal{A}$ could also fabricate the context measurement $C_{\mathcal{A}}$ by building a model of the context X by using several *earlier* measurements $C_{\mathcal{A}}$ made in the target context X.

In earlier works, context-based co-location verification has been based on direct measurements of contextual values in different modalities. For example, the acoustic environment [5, 15], ambient light [5], atmospheric gases, temperature, humidity and air pressure [14], as well as WiFi [16], Bluetooth and GPS [15], have been investigated as modalities for contextual proofs of presence.

In their very recent work, Truong et al. found that the sets of WiFi and Bluetooth devices observed along with their received signal strengths provide good performance in co-location verification [15]. We therefore decided to use WiFi and Bluetooth as the basic modalities for PoPs and tested whether PoPs based on Bluetooth or WiFi are vulnerable to the context guessing attack. The authors of [15] kindly provided us the dataset they used for their experiments so that we could make a direct comparison with their results. Note, however, that their usage scenario relates to zero-interaction authentication (ZIA) settings, where the attack model is different: their co-location verification is intended to protect against relay attacks because the prover and the verifier mutually trust each other. In contrast, in our scenario the potential attacker is a malicious prover, rather than the threat of a relay attack.

3.1 Attack Implementation

In our implementation, V uses a classification model to distinguish between co-located and non-co-located context measurements. V trains his classification model with a benign dataset containing examples of co-located and non-co-located measurement pairs. To test the model's performance against context guessing attacks, we construct an attack dataset where benign verifier V measurements are paired by measurements that could have been fabricated by $\mathcal{A}$ by replaying all measurements from the same context that were made 6 to 24 hours earlier.

From the measurement pairs, V calculates a set of features that represent different distance measures between the measurements and combines them to feature vectors. Feature vectors calculated based on the benign dataset are used to train the classification model, whereas features from the attack dataset are used as testing data to evaluate the classifier's performance against the context guessing attack. The classification algorithm used is Multiboost in combination with J48 Graft as the base learner. We used the Weka data mining suite [6] to execute our experiments.

For training the classifier we used the following features:

Feature 1 (Jaccard distance).

$$J_\delta(C_V, C_P) = 1 - \frac{\|C_V \cap C_P\|}{\|C_V \cup C_P\|}, \tag{1}$$

Feature 2 (Mean of Hamming distance).

$$H_\delta(C_V, C_P) = \frac{\sum_{i=1,2,\ldots,n} |m_i^P - m_i^V|}{n} \tag{2}$$

Feature 3 (Euclidean distance).

$$E_\delta(C_V, C_P) = \sqrt{\sum_{i=1,2,\ldots,n} (m_i^P - m_i^V)^2} \tag{3}$$

Feature 4 (Mean exponential of difference).

$$\Xi_\delta(C_V, C_P) = \frac{\sum_{i=1,2,\ldots,n} e^{|m_i^P - m_i^V|}}{n} \tag{4}$$

where $m_i^V \in C_V$ and $m_i^P \in C_P$ denote the individual elements of the context measurements of the verifier V and prover P, respectively.

Feature 5 (Sum of squared rank differences).

$$\rho_\delta(C_V, C_P) = \sum_{i=1}^{|C_V \cap C_P|} (\mathit{rank}(m_i^P) - \mathit{rank}(m_i^V))^2 \tag{5}$$

where $\mathit{rank}(m_i^P)$ and $\mathit{rank}(m_i^V)$ denote the ranks of m_i^P and m_i^V in C_P and C_V, respectively, sorted in ascending order.

3.2 Datasets

To evaluate the feasibility of context guessing attacks, we used two datasets: the zero-interaction authentication dataset by Truong et al., and the *ConXPoP dataset*, which we collected to test context guessing attacks and countermeasures against it. The ZIA dataset was primarily used to demonstrate the feasibility of the attack, whereas the ConXPoP dataset contains more context modalities and an explicit context labeling which we used to examine possible countermeasures against the context guessing attack.

ZIA Dataset contained measurements of the MAC addresses of visible Bluetooth devices and WiFi access points and their received signal strengths, simultaneously collected from two devices. The dataset contained a total of 2302 sample pairs, out of which 1140 were such that the devices were co-located, and 1162 pairs were samples from non-co-located devices. We used this dataset to derive features to train the benign dataset for training the classification model of the verifier V. As a baseline, we examined the classifier's performance on the benign dataset using 10-fold cross-validation, and could corroborate the results of [15], obtaining a false positive (FP) rate of 2.5 % for Bluetooth features and 1.6 % for WiFi features.

The attack dataset simulating context replay attacks was constructed by remapping the experiments in ZIA dataset by pairing measurements that were made in the same location, but at different times. The ZIA dataset contained ground truth labels telling whether measurement pairs were co-located or not, but the actual location in which the measurements had been made was not included in the dataset. Therefore we had to use the set of observed WiFi access points associated with each measurement as representing the location in which the measurement had been made.

