Albayram *et al. Human-centric Computing and Information Sciences* ⊠⊠⊠⊠⊠⊠⊠⊠⊠⊠
DOI 10.1186/s13673-015-0032-3

Human-centric Computing and Information Sciences
a SpringerOpen Journal

RESEARCH

Open Access

Designing challenge questions for location-based authentication systems: a real-life study

Yusuf Albayram[1]*, Mohammad Maifi Hasan Khan[1], Athanasios Bamis[2], Sotirios Kentros[3], Nhan Nguyen[1] and Ruhua Jiang[1]

*Correspondence:
yusuf.albayram@uconn.edu
[1] Department of Computer Science and Engineering, University of Connecticut, CT, Storrs, USA
Full list of author information is available at the end of the article

Abstract

Online service providers often use challenge questions (a.k.a. knowledge-based authentication) to facilitate resetting of passwords or to provide an extra layer of security for authentication. While prior schemes explored both static and dynamic challenge questions to improve security, they do not systematically investigate the problem of designing challenge questions and its effect on user recall performance. Interestingly, as answering different styles of questions may require different amount of cognitive effort and evoke different reactions among users, we argue that the style of challenge questions itself can have a significant effect on user recall performance and usability of such systems. To address this void and investigate the effect of question types on user performance, this paper explores location-based challenge question generation schemes where different types of questions are generated based on users' locations tracked by smartphones and presented to users. For evaluation, we deployed our location tracking application on users' smartphones and conducted two real-life studies using four different kinds of challenge questions. Each study was approximately 30 days long and had 14 and 15 users respectively. Our findings suggest that the question type can have a significant effect on user performance. Finally, as individual users may vary in terms of performance and recall rate, we investigate and present a Bayesian classifier based authentication algorithm that can authenticate legitimate users with high accuracy by leveraging individual response patterns while reducing the success rate of adversaries.

Keywords: Authentication; Usability; Security; Location based challenge questions; Smartphones; Android

Introduction

Various forms of challenge questions are often used as a form of backup authentication mechanism (a.k.a. fallback authentication) to facilitate resetting of passwords or as an extra layer of security if service providers are suspicious of malicious activities [1]. This approach is more commonly known as knowledge-based authentication (KBA) scheme [2, 3]. Current KBA schemes can be further divided into two categories, namely, static KBA and dynamic KBA. In static KBA, a user usually selects a number of predefined personal questions at the time of registration/account creation such as "What is the name of your first pet?" or "What is the name of the city where you grew up?". However, the use of pre-agreed personal challenge questions has been widely criticized and considered to be a weak form of authentication as the answers are often easy to guess [1, 4–7]. To address the limitations of static KBA schemes, recently, dynamic KBA schemes are being investigated where questions and answers are generated on the fly leveraging user's activity logs such as past financial activity, shopping behavior, browsing history, emails, and smartphone usage patterns [8–12].

While prior dynamic KBA based schemes leverage dynamicity to improve security and demonstrated promise, prior efforts do not systematically investigate the problem of designing challenge questions leveraging user's spatio-temporal behavior data (e.g., location traces). Interestingly, as different users differ in terms of recall ability, we argue that, the style of challenge questions itself can have a significant effect on user performance (e.g., accuracy) and usability of such systems, and needs to be systematically studied in real-life.

To address this void and complement existing dynamic KBA schemes, in this paper, we investigate the problem of designing challenge questions and measure the effect of different styles of challenge questions on user performance. Briefly, in our work, we leverage users' location information tracked by smartphones over an extended period to generate challenge questions and authenticate users. To be able to track people inside buildings and minimize the energy overhead, in our system, a tracking application is installed on users' smartphones that collects Wi-Fi access point (AP) information periodically in the background, and subsequently uses Wi-Fi fingerprinting techniques to approximate the locations of a user based on Wi-Fi AP data.

Once a user's locations are identified, the algorithm calculates an "interestingness" weight for each location based on statistical measure of randomness in the user's spatiotemporal data, and subsequently picks locations for generating challenge questions that have the highest weight (i.e., rare locations have higher weights). The goal of this weight is to give preference to more infrequent events/locations (e.g., visiting a new restaurant) which are more likely to be remembered by a legitimate user, and harder to guess by an attacker.

To evaluate the effect of different question styles on user performance, we designed two separate studies as follows. During the course of the first study, we asked three different kinds of questions. Type - 1 question asks users to identify the locations visited at a specific time. Type - 2 question asks users to identify the locations and the order of visits for a given day. Finally, Type - 3 question asks users to identify the locations and the time window when the places are visited for a given day. In each case, possible answers to challenge questions were presented in the form of multiple choices. To simplify the authentication process and improve the usability of the system, possible answers to each authentication question are presented to the user in visual form by mapping each place to a location on a map generated using Google Maps Street View service (see Fig. 3).

In the second study, challenge question asks where the user was at a certain time. An example of this type of question appears in Fig. 4. Unlike the first study, where possible answers are presented in the form of multiple choices, in this case, users are presented with a map and requested to select locations that he/she has visited during a certain time window of a specific day.

As one of the key goals of our study is to measure the strength of location-based authentication schemes against different kinds of adversaries, in our study, we classify adversaries into two broad categories with increasing capabilities. Specifically, in our study, a naive adversary is someone who is trying to break into the system with limited or no knowledge regarding a user. For example, a naive adversary may know (or guess) the state/city of residence of the person being attacked but may not know the details regarding a target user's daily travel patterns. In contrast, a strong adversary is someone who has significant knowledge regarding a user's daily schedules (e.g., co-workers, close friends, spouse).

For evaluation, to simulate strong adversarial users, we recruited volunteers in pairs (e.g., close friends, married couples). We had 6 pairs and 2 individual volunteers in Study 1, and 6 pairs and 3 individual volunteers in Study 2. During the first study, users answered a total of 1833 questions (i.e., 611 of Type - 1, 611 of Type - 2, and 611 of Type - 3). During the second study, users answered a total of 2423 questions (i.e., Type - 4). Our findings suggest that question styles can have a significant impact on both user and adversary performance. In addition, our study reports higher accuracy compared to prior efforts for legitimate users, and lower accuracy for naive adversaries. Our key findings are presented in Section "Evaluation" in more details.

To summarize, this paper makes the following key contributions:

- To investigate the effect of question types on user performance, this paper conducted two real-life studies where different types of challenge questions are generated based on users' locations tracked by smartphones and presented to users.
- To evaluate the strengths and weaknesses of our presented schemes against various attackers in real-life (e.g., strong, naive), we recruited users in pairs (e.g., close friends) to simulate naive and strong adversaries.
- To account for the variability in user's recall ability, a Bayesian classifier based algorithm is developed to authenticate legitimate users with high accuracy while reducing the success rate of adversaries.
- Finally, in-depth analyses of key findings along with limitations of our work and possible future directions are presented in the paper.

The rest of the paper is organized as follows. Section "Related work" presents a summary of prior work. Section "Overview" presents the overview of the system along with the addressed research challenges in this work. Section "Evaluation" presents the results from our experiments with real users. The limitations and future directions of our work is discussed in Section "Discussion". Finally, Section "Conclusion" concludes the paper.

Related work

Various forms of backup authentication mechanisms are being investigated by prior efforts to facilitate resetting of passwords or provide an extra layer of security for authentication if service providers are suspicious of malicious activities [1]. The most widely known backup authentication mechanism is based on challenge questions where a service (e.g., website) requires a user to answer personal verification questions, which is commonly referred to as knowledge-based authentication (KBA) in the literature [2, 3]. KBA can be further divided into two categories, namely, static KBA and dynamic KBA.

In static KBA, the questions are often generated based on personal information of a user such as “What is the name of your first pet?” or “What is the name of the city where you grew up?”. In the absence of a password, the challenge questions selected previously are used to authenticate users by comparing provided answers against the saved answers. However, the use of pre-agreed personal authentication questions has been widely criticized and considered to be a weak form of authentication method [1, 4–7] due to various vulnerabilities. For example, Zviran and Haga [13] found that participants were able to remember 78 % of their answers, and those who are close to the participants (e.g., spouses, close friends) were able to guess the answers correctly 33 % of the time. A similar study conducted by Podd et al. in 1996 [14] reported similar recall rate (80 %) for legitimate users and higher success rate by attackers (39.5 %). Both studies reported that participants forget 20 %-22 % of their answers within three months. More recently, Schechter et al. [6] performed a user study (N = 130) of the challenge questions used by four large webmail providers and they pointed out that 20 % of the participants forget their own answers within six months. 17 % of their answers was guessed correctly by acquaintances with whom participants were unwilling to share their webmail passwords. 13 % of the answers could be guessed within five attempts by guessing the most popular answers of other participants. Furthermore, Rabkin [5] identified that security questions are getting weaker due to improved information retrieval techniques and increase in online content. By mining online sources (e.g., social networking sites or public records), an attacker can obtain the details about one’s personal information to answer many of the challenge questions commonly used for backup authentication. For instance, the answer to the question “What year did you graduate from college?” may be found from one’s Facebook profile or LinkedIn profile. Moreover, as many of the challenge questions are often used across different websites, the consequences of compromising a single account can be overwhelming.