To obtain a criterion by which to decide whether measurements made at different times were made at the same location, we compared the co-located measurement pairs to the non-co-located ones in the ZIA dataset and observed that a Jaccard distance value of 0.9 for the sets of observed WiFi devices provided a good separation between co-located and non-co-located measurement pairs. We therefore concluded that if the Jaccard distance of two measurements is less than 0.9, we can assume that these measurements were made in the same location.

We then paired each experiment measurement with such measurements for which the Jaccard distance between the sets of WiFi measurements was below 0.9, i.e., that were made in the same location, but at a different time.

ConXPoP Dataset data collection was done using a purpose-built app running on Android smartphones given out to study participants. The app continuously measured contextual parameters and periodically uploaded them to a server for off-line data analysis. The collected data included link layer identifiers and observed signal strengths for WiFi and Bluetooth devices in proximity (sampled once a minute), as well as a continuous trace of the ambient noise level and luminosity, as observed by the smartphone's sensors.

Participants included volunteers from the research lab staff sharing nearby offices and visiting the same lunchtime restaurant. This enabled the participants to provide a rich dataset of co-located measurements arising from natural everyday situations.

All participants were informed in writing about the purpose, goals and content of the data collection campaign beforehand. Participants were free to stop or interrupt data collection at any point by disabling the data collection app. All participants were also given the possibility to revoke their participation in the experiment by demanding the data collected by them to be deleted.

Participants were asked to provide, via the user interface of the app, information about particular contexts that they were visiting (e.g., Home, Office, Restaurant, etc.) and which other participant devices were co-located with the user's own context collector device. Devices of other participants were identified using easily recognizable nicknames. Participants were asked to mark only such other devices as co-located that were likely to be present in the same room with the user for the following two minutes.

Furthermore, in order to obtain examples of co-located observations from contexts where typically only one test participant is present (e.g. the test participants' homes), each test

Table 1: Results of the context guessing attacks

	FP Rate		
Dataset	BT	WiFi	BT+WiFi
ZIA benign	2.5%		
ConXPoP benign	14.2%	11.0%	9.3%
ZIA attack	35.1%		
Increase in FP rate	+32.6%		
ConXPoP attack	21.9%	26.0 %	23.5%
Increase in FP rate	+7.7%	+15.0%	+14.2%

participant was provided with two context collector devices: a main device and an "alter ego" device. By bringing the alter ego device together with the main device to contexts that no other test participants visited, users could provide co-located context samples also from such contexts.

During a data collection period of 10 days, participants generated a total of 5602 annotated co-located context measurement pairs. Using these data, we constructed for each participant a benign dataset and an attack dataset. The benign dataset for training each user's co-location classifier was constructed by pairing measurement pairs marked as being co-located by the user or some other user with a roughly equal amount of measurement pairs that were not marked as co-located.

The attack dataset was constructed by letting one participant at a time act as the verifier V. For each verifier observation $C_V(t)$ made in a named context X (where $X \in \{$"Home", "Office", "Restaurant"$\}$), potential attacker observations $C_{\mathcal{A}}(t-k)$ made in the same context X were selected allowing all participants to take the role of the malicious prover $\mathcal{A}$. We selected k to be 6 to 24 hours.

3.3 Results

We evaluated both the ZIA dataset and the ConXPoP dataset by training classifiers with the benign datasets and using the attack datasets as testing datasets. As a baseline to compare against, we used 10-fold cross-validation of the training dataset. Table 1 shows the results.

The differences of the attack scenarios to the benign dataset results are clear, showing the effect of the context guessing attack. For both ZIA and ConXPoP attack datasets, the FP rate increases significantly in comparison to the benign dataset results. This difference is especially clear for the ZIA dataset. For the ConXPoP dataset, the change is somewhat smaller, due to the higher FP rate in the benign dataset. This is caused by the more challenging experimental set-up in comparison to the ZIA dataset. Whereas in the ZIA dataset, co-located and non-co-located samples were more clearly separated from eachother, the ConXPoP set up was more ambiguous. The criterion for co-location was that we regard any devices in the same room to be co-located, other devices not [1]. However, in the office context, test participants used office rooms next to one another, so that their devices were not co-located according to the above criterion, but still the devices shared some common WiFi and Bluetooth environment. This makes it more difficult for the classifier to make a clear distinction between co-located and non-co-located observations, resulting in a higher False Positive rate also in the benign dataset.

However, we see that for both datasets, the context guessing attack yields a False Positive rate of 22% to 35%[2]. This gives an attacker a chance of at least one out of five to succeed in a context replay attack, showing that in settings where the prover cannot be trusted by the verifier, context measurements alone cannot provide the basis for a reliable proof of presence. The verifier needs also to have the possibility to assess how large the risk of a guessing attack associated with a PoP is.

4. HARDENING CONTEXT-BASED PROOFS

In this section, we introduce two countermeasures for hardening context-based proofs-of-presence against context guessing attacks. The first countermeasure aims at identifying such PoPs that are potentially easy to guess. We do this by estimating the entropy associated with a particular PoP. This estimation is based on the notion of *surprisal*, i.e., the self-information associated with a particular context observation of the verifier. The notion of surprisal is closely related to entropy but with a difference: surprisal is the uncertainty associated with the *particular outcome* of a random variable, whereas entropy measures the *average* uncertainty associated with a random variable.

In our case, we consider the observed context X of V as a random variable O_X taking particular measured context observations C_V as its value. The surprisal associated with a context measurement C_V is therefore a measure for the uncertainty of that particular outcome of the random variable. We utilize this and use surprisal-based filtering to dismiss such PoPs that can be potentially easily guessed by the attacker $\mathcal{A}$, as described below in Sect. 4.1.

The other countermeasure we propose aims at increasing the entropy of PoPs in order to make context guessing infeasible for the attacker. In contrast to earlier approaches for co-location verification [5, 13, 15, 16], where short momentary snapshots of the context were used to determine co-location, we use a longitudinal approach. By observing the context over a longer time period and observing changes in the context's ambient properties like luminosity and audio, we aim at extracting sufficient entropy from the context to make guessing of the context impractical. This approach is explained in Sect. 4.2.

4.1 Surprisal Filtering

Surprisal filtering is based on estimating how easy it would be for $\mathcal{A}$ to fabricate a PoP $C_{\mathcal{A}}$ that is similar enough to V's context measurement C_V to be accepted as genuine. The estimate is based on profiling V's contexts and utilising the profiled information to estimate the occurrence probabilities of individual context measurements C_V in a context X. Our intuition is that the lower the occurrence probability of a context measurement C is, the more difficult it is for an attacker $\mathcal{A}$ to fabricate the measurement, even if he has monitored the context X earlier. Based on the probability estimate of the proof, V can then reject such proofs, for which the risk of fabrication is high.

[1] This criterion for co-location was selected, since for providing ground truth information, participants needed to be able to visually observe any co-located persons and their associated devices.

[2] We do not report the false positive rates for WiFi for the ZIA dataset, since we use the WiFi observations in the attack dataset as ground truth for identifying measurements made in the same context.

More formally, we define surprisal filtering as a function $\varsigma : \mathcal{C} \times \mathcal{X} \rightarrow \{accept, reject\}$, where $\mathcal{C}$ denotes the domain of context measurements and $\mathcal{X}$ the set of V's known contexts. The surprisal filtering function ς maps a context measurement $C \in \mathcal{C}$ observed in a particular context $X \in \mathcal{X}$ to a filtering decision *accept* or *reject* based on the surprisal value $I_X(C)$ of the measurement in V's context X:

$$\varsigma(C, X) = \begin{cases} accept & I_X(C) \geq I_{thr} \\ reject & \text{otherwise} \end{cases} \tag{6}$$

The calculation of the surprisal value is described in Sect. 4.1.1.

The rationale for this defense is the following: Information representing a context is of two types. Static information, such as the link layer addresses of WiFi access points in an office, has a high probability of appearing in measurements taken in that context at any time. Therefore, an attacker who has previously visited that context is likely to be able to fabricate a context measurement containing such static information even when he is not present in the context. Dynamic context information, such as the Bluetooth link layer addresses of smartphones belonging to customers at a shop, is likely to be volatile and thus harder to predict. Naturally, contexts with more dynamic information are more amenable for reliable context-based PoPs. In the following, we describe a way to measure the 'dynamicity' of the information present in a context at a given time and show how it can be used to enhance protection against context guessing attacks.

4.1.1 Surprisal of Context Measurements

To be able to identify PoPs that are too easy to fabricate, we need to measure how difficult it would be for an attacker to guess a context measurement $C_{\mathcal{A}}$ based on the history of observations in the target context X. Since we are assuming a strong attacker model, we have to assume that the attacker $\mathcal{A}$ has equal opportunity to observe and generate a context profile on X as the target V has, and use this profile to fabricate PoPs that are likely to be observed in X.

To obtain optimal results, $\mathcal{A}$ needs to guess the correct context measurement C_V of V. The difficulty of fabricating a PoP $C_{\mathcal{A}}$ that is accepted by V is therefore dependent on the difficulty of guessing C_V.

We model the occurrence of a specific contextual measurement C (e.g., a set of WiFi or Bluetooth (BT) devices) in context X with the random variable O_X. The probability that a context measurement C is observed in context X is therefore $P(O_X = C)$. The *surprisal* associated with this context measurement is the *self-information* of this outcome.

DEFINITION 1. *The* surprisal *associated with a context observation C in context X is the self-information of this measurement*

$$I_X(C) = log(\frac{1}{P(O_X = C)}) = -log(P(O_X = C)) \tag{7}$$

and is measured in bits[3].

For example, if there is a 50% chance of observing a device d_i in context X, i.e. $C = \{d_i\}$, then the self-information related to an observation of d_i in X is $I_X(C = \{d_i\}) = -log(P(O_X = \{d_i\})) = -log(0.5) = 1$ bit.

In order to calculate the surprisal associated with a measurement C, we need to estimate the probability $P(O_X = C)$. To do this, we adopt a frequentist interpretation of probability and calculate the probability of context measurment C in context X as the fraction of the number of times that C has been observed in X. Hereby, we need to distinguish between measurements that consist of a single contextual event and multi-event measurements consisting of several co-occurring contextual events. In the following, we consider the occurrence of Bluetooth and WiFi devices in the context as contextual events d_i.