To address the limitations of static KBA schemes, dynamic KBA schemes are being investigated where challenge questions are not predetermined. Instead, questions and answers are generated on the fly based on user’s activities such as past financial activity, shopping behavior etc. For example, several major American credit bureaus authenticate users by generating questions about past financial transactions. Several variants of this dynamic form of knowledge based authentication technique have been proposed to avoid the drawbacks of static KBA challenge questions. For example, Asgharpour and Jakobsson introduced a technique where challenge questions are generated leveraging a user’s browsing history in a recent period [8]. Nosseir et al. [9] proposed a method that uses electronic personal history (e.g. personal calendar data) to generate challenge questions in order to authenticate users. Nishigaki and Koike [10] proposed to use user’s email history to generate challenge questions. Nosseir and Terzis [15] studied the effect of image based representation of questions in an authentication system that uses electronic personal history. In another similar approach, Terzis et al. [11] introduced a question based authentication scheme that generates challenge questions based on a user’s behavior (context) that occur within a particular smart environment. The proposed system, however, is limited in generating simple authentication questions based on the arrival and departure time of different people from a smart space. More recently, Ullah et al. [16] developed a Profile Based Authentication Framework (PBAF) where challenge questions are generated based on a student’s profile information (e.g., month started the current course) that are subsequently used to authenticate students during online examinations. However, the set of questions are relatively static as the profile information per student does not change overtime. Gupta et al. [17] investigated the memorability of various users' smartphone usage behavior (e.g., emails, calendar events, calls etc.) and attempted to leverage that to authenticate users. One of the main limitations of this work is that the challenge questions are generated based on a user's routine (e.g., who do you call the most?) rather than day-to-day activities which are more dynamic and is the focus of our work. Similarly, another work [18] used email activities to generate challenge questions (e.g., who sent you the most emails?). While Choi et al. [19] presented an authentication method by utilizing users' geo-temporal history to generate questions, this work does not provide any quantitative evaluation regarding user performance and also does it evaluate the system against various attackers in real-life. Das et al. [12] presented an authentication framework that exploits a user's recent phone usage, communications, and location traces to generate challenge questions, which is the closest in spirit to our work.

While several dynamic KBA schemes are being investigated by prior efforts, the focus of our work differs from prior efforts in one key aspect. Specifically, our work investigates the effect of various forms of questions that may be asked leveraging the location data on user and adversary's performance. In particular, we generate different types of questions, namely, location only, location and order of visits, and location and time of visits. As these questions evoke different kinds of memory, we expect significant difference in terms of accuracy in answering them. Also, prior work does not evaluate the strength of location-based scheme against various attackers in real-life, which is done in our work. The details of our work are presented in the following sections.

Overview

To evaluate the effect of different challenge questions on user performance in real-life, we implemented a location-based authentication framework that has four main components, namely, the *location tracking component*, the *location extraction component*, the *question and answer generation component*, and finally, the *authentication component*. Briefly, the *location tracking component* is responsible for tracking user's locations by running an application on user's Android phone continuously in the background. The *location extraction component* is responsible for extracting physical locations from location traces collected by the application. The *question and answer generation component* is responsible for (i) automatically identifying "important" locations of a user and using that information to generate authentication challenges, and (ii) generating plausible fake answers for each set of multiple choice question. Finally, the *authentication component* is responsible for authenticating users based on personalized models that are created using users' historical accuracy and response patterns. The details of each of these components along with the associated challenges are described below.

The location tracking component

For location-based authentication, one of the key challenges is to track user locations accurately with minimal energy overhead. As cell-phone tower association data only provides coarse-grained locations and GPS is ineffective for indoor localization, in this work, we choose to use Wi-Fi fingerprinting techniques for tracking user's locations. To do so, we develop a mobile client application for Android based smartphones that periodically (currently every 2 minutes) records all the Wi-Fi AP beacons in its range. From these beacons, the application composes Wi-Fi fingerprint records along with the relevant temporal information (i.e., time stamp), and stores them temporarily in a local cache in encrypted form. In our study, a Wi-Fi fingerprint record includes BSSIDs (MAC address of APs), and RSSIs (signal strength of received Wi-Fi signals) of the nearby APs.

Energy Overhead: To measure the energy requirement of the location tracking application, we measured the power consumption of our app on a Nexus S phone using GSam Battery Monitor application [20]. For accuracy, we repeated the measurement multiple times. The result indicated that our location tracking application consumed approximately 0.1 % of the total power used by the smartphone over a period of 24 hours.

The location extraction component

Once the Wi-Fi AP information is collected, the next challenge is to identify users' locations from Wi-Fi AP as accurately as possible. However, while developing and testing our prototype, we discovered that the APs in each area tend to change over time. This is particularly true for densely populated areas such as cities, university campus or apartment complexes. Also, in many cases, the location tracking application may recognize and report multiple AP information at a particular time. To avoid considering each AP as a unique physical location, in our work, we use a clustering algorithm that groups APs based on Wi-Fi signal strengths and use the clustering to infer user's locations. However, as a user may visit new places over time and we do not know a priori the total number of places a user may visit, we chose to employ a density-based clustering approach that can incrementally adapts the number of clusters (i.e., the number of unique physical locations) in our system. More specifically, we use the DBSCAN (Density-Based Spatial Clustering of Applications with Noise) algorithm [21] that is based on the notion of density reachability. Briefly, in our algorithm, a point p is density-reachable from a point q if their distance is less than ϵ (*eps*) or if there is a path of pair-wise density-reachable points between them. In practice, the DBSCAN algorithm starts by assigning a random point in a cluster and expands it with neighborhoods of at least min_{pts} points that are within a distance ϵ from it. In our work, to calculate the distance between two Wi-Fi records, we develop a new metric that is similar to the Jaccard similarity coefficient. In our case, a measurement M reported by the location tracking application consists of the following: $\mathrm{M} = \{A, S_A, \mathrm{timestamp}\}$, where A is the set of observed AP MAC addresses and S_A are the signal strengths for these APs. The elements in S_A are indexed by the elements in A. We obtain the weights for each AP from the signal strength measurements as follows.

$$W_A[m] = \begin{cases} 0.5 & , \text{if } \ S_A[m] > -50 \\ \frac{150+S_A[m]}{200} & , \text{if } \ -99 \leq S_A[m] \leq -50. \end{cases}$$

Here, $W_A[m]$ is the weight for the AP address m in A, and the signal strength is measured in dBm. In order to give priority to APs that are closer to the user, we give higher weights to APs with higher signal strength. Once the weight for each AP is calculated, the similarity between the two measurements M_A and M_B is computed as follows:

$$S(M_A, M_B) = \frac{\sum_{m \in A \cap B} W_A[m] + W_B[m]}{min(|A|, |B|)} \tag{1}$$

Here, A and B are the sets of observed AP addresses in the two measurements M_A and M_B, and W_A and W_B are the weights for the AP addresses in M_A and M_B respectively.

As $W_A[m] \in [0.25, 0.5]$, $W_A[m] + W_B[m] \in [0.5, 1]$. Hence, APs that are common in M_A and M_B will get a weight between 0.5 and 1, whereas APs that are not common will be penalized by receiving a weight of 0.

We use ϵ (*eps*) as a threshold to determine if two measurements M_A and M_B are in "close" proximity or not based on their similarity score $S(M_A, M_B)$ as defined in equation 1.

Based on the above similarity measure and the threshold ϵ, the DBSCAN algorithm either creates new clusters for the measurements (i.e, when the similarity measurement is less than *eps*) or expands/updates the clusters as new measurements are reported by the location tracking application.

Each cluster represents a physical location in the system and is represented by the MAC addresses that appear in the cluster, and this set of MAC addresses becomes the "fingerprint" for the specific physical location. An illustrative example of the clustering algorithm is depicted in Fig. 1 where APs are grouped in two clusters that represent two physical locations.

To evaluate the performance of our clustering scheme, we ran experiments in multiple buildings at UConn campus. We observed that the DBSCAN algorithm can achieve floor-level accuracy within the buildings. For example in Fig. 2, cluster-18 (on the second floor) and cluster-19 (on the third floor) in the ITE building represents two different locations in the same building.

As new measurements arrive, new points are first examined to determine whether they can be assigned to any of the existing clusters. If not, the new points are given as input to the DBSCAN algorithm to regenerate the clusters for the new locations.

Finally, once the Wi-Fi APs are clustered, we utilize the Google Maps Geolocation API [22] in order to map the Wi-Fi fingerprints to specific physical locations. More specifically, the MAC addresses of the Wi-Fi APs along with the corresponding RSSI values are used to query the Google Location Service to retrieve the corresponding GPS coordinates and accuracy radius for the requested Wi-Fi fingerprints. Subsequently, the GPS coordinates are used to show the location on Google Maps and retrieve street view images of the corresponding places. Once the locations are identified, the *question and answer generation component* generates the challenge questions and possible answers as follows.

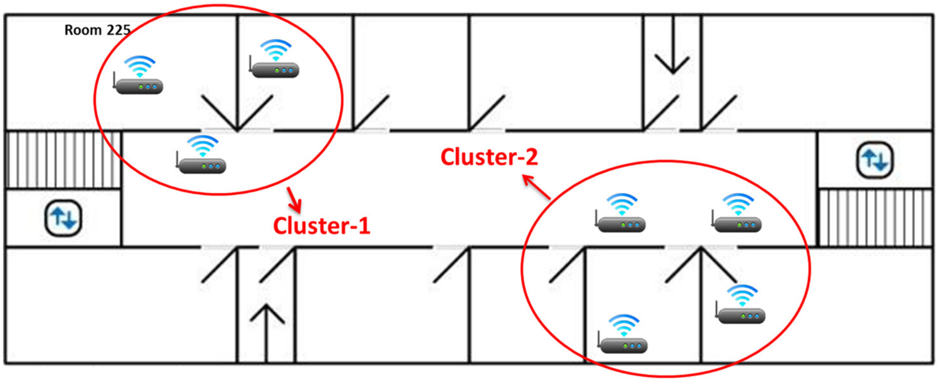


Fig. 1 Nearby Access Points are clustered together and represents a single physical location collectively