Single-Event Measurements. In the case that the measurement consists of a single contextual event d, i.e., $C = \{d\}$, the calculation of surprisal of C is straightforward. We can calculate the estimated probability of the event as the fraction of measurements containing this event within the whole observation history database $\mathcal{H}_X$ for context X.

$$C = \{d\} : P(O_X = C) = \frac{\|\{C_i \in \mathcal{H}_X \mid d \in C_i\}\|}{\|\mathcal{H}_X\|} \tag{8}$$

Multi-Event Measurements. For context measurements containing more than one contextual event, the formulation is slightly more complicated. We cannot merely multiply the probabilities of the individual events, since in reality, the events might be highly correlated with one another, and assuming independence between events could therefore significantly over- or underestimate the true probability of event combinations. Therefore, we need to estimate the probability of a multi-event context measurement through its occurrence frequency in the observation history database. Thus, given a context measurement $C = \{d_1, d_2, \ldots, d_n\}$ that consists of several context elements, the occurrence probability of C can be calculated as

$$P(O_X = C) = \frac{\|\{C_i \in \mathcal{H}_X \mid \forall d_i \in C : d_i \in C_i\}\|}{\|\mathcal{H}_X\|} \tag{9}$$

As an example, let us consider context measurements of Bluetooth devices. Let us assume that we have a total of $n = 100$ context measurements of context X in the context history database $\mathcal{H}_X$. Each measurement represents the set of Bluetooth devices observed in context X during a time window of two minutes. In the observation history, device A has been observed in 55 measurements and device B in 35 measurements. Out of these measurements, 15 are such that both A and B occur in the same measurement. Let us now consider the probability estimates for different context measurements. For individual measurements of the devices A and B, we have $P(O_X = \{A\}) = \frac{\|\{C_i \in \mathcal{H}_X \mid A \in C_i\}\|}{\|\mathcal{H}_X\|} = \frac{55}{100} = 0.55$ and $P(O_X = \{B\}) = \frac{\|\{C_i \in \mathcal{H}_X \mid B \in C_i\}\|}{\|\mathcal{H}_X\|} = \frac{35}{100} = 0.35$. For a measurement containing both devices, the estimate is $P(O_X = \{A, B\}) = \frac{\|\{C_i \in \mathcal{H}_X \mid B \in C_i \wedge A \in C_i\}\|}{\|\mathcal{H}_X\|} = \frac{15}{100} = 0.15$.

Given these measurements, we can calculate the surprisal values for these measurements $I_X(\{A\}) = -log(0.55) \approx 0.86$ bits, $I_X(\{B\}) = -log(0.35) \approx 1.51$ bits, and, $I_X(\{A, B\}) = -log(0.15) \approx 2.74$ bits.

To estimate the probability of a multi-event measurement $C = \{d_1, d_2, \ldots, d_n\}$ in a context X in practice, we need to identify the occurrence probability of the combination of events in C occurring in context X. We do this by calculating the event combination's occurrence probability in the

[3] All logarithms are calculated with base 2, unless otherwise noted.

observation history database $\mathcal{H}_X$. This is a problem that has been extensively studied in the data mining literature in the context of *frequent itemset mining*. For example, the *Apriori* algorithm [1] constructs the set of frequent itemsets, i.e., combinations of items occurring more frequently than a given threshold value in a given input database and their occurrence counts. Also any other data mining algorithms for mining frequent itemsets could be used. Apriori requires as input a transaction database and a frequency threshold. It returns the set of frequent itemsets in the transaction database with regard to the frequency threshold and the occurrence counts of these frequent itemsets. We utilise this and use the Apriori algorithm to calculate the frequent event combinations observed in each context and their occurrence counts by invoking the algorithm on the context observation history database $\mathcal{H}_X$. We thus denote the set of frequent event combinations in context X for frequency threshold $f \in [0, 1]$ with $D_X(f)$, where $D_X(f) = Apriori(\mathcal{H}_X, f)$.

Given $D_X(f)$, we can calculate a lower bound for the surprisal of any measurement C using equation 7

$$P(O_X = C) = \begin{cases} \dfrac{C.count}{\|\mathcal{H}_X\|}, & \text{if } C \in D_X(f) \qquad (10) \\ f, & \text{otherwise.} \qquad (11) \end{cases}$$

where $C.count$ denotes the occurrence count of C in the history database $\mathcal{H}_X$ of context X.

This estimate is a lower bound, since for measurements not in the set of frequent event combinations, we do not have the exact occurrence count information (Apriori returns this information only for the frequent combinations). We only know that this occurrence frequency is smaller than f. Therefore, we take f as the upper bound for the occurrence probability of the measurement.

4.1.2 Evaluation

In order to evaluate the effectiveness of surprisal filtering against context guessing attacks, we determined separately for each user's context X the sets of frequently occurring Bluetooth and WiFi devices in the ConXPoP dataset using the Apriori algorithm as well as their occurrence frequencies. Using the patterns' occurrence frequencies, we calculated their corresponding surprisal values and filtered the classification results in the attack datasets by matching the measurements against the patterns and removing any such measurements whose matching pattern fell below the surprisal threshold. We then evaluated the impact of surprisal filtering on the False Positive and False Negative rates. The results are shown in Table 2.

As can be seen from the results, the filtering of PoPs based on their surprisal value reduces the FP rate of the attack scenarios by 52% to 60% for Bluetooth and 17% to 20% for WiFi, depending on the selected minimal surprisal threshold, thereby significantly reducing an attacker's odds for a successful context guessing attack.

Table 2: Improvement in FP rates when applying surprisal filtering on attack datasets

	Unfiltered	Improvement for $I_{thr} = n$ bits			
User	FP Rate	BT, n=2 bits	WiFi, n=2 bits	BT, n=4 bits	WiFi, n=4 bits
A	13.0%	-6.1%	-2.0%	-8.9%	-2.8%
B	37.8%	-27.2%	-5.4%	-31.1%	-5.6%
C	37.2%	-0.3%	-4.9%	-0.3%	-5.3%
D	21.4%	-17.4%	-0.0%	-19.4%	-0.0%
E	16.2%	-11.6%	-7.6%	-13.8%	-10.0%
F	40.5%	-23.8%	-7.7%	-26.8%	-9.5%
Avg	27.7%	-14.4%	-4.6%	-16.7%	-5.5%
Relative change		-52.0%	-16.6%	-60.4%	-20.0%

4.2 Longitudinal Ambient Modalities

While surprisal filtering effectively reduces false positives, our evaluation reveals that unfortunately, the False Negative (FN) rate also increases, especially in the Home contexts from 1.1% to 28.6% for WiFi-based and 55.0% for Bluetooth-based filtering on the average. This is understandable, since the device set-up in these contexts is usually quite static and the inherent entropy of the Bluetooth and WiFi environment therefore does not support the creation of effective PoPs based on these context modalities. We therefore introduce a complementary approach for handling PoPs in contexts that provide low surprisal in the context using the basic PoP schemes introduced above. In the following we show how longitudinal observations of ambient noise and luminosity can be used to construct PoPs that are hard to guess, even in contexts where the device set-up with respect to Bluetooth and WiFi devices is too static to provide valid proofs using the basic approach.

4.2.1 Ambient Light

Most smartphone devices today are equipped with a luminosity sensor, primarily used for adjusting the brightness of the smartphone's display in different lighting conditions. Sensor information about ambient luminosity is therefore readily available. The luminosity sensor does not consume much energy, making continuous tracking of ambient luminosity feasible.

Halevi et al. [5] investigated the use of ambient light for co-location verification for trusted end devices. However, they only consider the mean lighting level recorded by the devices during a short snapshot and conclude co-presence, if the average lighting levels do not deviate too much from one another. Such a simple scheme is obviously vulnerable to a malicious prover $\mathcal{A}$, who profiles the lighting conditions in a target context beforehand. For a context guessing attack, he just needs to replay the average lighting level in the target context as his fabricated measurement and he will succeed with high likelihood.

Therefore, we adopt a more sophisticated scheme. We do not consider only the average lighting level in the context, but focus on the relative changes in the lighting conditions over a slightly longer period of time, e.g., one minute. Our intuition is that such changes often arise from random events like human activity in the context and are therefore difficult to predict.

4.2.2 Ambient Noise Level

The use of ambient noise for the purpose of co-location verification has been investigated in the literature [5, 15]. Halevi et al. used time- and time-frequency-based similarity measures between two short 1-second audio snapshots, whereas Truong et al. [15] used similar measures but 10-second snaphots. In contrast to these approaches we take a slightly different approach and focus on the changes in the ambient noise level over a longer period of time, e.g., one minute. Here also, our intuition is that changes in ambient noise are likely to result from human activity (e.g., people talking) which for a malicous prover $\mathcal{A}$ is very difficult to predict.

For the purpose of context-based PoPs, we take the following approach: when the prover P requests a PoP, the verifier V and prover P record a snapshot $M = \{m_1, m_2, \ldots, m_n\}$, where the measurements m_i denote average readings of luminosity or ambient noise level, depending on the modality used, during subsequent time windows of width w. In practice, we propose to use windows of width $w = 1$ sec and to use snapshots of one minute, i.e., $n = 60$. We think that one minute is a long enough time period to capture enough changes in the context, while short enough to allow practical PoPs to be executed with modest delays. Since in practice many applications will execute PoPs asynchronosly and semi-automatically in the background, a delay of one minute should not present a big problem for user interaction. For example, an on-line social networking app might execute the PoP in the background after the user "checks in" in a particular location after arriving there, without requiring the user to wait for the completion of the protocol. Only in the case that the PoP fails, the user might get an error notification that the check-in failed.

Similar to the approach with WiFi and Bluetooth, we calculate the mean Hamming distance, the euclidean distance, and the mean exponential of difference (Features 2- 5 in Sect. 3) between the snapshots M_V and M_P of the verifier V and prover P.