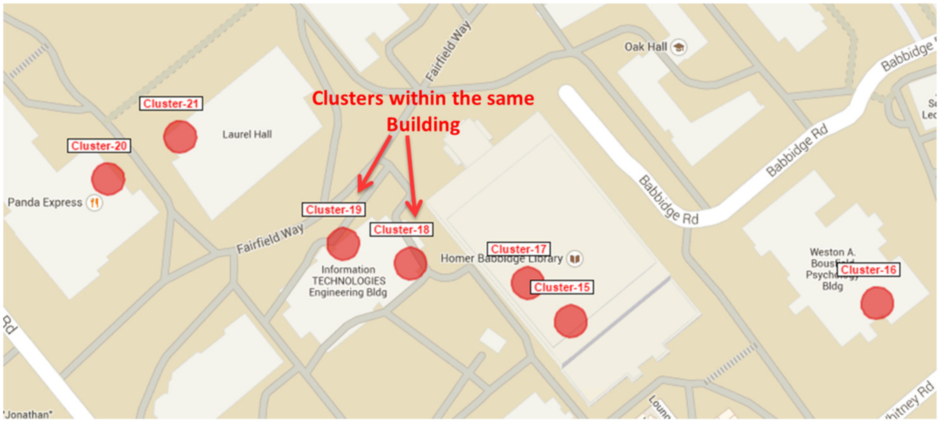


Fig. 2 Each circle represents a cluster generated by the DBSCAN algorithm based on Wi-Fi AP data. Please note that there can be more than one cluster within a building

The question and answer generation component

Once the system identifies the various locations of a user, the next challenge is to generate challenge questions automatically. However, as different locations are visited at different frequencies (e.g., doctor's office vs. grocery store), certain locations are more likely to be remembered by users than some other locations. Hence, an algorithm is needed to identify the "interesting" locations that are more appropriate for generating challenge questions. Another key challenge is to balance the difficulty level of the questions and the usability of the system. For instance, questions must be generated and presented in a way that facilitates recall for legitimate users. At the same time, it should be as difficult as possible for an adversary to guess the answers correctly due to his/her partial knowledge of a user's schedule. To address the aforementioned challenges, for a given location history of a user, the system generates questions leveraging location traces in multiple steps as follows.

Let H be a user's location history that consists of a sequence of spatio-temporal events of a user. The history H is sequences of events generated by the motion of the user over time and space. Each such event has the form $h_i = (l_i, t_i)$, where l_i represents a uniquely identifiable location and t_i is a time-stamp. Essentially, H is a sequence of location-time pairs that reveal the presence of the user at specific locations during specific times. Given the set of locations $L = \{l_0, \ldots, l_i\}$ that a user has visited in the past, we generate the history H by splitting each day into a set $W = \{w_0, \ldots, w_m\}$ of time windows of fixed size (e.g., 15 minutes). For each such time window w_m in a day, a location l_i is assigned to it from the set $L \cup \{u\}$, where l_i represents the Wi-Fi AP fingerprint based on MAC addresses that appear in that time window more than a specified threshold (e.g., 75 %). The "unknown" location u is used to represent movement between locations or cases where no measurement was recorded due to reasons such as absence of Wi-Fi. The history H of measurements is thus converted into matrices of time windows per day, where each matrix cell contains a location, as it can be seen in Table 1.

Once locations are assigned for each time window, the system computes an "interestingness" weight for each location based on statistical measure of randomness in a user's spatio-temporal behavior. More specifically, the system attempts to pick locations

Table 1 Time Window-Location Matrix of a user's history

Window\Day	Nov 14	Nov 15	. . .	Nov 21
00 : 00 − 00 : 14	*Home*	*Home*	. . .	*Bronx*
00 : 15 − 00 : 29	*Home*	*Home*	. . .	*Bronx*
⋮	⋮	⋮	⋮⋮⋮	⋮
14 : 00 − 14 : 14	*Office*	*Lab*	. . .	*Bronx*
14 : 15 − 14 : 29	*Unknown*	*Lab*	. . .	*Bronx*
14 : 30 − 14 : 44	*Lab*	*Lab*	. . .	*Unknown*
⋮	⋮	⋮	⋮⋮⋮	⋮
18 : 00 − 18 : 14	*Supermarket*	*Home*	. . .	*IppudoNY*
18 : 15 − 18 : 29	*Supermarket*	*Home*	. . .	*IppudoNY*
18 : 30 − 18 : 44	*Unknown*	*Home*	. . .	*Unknown*
18 : 45 − 18 : 59	*Home*	*Home*	...	*Cinema*
⋮	⋮	⋮	⋮⋮⋮	⋮
21 : 00 − 21 : 14	*Friend′sHome*	*Restaurant*	. . .	*RidgeHill*
21 : 15 − 21 : 29	*Friend′sHome*	*Restaurant*	...	*RidgeHill*
⋮	⋮	⋮	⋮⋮⋮	⋮
23 : 45 − 23 : 59	*Home*	*Home*	...	*RidgeHill*

for generating questions that have the highest weight based on a user's spatio-temporal behavior. However, as different users may show different degree of randomness in their spatio-temporal behaviors (i.e., some users may have less predictable location-patterns than others), using a certain weight threshold to pick questions would not be a good idea. Hence, in our work, the questions for each user are chosen based on a weight that gives preference to more infrequent events in a user's schedule. To do so, the system analyzes daily and weekly location-patterns of a user. For instance, a user may spend weekdays at work during specific hours (e.g. between 8am and 5pm) and may spend weekends doing shopping and/or visiting restaurants and new places. Thus, finding these infrequent events (i.e., locations) is crucial to generate challenge questions which are more likely to be remembered by a user, yet harder for an attacker to guess compared to routine events. Hence, to identify the infrequent events/places, for a given *Time Window-Location Matrix* (e.g., as shown in Table 1), weights of locations are computed as follows.

1. Extract the distinct locations and locate them in a list L_i where $L_i \subseteq (L \cup \{u\})$.
2. Calculate $P(l_i)$ which denotes the probability of being at location l_i.
3. Calculate $P(l_i|w_m)$ which denotes the probability of being at location l_i during time interval w_m. For example, the probability of being at home between 10:00am and 10:30am. This probability is calculated to identify daily location-patterns.
4. Calculate $P(l_i|w_m, d_k)$ where d_k is day of the week from $DOW = \{d_1, \ldots, d_k\}$ where $k = 1 : Monday, \ldots, 7 : Sunday$. $P(l_i|w_m, d_k)$ denotes the probability of being at location l_i during time interval w_m on day d_k of the week. For example, the probability of being at home between 10:00am and 10:30am on Mondays. This probability is calculated to identify weekly location-patterns.
5. Compute the *weight* for a location using the probabilities calculated at step (2), (3) and (4) as follows:

$$Weight = 1 - [\, P(l_i)\, P(l_i|w_m)\, P(l_i|w_m, d_k)\,]$$

6. Finally, sort all locations based on *weight* and pick according to that order whenever the system needs to generate challenge questions for a user during an authentication session.

Using the above scheme, lower weight questions that are relatively easy to guess based on an adversary's knowledge of a user's schedule are filtered out and the preference is given to more infrequent events (e.g., visiting a new restaurant) which are more likely to be recalled by users. Once the weight of locations are calculated, the system generates four different kinds of questions. Type 1, 2, and 3 questions are used in Study - I and Type - 4 question is used in Study - II. The details are below.

Type - 1 Question for Study - I: The task of identifying the recently visited places.

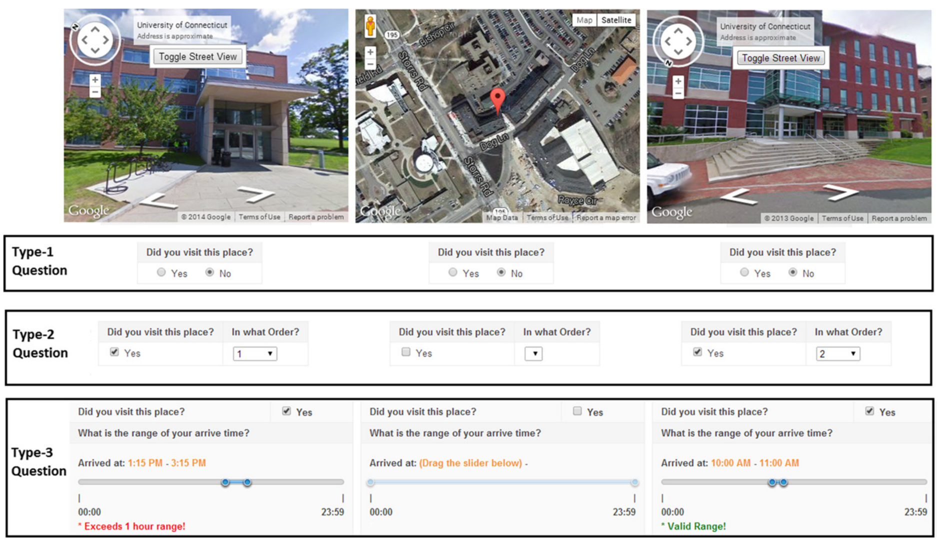
This type of question asks a user to identify the places that he/she visited during a specific day (e.g., What are the places that you visited on May 8th?). For this question type, a user is shown a list of places (i.e., multiple correct answers along with multiple incorrect answers) in the form of both Google Street view images and Google Maps (see Type - 1 Question in Fig. 3).

Type - 2 Question for Study - I: The task of identifying the recently visited places along with the order of visit.

In this case, a user is requested to select all the places that he/she visited during a specific day along with the order of visits (e.g., What are the places that you visited on May 8th and in what order?), see Type - 2 Question in Fig. 3. As in Type - 1, the list may contain multiple correct answers along with multiple incorrect answers (i.e., distracters).

Type - 3 Question for Study - I: The task of identifying the recently visited places along with the time of the visit.

In this case, a user is requested to select all the places that he/she visited during a specific day along with the time window (e.g., What are the places that you visited on May 8th and


University of Connecticut
Address is approximate
Toggle Street View
Google
© 2014 Google | Terms of Use | Report a problem
Map
Satellite
Storrs Rd
Dog Ln
Royce Cir
Google
Map Data | Terms of Use | Report a map error
University of Connecticut
Address is approximate
Toggle Street View
Google
© 2013 Google | Terms of Use | Report a problem

Type-1 Question

Did you visit this place? ○ Yes ● No

Did you visit this place? ○ Yes ● No

Did you visit this place? ○ Yes ● No

Type-2 Question

Did you visit this place? ☑ Yes — In what Order? 1 ▾

Did you visit this place? □ Yes — In what Order? ▾

Did you visit this place? ☑ Yes — In what Order? 2 ▾

Type-3 Question

Did you visit this place? ☑ Yes
What is the range of your arrive time?
Arrived at: 1:15 PM - 3:15 PM
00:00 23:59
* Exceeds 1 hour range!