We also introduce an additional feature for luminosity and audio measurements, the maximum cross-correlation between the measurement snapshots.

Feature 6 (maximum cross-correlation).

$$M_{corr}(M_V, M_P) = \max\{cross - correlation(M_V, M_P)\} \tag{12}$$

Since the placement of the sensors of a mobile device in a context plays a significant role on the intensity of the light and audio measurements these sensors pick up, two co-located devices might record measurements at significantly different signal levels. For our scheme this is, however, not a problem, since we are primarily interested in the changes in the context values and not the absolute readings as such. We therefore scale the context snapshots M_V and M_P by applying a min-max scaling so that all scaled measurements assume values between 0 and 100.

Similarly to the approach taken in Section 3, we use the obtained distance measures calculated from the benign datset to train a classification model for co-location for making predictions about whether a prover P's context measurement is co-located with the verifier V or not.

4.2.3 Evaluation Results

To evaluate the effectiveness of using longitudinal measurements of luminosity and audio as context modalities for PoP, we added features 2 - 6 calulated based on luminosity and audio to the WiFi and Bluetooth-based features we used for the co-location classifier in the basic PoP scheme. We then evaluated the effectiveness of different feature combinations on the benign and attack datasets. Table 3 shows the results for the ConXPoP dataset (The ZIA dataset did not contain luminosity nor audio measurements).

Table 3: PoPs utilizing audio and luminosity modalities

Classifier features	FP rate	FN rate
Benign dataset		
Luminosity	20.1%	14.3%
Audio	19.2%	16.0%
Luminosity+Audio	9.3%	9.2%
BT	16.1%	9.8%
WiFi	11.0%	9.9%
BT + WiFi	9.3%	6.4%
Luminosity+Audio+BT+WiFi	4.2%	2.4%
Attack dataset		
Luminosity	1.1%	0.0%
Audio	0.4%	0.0%
Luminosity+Audio	0.4%	0.0%
BT	21.9%	0.0%
WiFi	26.0%	0.0%
BT + WiFi	23.5%	0.0%
Luminosity+Audio+BT+WiFi	3.6%	0.0%

As can be seen, adding audio and luminosity as PoP features significantly decreases the False Positive rates for both the benign and attack datasets' classification results. Especially for the attack dataset, the luminosity and audio context modalities clearly outperform PoPs based on WiFi and Bluetooth. This significantly impacts the attacker's ability to succeed in context guessing attacks.

5. DISCUSSION

Our results show that in scenarios in which the prover can not be trusted, context guessing attacks pose a serious problem for contextual proofs-of-presence for some context modalities like WiFi or Bluetooth observations. However, by profiling the user's contexts and using the surprisal of a contextual PoP as a filtering criterion, we can to some degree mitigate this threat. The impact of the countermeasure on the acceptance of benign PoPs is, however dependent on the type of context. In contexts with only little dynamic context information (e.g., a person's home) it is challenging to conduct valid PoPs with a sufficient surprisal. However, many contexts that are relevant for our usage scenarios are typically public in nature (e.g., restaurants or shops) and contain significant amounts of dynamic context information. The lack of surprisal in some context modalities can be encountered by extending PoPs to further ambient context modalities providing more entropy. As we showed in Sect. 4.2, the addition of luminosity and audio to the PoP modalities provide good performance against context guessing attacks.

It seems therefore likely that constructing PoPs with sufficient surprisal in most contexts is feasible. In our future work we intend to investigate this issue further. In situations in which ambient context entropy is not sufficient (e.g., in a dark and silent room during the night), PoPs can be still feasible by combining the context-based and beaconing-based PoP approaches. Our currently ongoing research regarding the use of ambient context sensor modalities indicate, e.g., that beaconing-based PoPs using the visible light channel are feasible. They require, however active user involvement, which limits the applicability to such use cases, in which the user is actively involved, e.g., making a location chek-in.

In contrast to other earlier works utilizing audio measurements for co-location proofs [5, 15], our approach has considerable privacy advantages since the PoP utilizes ambient noise level and not the actual fine-grained audio signals. Therefore, the prover P does not need to transmit potentially sensitive audio recordings to the verifier V in order to obtain a proof-of-presence. This is important especially in the peer-to-peer scenario, in which all users can assume both the role of a prover P and a verifier V.

5.1 Limitations

Relay attacks pose a fundamental problem for proof-of-presence schemes, and to the best of our knowledge, only distance-bounding based techniques (cf., e.g., [7]) are able to provide an effective protection against such attacks. However, the drawback of distance bounding is, that it requires special high-accuracy hardware that is typically not available on regular mobile devices.

For our application scenarios, relay attacks would not seem to pose a major problem for economic reasons. For instance, in the peer-to-peer scenario, it would be prohibitively complex and costly for a malicious prover to place an accomplice in all possible contexts that a target node visits. In LBS scenarios it might be conceivable that some malicious clients could be motivated to stage targeted relay attacks against selected venues. However, also here the usage of several different contextual modalities for PoPs significantly raises the complexity and cost of the attack for a potential attacker and especially his possible accomplices. A simple replaying of PoP protocol messages by the attacker's accomplice would not be sufficient, but the accomplice would need to actively participate in sensing the context of the verifier in several different context modalities.

6. RELATED WORK

Closely related to our work are the papers by Truong et al. [15] and Shrestha et al. [14]. They use direct measurements of elements of the ambient context for determining the co-presence of two devices in a zero-interaction authentication scenario. However, they assume both endpoints of the scenario to be trusted. The context guessing attack is therefore not applicable to their scenario.

The concept of using context-profiling for evaluating contexts for security enforcement has been discussed by Gupta et al. [4]. Their work focuses on estimating the threat level in a particular context for the purpose of making access control decisions. Our work, takes a different viewpoint: we estimate the occurrence probability of a particular context measurement in view of the observation history, in order to estimate the threat of a guessing attack.

6.1 Beaconing-based Proofs of Presence

Saroiu and Wolman [12] hypothesize six different LBS-based scenarios, in which users of the LBS might have an incentive to engage in location cheating. To tackle such scenarios, they propose a simple protocol for providing location proofs based on beaconing of information over the WiFi SSID of dedicated access points (APs) installed at the target venue. The proof of presence is based on the fact that only devices in the access point's proximity will be able to receive these beacon signals. Our solution, however, is not dependent on dedicated APs.

Another approach based on beaconing of information into the context is the SMILE framework of Manweiler et al. [9], which allows users to establish proofs of co-location after an encounter that took place between the users. It is based on users' devices beaconing cryptographic keys into the proximity of their device and recording keys beaconed by other devices. Later the devices are able to rediscover each other with the help of a third-party server. Contrary to our approach, SMILE requires the use of a central server and requires all devices to engage simultaneously in beaconing and scanning of the context, potentially impacting the privacy of users by making their devices traceable across different contexts. Carbunar et al. [3] present a scheme for privacy-preserving Geo-Social Network logins. They utilize mix networks and a protocol involving blind signatures to provide *GeoBadges*, i.e., anonymous proofs of repeated visits to a specific venue. Their system relies on dedicated hardware at the venues, like display changing QR codes used for location verification. Polakis et al. [11] present a similar scheme for location proofs, which relies on the use of temporary codes which a location-based service can verify. These codes are transmitted over NFC to client devices. The use of NFC as a close proximity protocol thus acts as the proof of co-location.

6.2 Context-based Proofs of Presence

Varshavsky et al. [16] describe a system for co-location verification. They combine Diffie-Hellman key agreement with profiling of WiFi packets for verification of co-location. They compare the received signal strengths of the received packets on a WiFi network in common for both parties. If these are similar enough, the peers are determined to be co-located. According to their paper, the protection of this scheme arises from the fact that fluctuations in the RF environment are unpredictable and spatially limited. Devices located close to each other will be able to observe such fluctuations, whereas devices that are farther away from each other will not be able to do so. However, due to this same property, the prover and verifier need to be located relatively close to each other in order for their approach to work, limiting its practical applicability. Most real-world scenarios, in which peers are in the same room (e.g., Alice and Bob at the same restaurant), but not in immediate proximity would not be feasible using their approach.

Narayanan et al. [10] present three alternative asymmetric protocols for principals to test for proximity in a privacy-preserving manner. Their solutions are based on Private Equality Testing and Private Threshold Set Intersection. They also utilise *location tags* obtained by the principals from ambient information in the context. They discuss location tags derived from WiFi broadcast packets, WiFi access point IDs, Bluetooth devices, GPS signals, GSM radio features, audio fingerprinting, and, even atmospheric gases, but present practical analysis only for the WiFi broadcast packet-based solution. They estimated that using the address fields, the packet sequence numbers and packet payload, one could obtain roughly 10 bits of entropy from each broadcast protocol. Their approach, however has some practical limitations, which they also acknowledge. Firstly, the prover and verifier need to agree on using the same WiFi access point and both be able to connect to it. Therefore, the method is not applicable in situations in which no access points are available, or, access to the AP is password protected. The ability to generate location tags is also heavily dependent on the traffic patterns of the WiFi access points.

On more low-traffic networks like residential private access points, acquiring a sufficient number of packets during a reasonable time frame might actually be challenging. The work by Varshavsky et al. [16] suffers from similar limitations. Also, in some jurisdictions, it is legally prohibited to intercept packets from foreign networks without proper authorisation or explicit permission of the network's operator.

6.3 Distance-Bounding Based Approaches

Hu et al. [7] investigated the problem of proximity verification in the context of mobile ad-hoc networks as a defence against *wormhole attacks*. They proposed to use a *distance bounding* approach in order to verify an upper limit on the distance to a node in the network. The distance bounding approach, however, requires the ability to make timing measurements with a very high accuracy and is usually not possible without special hardware. Distance bounding is therefore usually not feasible on regular mobile devices. Also Polakis et al. [11] and Carbunar et al. [2] proposed the use of distance-bounding in their schemes in order to protect against relay attacks. This attack is feasible for our scenario, but not very relevant, since the attacker would need to instrument all target contexts with a relaying node. Given the vast amount of different contexts that an attacker would want to target, this would be clearly uneconomical for the vast majority of potential attackers.