Did you visit this place? □ Yes
What is the range of your arrive time?
Arrived at: (Drag the slider below) -
00:00 23:59

Did you visit this place? ☑ Yes
What is the range of your arrive time?
Arrived at: 10:00 AM - 11:00 AM
00:00 23:59
* Valid Range!

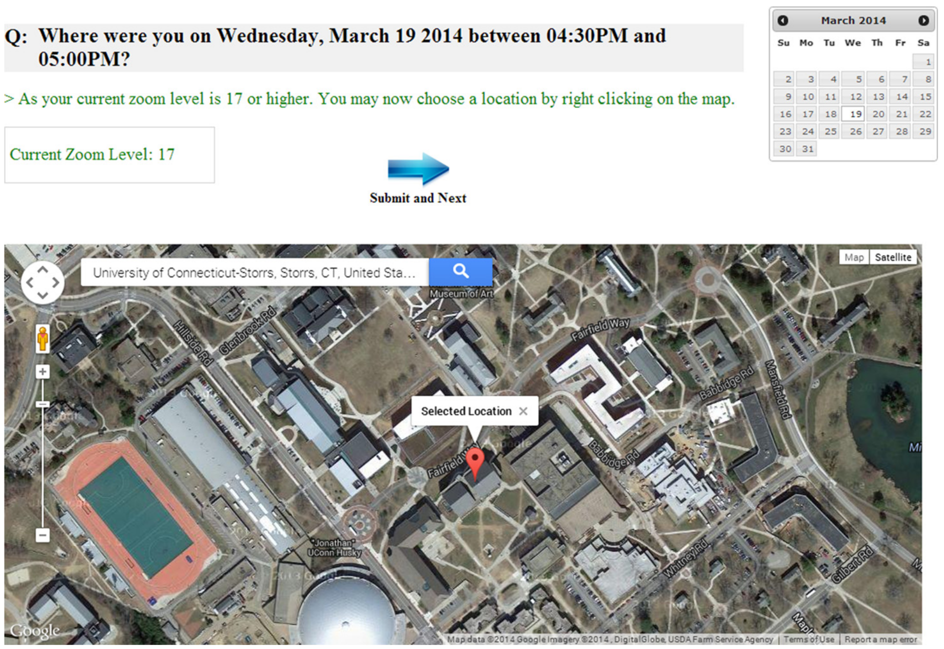
Fig. 3 Screenshots showing examples of three different types of questions used in Study - 1. For brevity, only 3 places are displayed. In reality, up to 15 places are displayed

what time did you arrive there?), (see Type - 3 Question in Fig. 3). As before, the list may contain multiple correct answers along with multiple incorrect answers.

Please note that, as the correct answers involve specifying the order of visits and the time window of visits in Type - 2 and Type - 3 questions respectively, we expect that these types of questions will be much harder for an adversary to guess correctly in real life.

Type -4 Question for Study - II: The task of identifying the recently visited place at a certain time
This type of question asks where a user was at a certain time. An example of Type - 4 questions is shown in Fig. 4. Unlike other question types that have multiple answer choices, in this case, a user is presented with a map and is requested to select the location that he/she had visited during a certain time window of a specific day.

The answer generation component. Apart from the correct choices (i.e., places), the system also needs to generate a list of plausible incorrect answers for multiple choice questions in order to make it sufficiently hard for the adversary to guess. The system leverages several heuristics to pick the incorrect answers. For instance, the system may pick incorrect answers from a user's past important locations that are not visited recently. Moreover, the system computes the Haversine distance [23] between places to avoid selecting places that are too close to each other or too far away from a user's locations, which may be guessed by an adversary more easily. Finally, if the number of incorrect choices is not adequate, additional incorrect choices are generated using Google Map's "Nearby Search" service where the system picks random places within a certain radius of



Q: Where were you on Wednesday, March 19 2014 between 04:30PM and 05:00PM?

> As your current zoom level is 17 or higher. You may now choose a location by right clicking on the map.

Current Zoom Level: 17

Fig. 4 A screenshot showing example of Type - 4 question. A user need to right click on a desired location once the zoom level is 17 or higher to select a location in this question type. The marker specifies the user's selected location. When a user reaches zoom level of 17, he/she will be informed by seeing a message at the top of the page (the message bar turns from red to green) and the right-clicking gets enabled to let a user select a location on the map

a user's actual locations. Furthermore, the Google Places API provides a wide range of place categories [24], which allows us to select places based on the types of places as well.

The authentication component

The *authentication component* is responsible for classifying a user as either legitimate or adversary based on the response of the user to the challenge questions. This is done in two steps. In the first step, the authentication module calculates the score of a user based on the user's response to the challenge questions. In the second step, the system applies a personalized model to classify a user in one of the three categories (e.g., legitimate, naive adversary, and strong adversary). The details are below.

Step 1: user score calculation

In multiple choice questions, as a user can choose multiple answers, we develop a simple mechanism to calculate the score for a particular authentication session. In a session, if a user selects an incorrect answer, he/she gets penalized (i.e., receives negative points). In the current implementation, the penalty for an incorrect answer is equal to the points for a correct answer. The main reason behind penalizing for incorrect answers is to prevent statistical guessing attack where someone can simply select all possible answers to compromise the system. Please note that the lowest possible score is set to 0 to prevent negative scores which may happen due to the penalization scheme. For example, consider a Type-1 question that has 9 options, and 4 out of 9 options are correct. In this case, if a user selects 4 options where 3 of them is correct and one of them is incorrect, the score for this question will be 0.5 out of 1. Likewise, the same penalization formula applies to Type-2 and Type-3 questions. However, if a user selects a correct location but fails to identify the correct order for Type-2 questions or fails to identify the correct time window for Type-3 questions, he/she is not penalized.

The methods for calculating scores for different types of questions are described below.

(a) Score calculation for question Type - 1 (Study - I):

$$Score_{q1} = (P \times L_{correct}) - (P \times (S - L_{\mathrm{correct}}))$$

(b) Score calculation for question Type - 2 (Study - I):

$$Score_{q2} = (P \times O_{correct}) - (P \times (S - O_{correct}))$$

(c) Score calculation for question Type - 3 (Study - I):

$$Score_{q3} = (P \times T_{correct}) - (P \times (S - T_{correct}))$$

In the above formulas,

$$L_{correct} := \textit{number of locations answered correctly,}$$
$$O_{correct} := \textit{number of locations ordered correctly,}$$
$$T_{correct} := \textit{number of correctly answered time range,}$$
$$S := \textit{number of selected locations,}$$
$$P = \frac{1}{N_{correct}} \text{ where } N_{correct} := \textit{number of correct answers.}$$

(d) Score calculation for question Type - 4 (Study - II): In case of Type-4 questions, users answer to questions by placing the marker based on their best guess estimate of their location on the Google Map. As users may not place the marker on exactly the same location coordinates estimated/identified by the location extraction component, there is an error tolerance (e.g., 100 meters great circle distance) in our system, which is calculated based on the Haversine distance between the selected coordinates and the estimated location. If the distance between the selected geographical location and the estimated location is greater than 100 meters, the answer is considered as incorrect and the score is set to 0. For multiple questions, users may receive partial credit. For instance, if a user answers 4 out of 5 questions correctly, the user will receive 0.8 point.

Step 2: model-based authentication

As different users often differ in terms of mental ability to recall past events, they are expected to perform differently in answering location-based questions. Therefore, authenticating users by relying solely on their accuracy score may not be a good idea. Hence, in our work, to authenticate users, we consider their historical performance and response patterns as well. More specifically, we attempt to build a model for each user that learns each user's response pattern, and subsequently leverages this model to identify legitimate users. Please note that, due to the use of models, to compromise the authentication scheme, an attacker needs to closely imitate the response behavior of a legitimate user (i.e., having high accuracy does not necessarily guarantee access). In this work, we evaluated a Bayesian classifier based authentication algorithm and compared the performance against a simple threshold based scheme. They are presented below in increasing order of complexity.

(i) Threshold-based scheme for authentication. Unlike textual password, where a user either enters the correct password or fails, in location-based scheme, a user may answer a subset of questions correctly. Hence, ideally, it should be possible for someone to get access even with less than perfect score. In our system, to compare the performance of more sophisticated models, we calculated the authentication rate of legitimate users using a global threshold based scheme, where a user is identified as a legitimate user if his/her score is greater than some predefined threshold δ. We vary the value of δ from 100 % to 50 % in our study.

(ii) Bayesian classifier based scheme for authentication. We next explore a model based on a Bayesian classifier to predict whether a given response comes from a legitimate user (i.e., u) or an adversary (i.e., u') based on k response features ($f_1 \ldots f_k$). For example, response time (i.e., the time taken to answer the questions) can be one such feature. Please note that we create separate models for each question types. For example, for user 1, we have three different models for three different question types. Moreover, we have one model that represents strong adversary, one model that represents naive adversary and one model that represents the community of strong and naive adversaries in the system (see Section "Accuracy of model-based authentication for Study - I and Study - II").

Let's assume that for each question type Q_i, we have n responses $(r_1, \ldots, r_n)$ which are obtained from n different sessions for a user. Each such response r_i can be represented by the response features ($f_1 \ldots f_k$). Hence, Naïve Bayes Classifier for this case can be written as follows for each question type Q_i and for each response r_i:

$$P(u|f_1 \ldots f_k) = \frac{P(u)P(f_1 \ldots f_k|u)}{P(f_1 \ldots f_k)}, \tag{2}$$

where $P(u|f_1 \ldots f_k)$ (i.e., $posterior(u)$) is the probability of being a legitimate user based on the response features. $P(u)$ is the prior probability distribution of the legitimate user. We assume that the chance of being a legitimate user is 50 % (i.e., equal probability), so $P(u) = P(u') = 0.5$. $P(f_1 \ldots f_k|u)$ represents the joint probability of responses given a user is legitimate. Since f_i's are independent (based on our assumption), $P(f_1 \ldots f_k|u)$ can be rewritten as the product of the component probabilities as follows:

$$P(f_1 \ldots f_k|u) = p(f_1|u)p(f_2|u) \ldots p(f_k|u), \tag{3}$$

The denominator $P(f_1 \ldots f_k)$ of equation 2 represents the joint probability of responses' features. This can be expressed as follows:

$$P(f_1 \ldots f_k) = P(u)P(f_1 \ldots f_k|u) + P(u')P(f_1 \ldots f_k|u'), \tag{4}$$

where $P(f_1 \ldots f_k|u')$ denotes the probability of being an adversary (i.e., non-user) based on the response features. Once we train the model, the system uses the individualized model to calculate the probability of a person being a legitimate user based on his/her response patterns.

Evaluation

For evaluating various aspects of location-based authentication systems, we designed two separate studies based on the type of questions that were asked. In both studies, users are allowed to participate either alone or in pairs (to simulate strong adversary).

In case of single participants, over the course of the experiment, a user was presented with two sets of authentication questions multiple times each week. The first set of questions was generated based on participant's own data. The second set of questions was generated based on a randomly selected user's data whose identity was not revealed. We compensated each participant $15 for two weeks of participation. A participant may continue the study for longer than two weeks.

In case of paired participation, to simulate strong adversaries, we recruited participants in pairs (e.g., close friends, married couples). Over the course of the experiments, each participant was presented with three sets of authentication questions multiple times each week. The first set of questions was generated based on participant's own data. The second set of questions was generated based on participant's pair's (e.g., friend or couple) data. The third set of questions was generated based on a randomly selected user's data whose identity was not revealed. In case of paired participation, we compensated each participant $25 for two weeks of participation. Participants may continue the study for longer than two weeks.

After a participant signed the consent form, during the course of the studies, questions for each user are generated and posted on a website periodically (e.g., multiple times each week), and the link (i.e., URL) to the web-based survey page is emailed to the participant's email address. Accuracy in answering the questions for each session was logged by the server. To preserve the privacy of participants, (1) user's identity was not stored along with the data, and (2) a unique URL associated to a user was generated for each session by encrypting combination of *User Id* (assigned randomly at the beginning of the study) and *Session number*.

Both studies were approved by the IRB. Flyers were posted on campus and emails were sent to the engineering department email list to recruit participants. Participants were not given any feedback regarding his/her performance throughout the experiments.

Results from Study - I

During a period of 30 days, we collected a total of 1833 question-answer responses from 14 participants (6 paired and 2 single participants). One of the couples withdrew from the study after two weeks of participation. All of the participants were university students at UCONN. 8 of the 14 participants were UCONN graduate students (57 %) and 6 of them (43 %) were UCONN undergraduate students. 9 of the 14 participants were male (64 %).

To explore the effects of different factors on performance of different categories of users (e.g., legitimate, adversary), we used an ordinary least squares (OLS) regression model [25] to analyze the data. Since we have repeated measurements from the same individual, each user is represented using an indicator (i.e., dummy variable) to control unobserved heterogeneity between individuals such as ability of recalling past events. The coefficients listed in Table 2 show the relationships between the dependent variable (i.e. accuracy score) and different independent variables (e.g., question type). The categorical variables are designated using their baselines. The coefficients marked with the "*" represent the variables that have statistically significant (p <0.05) effect on user performance (i.e., accuracy score). We discuss the details regarding our findings below.

- **Effect of question types on accuracy score.**
  In our evaluation, Type - 1 questions are answered correctly more often than Type - 2 and Type - 3 questions. The effect of Type - 2 and Type - 3 questions on accuracy were significant for all user types with a negative coefficient (i.e., all users were negatively affected). This result was expected as the first question type is relatively easier to answer compared to question Type - 2 and Type - 3 (e.g., recalling a location is easier than recalling an arrival time to the location). To further investigative the effect of different question types on user accuracy score, a Non-parametric Kruskal-Wallis test was used. There were significant differences in accuracy score among the three type of questions for all user types (i.e., legitimate users, strong and naive adversarial users). We found the followings: ($H(2)$ =29.872, p <0.01) for legitimate users, ($H(2)$ =84.360, p <0.01) for strong adversarial users, and ($H(2)$ =39.519, p <0.01) for naive adversarial users. Post hoc pairwise comparisons using the Wilcoxon signed-rank tests indicated that there is a significant difference

Table 2 Coefficients for the OLS model for the user Study -1

Feature	Coefficients			Baseline
	Legitimate	Strong adversary	Naive adversary	
Number of Correct Answers	-0.047 *	-0.015	0.004	
Days Ago	-0.024 *	-0.21 *	-0.003	
Time to Answer (seconds)	-8.877e-5	0.000	0.000 *	
Number of Options: 15 options	-0.012	-0.099 *	-0.023 *	9 Options
Order of Questions: [Q2-Q3-Q1]	0.045	0.008	-0.010	Order [Q1-Q2-Q3]
Order of Questions: [Q3-Q2-Q1]	0.089 *	0.065	-0.014	Order [Q1-Q2-Q3]
Question Type: QType-2	0.109 *	-0.270 *	-0.059 *	QType-1
Question Type: QType-3	-0.124 *	-0.249 *	-0.057 *	QType-1

between Type - 1 and Type - 2 questions, as well as Type - 1 and Type - 3 questions for all user types, but not between Type - 2 and Type - 3 questions for all user types other than legitimate users. The results are shown in Table 3.

- **Effects of ordering of questions on accuracy score.**
  As the difficulty levels of different question types were different which may require different levels of effort to recall the correct answers, during the experiment, we changed the order of questions to investigate the effect of ordering on users' accuracy score. We tried three different orderings as follows: [Q1-Q2-Q3], [Q2-Q3-Q1], and [Q3-Q2-Q1]. For example, if the ordering is [Q1-Q2-Q3], it implies that the user was presented Type - 1 question first during the session followed by Type - 2 and Type - 3 questions respectively.
  While the effect of different ordering on accuracy score is insignificant for all adversary types, the legitimate user's accuracy score appears to be affected positively by ordering [Q2-Q3-Q1] and [Q3-Q2-Q1]. This finding suggests that asking questions that involve recalling more details regarding past events actually improve legitimate user's performance (see Fig. 5).
  Another interesting observation is that, users spend more time to answer the first presented question for all user types. This is due to the fact that identifying the presented Google Map and Street View image locations for the first time takes more time, which reduces the time to answer the subsequent questions. For example, legitimate users took 138.90 seconds on average to answer the Type - 1 question when the Type - 1 question is presented first (i.e., ordering is [Q1-Q2-Q3]), compared to 37.46 seconds on average when presented last (i.e., ordering is [Q3-Q2-Q1]).
- **Effect of total number of answer choices on accuracy score.**
  In our study, "Number of Options" refers to the number of given answer choices presented for a given session. Intuitively, the higher the number of options, the harder to identify the correct choices based on guessing. In our study, we compared the performance between 9 options and 15 options. Increasing the number of options from 9 to 15 significantly reduced the accuracy of adversarial users, whereas the legitimate user's accuracy score is not affected significantly (see Fig. 6). This suggests that presenting a large number of choices may be an effective way to thwart attacks launched by adversaries.
- **Effect of number of correct answers on accuracy.**
  In our study "Number of correct answers" indicates the number of correct choices for a given question. For instance, if the total number of answer choices is 15 for a given question and 5 out of the 15 are correct, the "number of correct answers" is 5. In our study, the performance and number of correct answers was found to be negatively correlated for legitimate users. In other words, questions with more

Table 3 Pairwise Wilcoxon signed-rank tests for different question types for accuracy score analysis

	Legitimate		Strong		Naive	
	Z	*p*	*Z*	*p*	*Z*	*p*
Type-1 vs Type-2	-2.758	<0.006	-8.103	<0.001	-5.160	<0.001
Type-1 vs Type-3	-5.597	<0.001	-7.298	<0.001	-4.578	<0.001
Type-2 vs Type-3	-2.189	<0.029	-1.327	0.184	-0.745	0.456

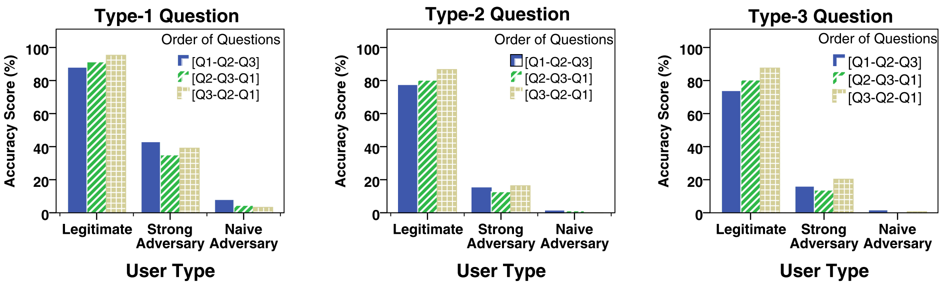


Fig. 5 Impact of ordering of questions on accuracy score for Study - I

correct answers were answered less correctly by legitimate users. Interestingly, while increasing the “number of correct answers” negatively affects accuracy score of legitimate users, there is no significant effect on accuracy score of adversaries.