7. SUMMARY

We show that *context-guessing attacks* can impact context-based proofs-of-presence in scenarios where a verifier cannot fully trust the prover. The feasibility of such attacks is shown on traces of Bluetooth and Wifi mobile data. To alleviate context guessing, a methodology based on the *surprisal* related to context measurements is designed and formalized. The effectiveness of this mitigation methodology is demonstrated on measurements collected from mobile phones. Our work also shows that in case there is insufficient entropy to encounter the context guessing, such attacks can be further thwarted using by adding ambient context modalities to the PoP which is experimented using measurements of ambient luminosity and noise levels.

8. REFERENCES

[1] Rakesh Agrawal, Heikki Mannila, Ramakrishnan Srikant, Hannu Toivonen, A Inkeri Verkamo, et al. Fast discovery of association rules. *Advances in knowledge discovery and data mining*, 12(1):307–328, 1996.

[2] B. Carbunar and R. Potharaju. You unlocked the mt. everest badge on foursquare! countering location fraud in geosocial networks. In *Mobile Adhoc and Sensor Systems (MASS), 2012 IEEE 9th International Conference on*, pages 182–190, 2012.

[3] Bogdan Carbunar, Radu Sion, Rahul Potharaju, and Moussa Ehsan. The shy mayor: Private badges in geosocial networks. In Feng Bao, Pierangela Samarati, and Jianying Zhou, editors, *Applied Cryptography and Network Security*, volume 7341 of *Lecture Notes in Computer Science*, pages 436–454. Springer Berlin Heidelberg, 2012.

[4] Aditi Gupta, Markus Miettinen, N. Asokan, and Marcin Nagy. Intuitive security policy configuration in mobile devices using context profiling. In *International Conference on Privacy, Security, Risk and Trust (PASSAT), and 2012 International Confernece on Social Computing (SocialCom)*, pages 471–480, September 2012.

[5] Tzipora Halevi, Di Ma, Nitesh Saxena, and Tuo Xiang. Secure proximity detection for nfc devices based on ambient sensor data. In Sara Foresti, Moti Yung, and Fabio Martinelli, editors, *Computer Security ESORICS 2012*, volume 7459 of *Lecture Notes in Computer Science*, pages 379–396. Springer Berlin Heidelberg, 2012.

[6] Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11(1):10–18, November 2009.

[7] Yih-Chun Hu, A. Perrig, and D.B. Johnson. Packet leashes: a defense against wormhole attacks in wireless networks. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 3, pages 1976–1986 vol.3, 2003.

[8] Foursquare Labs Inc. foursquare.com. (online, last referenced [2014-07-24]).

[9] Justin Manweiler, Ryan Scudellari, and Landon P. Cox. Smile: Encounter-based trust for mobile social services. In *Proceedings of the 16th ACM Conference on Computer and Communications Security*, CCS '09, pages 246–255, New York, NY, USA, 2009. ACM.

[10] Arvind Narayanan, Narendran Thiagarajan, Mugdha Lakhani, Michael Hamburg, and Dan Boneh. Location privacy via private proximity testing. In *NDSS*, 2011.

[11] Iasonas Polakis, Stamatis Volanis, Elias Athanasopoulos, and Evangelos P. Markatos. The man who was there: Validating check-ins in location-based services. In *Proceedings of the 29th Annual Computer Security Applications Conference*, ACSAC '13, pages 19–28, New York, NY, USA, 2013. ACM.

[12] Stefan Saroiu and Alec Wolman. Enabling new mobile applications with location proofs. In *Proceedings of the 10th Workshop on Mobile Computing Systems and Applications*, HotMobile '09, pages 3:1–3:6, New York, NY, USA, 2009. ACM.

[13] D. Schürmann and S. Sigg. Secure communication based on ambient audio. *Mobile Computing, IEEE Transactions on*, 12(2):358–370, Feb 2013.

[14] Babins Shrestha, Nitesh Saxena, Hien Thi Thu Truong, and N Asokan. Drone to the rescue: Relay-resilient authentication using ambient multi-sensing. In *Proc. Eighteenth International Conference on Financial Cryptography and Data Security*, 2014.

[15] Hien Thi Thu Truong, Xiang Gao, Babins Shrestha, Nitesh Saxena, N Asokan, and Petteri Nurmi. Comparing and fusing different sensor modalities for relay attack resistance in zero-interaction authentication. In *IEEE International Conference on Pervasive Computing and Communications, PerCom*, 2014.

[16] Alex Varshavsky, Adin Scannell, Anthony LaMarca, and Eyal Lara. Amigo: Proximity-based authentication of mobile devices. In John Krumm,

GregoryD. Abowd, Aruna Seneviratne, and Thomas Strang, editors, *UbiComp 2007: Ubiquitous Computing*, volume 4717 of *Lecture Notes in Computer Science*, pages 253–270. Springer Berlin Heidelberg, 2007.