- **Effect of staleness of the data on accuracy score.**
  As expected, our study confirmed that recalling recently visited locations is easier than recalling locations visited a long time ago. While accuracy score of naive adversaries were not affected by the staleness of the data, the effect of stale data is found to be significant for legitimate users and strong adversaries. The negative coefficient implies that accuracy score drops as the staleness of the data increases.
- **Response time and accuracy score.**
  In our study, “Time to Answer” indicates the amount of time that was taken by a user to answer a question. While, the effect of time on accuracy score is insignificant for legitimate users and strong adversaries, it is significant for naive adversaries. This could be due to the fact that, as the locations are not familiar to naive adversaries (i.e., have not seen nor visited before), they spend more time to identify presented locations and to select the answers. Although their accuracy score appears to increase as the time spent to respond increases, the average accuracy of naive adversaries remained below 2.7 %.
  We also observed that the adversarial users (i.e., strong and naive adversaries) generally took less time than legitimate users to answer the questions.
  Non-parametric Kruskal-Wallis tests show significant differences in response time among the three user types (i.e., legitimate users, strong and naive adversarial users) for 3 different question types, ($H(2)$=11.781, p <0.003) for Type - 1 questions, ($H(2)$=27.352, p <0.001) for Type - 2 questions, and ($H(2)$=82.481, p <0.001) Type - 3 questions. Furthermore, post hoc pairwise comparisons using the Wilcoxon

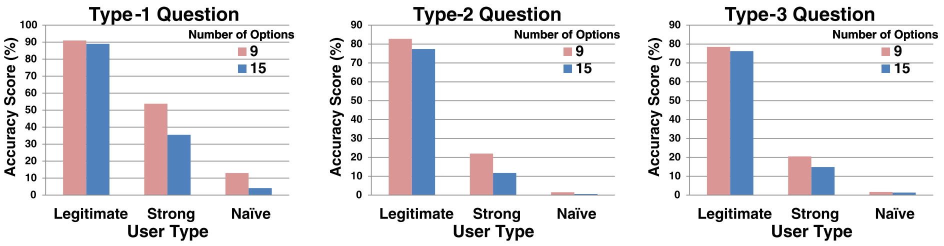


Fig. 6 Impact of number of options on accuracy score for Study - I

signed-rank tests indicated that there is a significant difference between legitimate users and strong adversary for question Type - 2 and Type - 3 as well as legitimate users and naive adversary for all question types (see Table 4). Moreover, even within the adversary types, the time taken to answer all question types were significant. The part of the reason is because while the naive adversarial users had no knowledge about the identity of a person, strong adversarial users had significant knowledge regarding schedule of the targeted users (i.e., close friends) thus spending more time to guess.

Results from Study - II

In Study - II, a user is asked to select the location(s) that he/she visited at a certain time. A user can select the location(s) using an interactive map that was implemented leveraging Google Maps API. We set the initial zoom level of Google Maps to 2 where most of the world is visible while there is no repetition of the map unlike zoom level 1. The rationale behind this choice is to avoid influencing users to select locations from a certain geographic area, which may also reduce overall security of the system [26]. In order to select a location on the map, the minimum required zoom level is set at zoom level 17, which gives reasonable details and higher security since an adversary has to guess a location at a finer resolution. Once a user zooms in at level 17, he is informed by displaying the following message at the top of the web page (the message bar turns from red to green): "As your current zoom level is 17 or higher, you may now choose a location by right clicking on the map". To select a location, "Right-clicking" functionality was chosen as left-clicking is used for another functionality (e.g., double click zooms in). Once a location is selected, users see a marker at the selected location (e.g., like the one in Fig. 4). As zooming in from zoom level 2 to 17 or higher may slow down the user and impact the usability aspect, we provide a search box that may be used to zoom-in on the right area/location very quickly. Specifically, we took advantage of the Google Place Autocomplete feature, which returns a list of suggestions for locations and predicted search terms. From this list, a user may select a place to zoom in quickly.

During the experiment, several response specific features (e.g., number of zoom-in/zoom-out, number of search etc.) were recorded while users were answering Type - 4 questions. We subsequently tried to leverage such user specific features along with accuracy to classify users as legitimate or attacker. During a period of 30 days, we collected a total of 2423 question-answer responses from 15 participants (6 paired and 3 single participants). All of the participants were university students at UCONN. 12 of the 15 participants were UCONN undergraduate students (80 %) and 3 of them (20 %) were UCONN graduate students. 9 of the 15 participants were female (60 %). The age of participants ranged from 18 to 29 years with an average age of 21.13 years (SD = 3.26).

Table 4 Pairwise Wilcoxon signed-rank tests for different user types for response time analysis

User Type\Question Type	Type-1		Type-2		Type-3	
	Z	*p*	*Z*	*p*	*Z*	*p*
Legitimate vs Strong	-0.83	0.40	-2.36	<0.01	-5.14	<0.01
Legitimate vs Naive	-3.35	<0.01	-5.11	<0.01	-8.82	<0.01
Strong vs Naive	-2.26	<0.02	-2.75	<0.06	-3.70	<0.01

Mixed-effect logistic regression model [27] was used to analyze the effect of the independent variables (e.g., number of zoom in) on the dependent variable (i.e. response correctness). Multivariate, mixed-effect logistic regression model contains fixed effects and random effects. In our analysis, a *user* was included as a random-effect variable to account for multiple measurements (i.e., multiple responses) within users. More specifically, we use a random-intercepts model to allow each user to have his/her baseline likelihood for answering a question correctly. All other independent variables were included as fixed-effect variables. Table 5 shows the model coefficients. The coefficients marked with the "*" represent the variables that have statistically significant (p <0.05) effect on user performance (i.e., response accuracy). The log odds was used as a measure of association between the dependent variable (i.e. response accuracy) and different independent variables (e.g., number of zoom in) and their influencing factors. The coefficients (e.g., question weight, time taken) listed in Table 5 represent the change in the dependent variable (i.e., response accuracy) when the coefficient is increased by one-unit while controlling all other numerical variables at their mean values.

The intercept represents the mean accuracy when all numerical variables are controlled at their mean values. We discuss the details regarding our findings below.

- **Effect of zoom level on accuracy score.**
  "Zoom Level" indicates how far a user has zoomed into the map at the time of selecting a location. Note that in our study the minimum required zoom level is 17 in order to select a location on the map. A user may zoom in upto level 21 if desired. While the effect of zoom level on accuracy is found to be significant for legitimate and strong adversarial users, it is insignificant for naive adversarial users. The positive coefficient for "Zoom Level" suggests that response accuracy increases when users select their locations at a higher zoom level.
- **Effect of number of zoom in/out on accuracy score.**
  "Zoom in/out" feature is used by a user to interactively select a location. Intuitively, an attacker may need to explore more before selecting a location. In our system, number of zoom in/out is calculated by comparing the previous zoom level against

Table 5 Coefficients for the Mixed-Effect Logistic Regression model for user Study-2. The coefficients show whether they had a statistically significant effect on response accuracy

Feature	Coefficients		
	Legitimate	Strong adversary	Naive adversary
Intercept	0.82	1.03	0.21
Time to answer (seconds)	-0.00142	0.00326	0.00545
Days Ago	-0.229*	-0.231	-0.102
Zoom Level	0.995*	0.750*	-0.567
Number of Zoom In	-0.231*	-0.368*	-0.0922
Number of Zoom Out	0.643	0.0788	0.205
Number of Search	-0.140	0.389	-0.251
Number of Drag	-0.0285	-0.0662	-0.0360
Number of Changes	0.370	-0.288	0.112
Question Weight	-3.997	-7.123*	-0.000781

Significant features are designated by a * next to their coefficients.

the current zoom level when a user zooms in/out. User performance and the number of *zoom in* was found to be negatively correlated for legitimate and adversarial users. In other words, users who zoomed in more were less likely to answer correctly, which was expected. This could be due to the fact that, if a user is not sure about his/her answer, he/she is more likely to perform more *zoom in* on the map. Please note that when users use search box to navigate to a location on the map, they are often brought to the appropriate zoom level in one step(i.e., generally near zoom level 17). In such cases, the number of zoom in operations is not counted. The number of zoom out had no significant effect on response correctness for all users.

- **Effect of number of search on accuracy score.**
  "Number of Search" refers to the number of times searching (via search box) was performed. The number of search did not appear to affect response correctness. Most of the participants used search box to navigate the map. We observed that 2385 out of the 2423 (98.4 %) questions were answered by using the search box, and the search box was used more than once in 45 of these cases (0.01 %). This indicates that they generally performed one search operation to find their best guess of estimated location. This also indicates that majority of the users used search box for faster navigation.
- **Effect of number of drags on accuracy score.**
  "Number of Drags" indicates the number of times panning was done on the map by clicking and dragging. The effect of this feature on response correctness was found to be insignificant for all user types. Part of the reason might be because the participants generally used search box to find a location rather than dragging the map.
- **Effect of number of changes on accuracy score.**
  "Number of Changes" indicates the number of attempts to select a location on the map. In other words, after placing the marker by right clicking on the map, a user may change his/her mind to choose another location. The number of changes had no significant effect on accuracy score for all user types. Although participants were allowed as many attempts as they want to choose a location on the map, they were quite steady in selecting a location by placing the marker on their best guess estimate of their location. We observed that participants chose a location in one attempt in 2313 out of 2423 responses (95.4 %).
- **Effect of staleness of data on accuracy score.**
  "Days ago" feature coefficient indicates that the number of elapsed days between the question date and the user's response date (i.e., staleness of the data). As expected, our study confirmed that recalling recently visited locations during a certain time window is easier than recalling locations visited a long time ago. While response correctness of both strong and naive adversaries was not affected by the staleness of the data, the effect of stale data is found to be significant for legitimate users. The negative coefficient implies that response correctness drops as the staleness of the data increases.
- **Effect of question weight on accuracy score.**
  "Question Weight" is calculated based on the randomness of the locations visited. Users are asked about locations that have the highest level of randomness based on their spatio-temporal behavior. Asking higher weight questions significantly reduced

the accuracy of strong adversarial users, whereas legitimate users' accuracy is not affected significantly. This suggests that giving preference to high weight questions (i.e., asking about infrequent events such as visiting a new restaurant) can be an effective way to thwart attacks launched by strong adversaries.

- **Effect of time to answer on accuracy score.**
  "Time to Answer" indicates the amount of time that was taken by a user to answer a question. The effect of time on response correctness was insignificant for legitimate and adversarial users. As shown in Table 6, we observed that adversarial users (i.e., strong and naive adversaries) generally took, on average, less time than legitimate users to answer a question. Answering a type - 4 question took on average 27.79 seconds with median 19 seconds for legitimate users, 24.38 seconds with median 17 seconds for strong adversarial users, and 17.91 seconds with median 13 seconds for naive adversarial users.

In summary, legitimate users performed significantly better compared to adversaries. As we expected, strong adversaries performed better compared to naive adversaries due to their partial knowledge regarding the schedule of legitimate users. We observed that it was very difficult for a naive adversary to guess where a random person was at any given time. We analyzed responses of naive adversarial users and plotted the locations that naive adversarial users selected in user Study - 2 in order to determine the characteristics of their answers. Figure 7 presents a heatmap that indicates the popularity of naive adversaries choosing locations. As we can see from this heatmap, most of the locations (generally common spots such as Student Union Building) were chosen around the UCONN, Storrs campus where the participants were enrolled. 731 out of the 846 (86.4 %) responses of naive adversarial users were within 1 km radius of UCONN, Storrs campus. This observation suggests that avoiding questions regarding commonly visited places in a community may increase the chance of thwarting an attack by an adversary significantly.

Accuracy of model-based authentication for Study - I and Study - II

As different users often differ in terms of mental ability to recall past events and may answer partially (e.g., may select 2 out of 3 correct choices), we attempt to authenticate users based on their imperfect performance instead of relying solely on their accuracy score. More specifically, we build models for each user based on past response patterns, and subsequently leverage that model to identify legitimate users. In our work, we compared the performance of a Bayesian based classifier against a simple threshold based scheme as follows.

Table 6 Time taken for legitimate and adversarial users to answer Type - 4 questions

	Mean	SD	Median	Skewness	Kurtosis
Legitimate Users	27.80	26.097	19.00	3.160	13.226
Strong Adversarial Users	24.38	27.024	17.00	4.366	26.272
Naive Adversarial Users	17.91	17.804	13.00	5.274	48.169

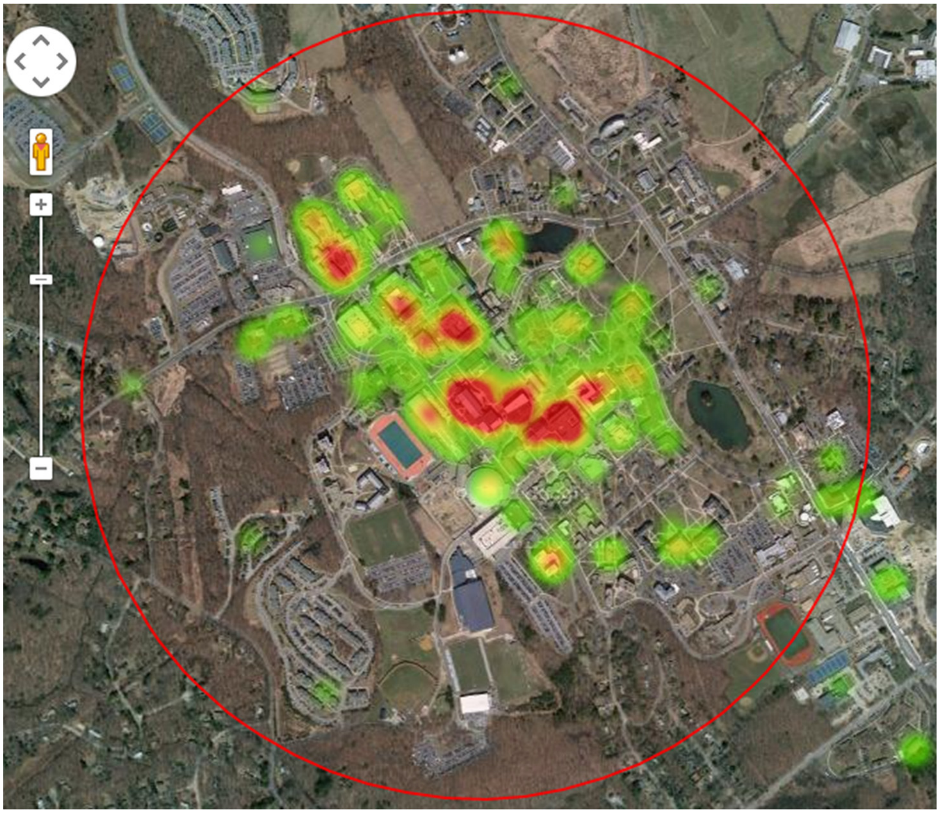


Fig. 7 Visualization of locations chosen by naive adversaries within 1 km radius of UCONN, Storrs campus. The darker red colors indicate the most popular locations that are often selected by naive adversaries. Examples of the most popular locations selected by the naive adversarial users were UCONN's Student Union Building, Main Library, and Northwest Dorm

Classification accuracy of threshold based scheme for Study - I

In our system, to compare the performance of more sophisticated models, we calculated the authentication success rate of legitimate users using a simple global threshold based scheme, where a user is identified as a legitimate user if his/her score is greater than some predefined threshold δ. We vary the value of δ from 100 % to 50 % in our study. The performance is shown in Table 7. As different user's performance vary significantly, the authentication success rate using threshold based scheme is not impressive. Hence, we attempted to authenticate users leveraging personalized models as follows.

Table 7 Authentication success rate for threshold based scheme in Study - 1

	Type-1			Type-2			Type-3		
	Legitimate	Strong	Naive	Legitimate	Strong	Naive	Legitimate	Strong	Naive
Th = *100 %*	68.5 %	10.9 %	0.0 %	59.9 %	7.0 %	0.0 %	44.1 %	0.0 %	0.0 %
Th = *80 %*	77.7 %	17.0 %	0.0 %	64.7 %	7.0 %	0.0 %	53.4 %	0.0 %	0.0 %
Th = *60 %*	90.5 %	39.1 %	2.2 %	74.7 %	8.4 %	0.0 %	71.7 %	11.3 %	0.4 %
Th = *50 %*	95.1 %	51.2 %	6.4 %	80.2 %	11.6 %	0.4 %	83.3 %	17.9 %	1.3 %

Classification accuracy of bayesian classifier based scheme for Study - I

To evaluate the classification accuracy of this scheme, we split a legitimate user's responses into n folds where n denotes the number of sessions. Subsequently, we use data from $(n-1)$ sessions to train the classifier and use the remaining session for testing. We repeat the process n times where each time we use a different session for testing. To test the system, we try three different scenarios as follows.

In the first case, we assume the existence of only strong adversary in the system (i.e., all attackers are strong adversaries who have significant knowledge regarding the targeted user's schedule). In this case, we generate two models. One model represents the legitimate user which is trained using the legitimate user's data. The second model represents the strong adversary which is trained using the strong adversaries' data collected from our field study for all different users excluding the corresponding legitimate user's strong adversary data. This lets us to test the effectiveness of classification accuracy of our system against unknown strong adversary (i.e., how good the classifier is in identifying unknown strong adversary based on community model).

In the second case, we assume the existence of only naive adversary in the system (i.e., all attackers are naive adversaries who are trying to compromise the system without any knowledge regarding the daily routines of the targeted user). In this case, as before, a model that represents the community naive adversary is trained using data from all naive adversaries excluding the corresponding legitimate user's naive adversary data.

Finally, in the third case, we do not distinguish between naive and strong adversary and trained one model that represents the community of strong and naive adversaries in the system. Again, we exclude the corresponding legitimate user's strong and naive adversaries' data.

For evaluation, the cross-validation process is repeated n times in order to obtain an average classification accuracy result. The classification accuracies for legitimate and for adversary for each question type using 3 different scenarios are summarized in Table 8. From Table 8, it can be seen that, regardless of the assumption regarding the existence of different kinds of adversaries, the Bayesian based classifier scheme can distinguish between legitimate users and adversaries with high accuracy. The highest classification accuracy is obtained when the system assumes the existence of naive adversary alone (e.g., average accuracy is 95.9 % for Type - 1 question, 91 % for Type - 2 question, and 95.8 % for Type - 3 question). On the other hand, when modeled against strong adversary, the classification accuracy rate is slightly lower (e.g., average accuracy is 84 % for Type - 1 question, 74.3 % for Type - 2 question, and 79.9 % for Type - 3 question). Intuitively, since a strong adversary has significant knowledge regarding a user's schedule, a strong adversary is more likely to gain access to the system by answering questions more accurately.

Table 8 Authentication success rate for Bayesian classifier based scheme in study - 1

Question Type\ Against Adversary	Against naive adversaries		Against strong+ Naive adversaries		Against strong adversaries	
	Legitimate	Naive	Legitimate	Strong+Naive	Legitimate	Strong
Type-1	95.9 %	2.3 %	89.0 %	14.8 %	84.0 %	25.2 %
Type-2	91.0 %	1.7 %	80.3 %	8.3 %	74.3 %	17.0 %
Type-3	95.8 %	1.6 %	86.9 %	9.7 %	79.9 %	17.2 %

Table 9 Authentication success rates after answering different number of questions for the threshold based scheme in study-2.*Th* denotes threshold value and *N* denotes the number of questions aggregated across all sessions

	N = 2			N = 4			N = 6		
	Legitimate	Strong	Naive	Legitimate	Strong	Naive	Legitimate	Strong	Naive
Th = 100 %	73.0 %	12.5 %	1.1 %	57.6 %	5.6 %	0.0 %	44.8 %	2.4 %	0.0 %
Th = 80 %	73.0 %	12.5 %	1.1 %	57.6 %	5.6 %	0.0 %	70.5 %	7.1 %	0.0 %
Th = 60 %	73.0 %	12.5 %	1.1 %	80.8 %	12.9 %	0.5 %	85.8 %	14.0 %	0.3 %
Th = 50 %	90.9 %	30.3 %	8.3 %	92.2 %	24.5 %	3.5 %	93.9 %	21.2 %	1.4 %

Classification accuracy of threshold based scheme for Study - II

Similar to user Study - I, we first tried the simple global threshold based scheme for Study - II. In this case, we calculated the score by calculating average accuracy for multiple questions answered. If a user's average response accuracy score is greater than some predefined threshold δ, we identified the user as a legitimate user. We vary the value of δ from 100 % to 50 % and the number of questions from 1 to 6. For brevity, we only show the performance for $N = 2$, $N = 4$, and $N = 6$ in Table 9. In Table 9, for example, if the threshold is 50 % and $N = 4$, then the user needs to answer at least 2 out of the 4 questions correctly to be classified as a legitimate user. As can be seen in Table 9, the performance of threshold based scheme is not impressive, which motivates us to explore the Bayesian classifier based scheme as explained next.

Classification accuracy of bayesian classifier based scheme for Study - II

Similar to user Study-I, we used the Bayesian classifier based scheme and assessed the classification accuracy of this classifier to identify legitimate users. In this case, we utilize eight response features (listed in Table 10) and observed different classification accuracy for different response features. Figure 8 shows the effect of individual response feature on authentication success rate considering 3 different scenarios, namely, against naive adversaries, against strong adversaries, and against strong and naive adversaries). We also vary the number of challenge questions that we consider in each session. The classification accuracies for legitimate and for adversarial users for Type - 4 question for 3 different scenarios considering different number of questions (e.g., $N = 2$, $N = 4$, and $N = 6$) are summarized in Table 11.

In Fig. 9, each graph represents the effect of combinations of eight features on mean classification accuracy after answering N questions aggregated across all sessions. For brevity, we only show the plots for $N = 1$, $N = 3$, and $N = 6$. From Fig. 9, we observe that,

Table 10 Eight response features used in Study - II for calculating authentication success rates

	Feature
1.	Response Accuracy
2.	Time to Answer
3.	Zoom Level
4.	Number of Drags
5.	Number of Search
6.	Number of Zoom In
7.	Number of Zoom Out
8.	Number of Changes

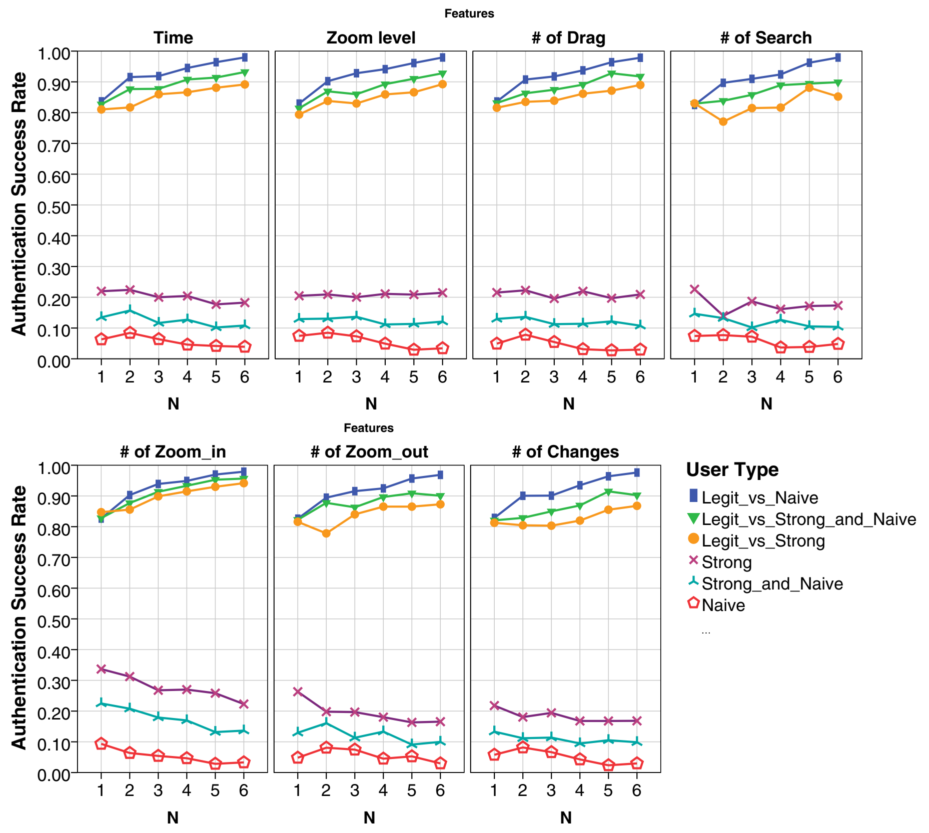


Fig. 8 Each plot represents the effect of response features on the authentication success rate for 3 different attack scenarios (only naive, only strong, strong and naive) for different number of questions (e.g., N=1 to 6). Response accuracy along with the designated response features are used to compute mean authentication success rate for different scenarios

as the value of N increases from 1 to 6, regardless of the modeled adversary, the authentication success rate increases for legitimate users and decreases for adversarial users. This indicates that increasing the number of questions per session is an effective way to increase the security of the system. Moreover, combining more response features generally increases the authentication success rate for both legitimate and adversarial users, but the increase is steeper for adversarial users. In our evaluation, setting N = 6 while considering all 8 response features gives the highest classification accuracy.

Table 11 Authentication success rate after answering different ♯ of questions aggregated across all sessions and using all possible combinations of eight response features for Bayesian classifier based scheme in Study - 2

♯ of questions\ against adversary	Against naive adversaries		Against strong+ Naive adversaries		Against strong adversaries	
	Legitimate	Naive	Legitimate	Strong+Naive	Legitimate	Strong
N = 2	91.6 %	14.9 %	87.4 %	23.0 %	82.2 %	25.6 %
N = 4	96.4 %	11.0 %	93.3 %	18.2 %	89.5 %	24.5 %
N = 6	98.1 %	5.7 %	95.6 %	14.2 %	94.1 %	19.9 %

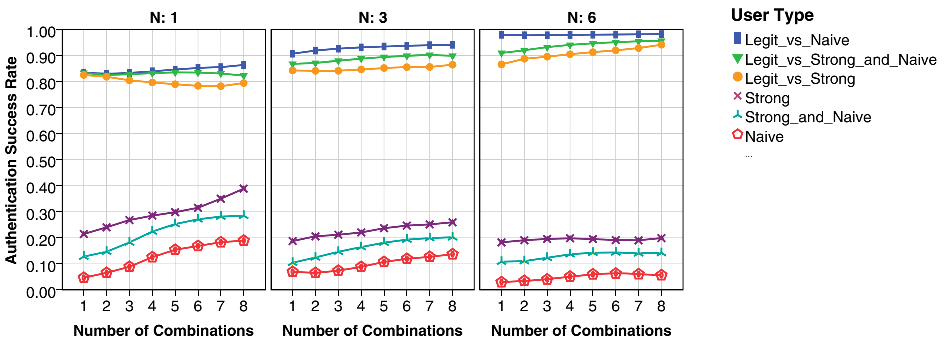


Fig. 9 Each plot represents the effect of combing 8 response features on the authentication success rate for 3 different scenarios for 1, 3, and 6 questions respectively

Comparison of Study - I and Study - II

To summarize, in Study - I, three different kinds of questions are used with varying difficulty level. Specifically, Type - 1 question asks users to identify the locations visited for a given day. Type - 2 question asks users to identify the locations and the order of visits for a given day. Finally, Type - 3 question asks users to identify the locations and the time window when the places are visited for a given day. In each case, possible answers to challenge questions are presented in the form of multiple choices to facilitate recall. In contrast, in Study - II, Type - 4 question asks where a user was at a certain time. In this case, users are shown a map (i.e., possible answers are not presented) and are asked to select locations that he/she has visited during a certain time window for a specific day.

To authenticate users based on their performance, we first tried the simple global threshold based scheme. Due to poor classification accuracy of the threshold based scheme, we subsequently implemented a personalized Bayesian model based classifier that leverages user's response pattern (e.g., time taken to answer a question and accuracy) to authenticate users.

Using the Bayesian model, we observed average authentication success rate of 95.9 % for Type - 1 question, 91 % for Type - 2 question, 95.8 % for Type - 3 question, and 98.1 % for Type - 4 question while considering only naive adversary. On the other hand, when modeled against strong adversary, we observed authentication success rate of 84 % for Type - 1 question, 74.3 % for Type - 2 question, 79.9 % for Type - 3 question, and 94.1 % for Type - 4 question. In both cases, Type - 4 question was found to be the best option in terms of accuracy. In all cases, Bayesian classifier based scheme outperforms simple threshold based scheme.

Finally, in our study, question types indeed had a significant effect on user performance that may affect the usability and security of smartphone based authentication systems.

Discussion

In this paper, we investigate the problem of generating challenge questions leveraging user's location data and the effect of question types on user performance. We also presented Bayesian classifier based authentication models that learn each user's response pattern and subsequently leverage that model to identify legitimate users with high accuracy. Please note that as one of the main purposes of this paper is to demonstrate the effect of challenge question types on user performance, we tried two different classification algorithms to identify legitimate users (i.e., threshold based scheme and Bayesian classifier based scheme). To further improve the accuracy, trying other machine learning algorithms (e.g., k-means clustering, support vector machine (SVM)) on the same data set is one of our future works. Also, we realize that there are other types of smartphone usage data that might be used for generating challenge questions (e.g., SMS log, call history, application usage behavior). We aim to explore all these different data types for generating challenge questions in future.

Finally, while our study suggests that location based scheme holds great promise for generating dynamic challenge questions, we do realize that the presented system may raise privacy concerns among users if they require sharing the location information with the server. One possible solution is to run the authentication service locally on smartphones and store the data in encrypted form on local devices, which may be developed by trusted third party or the service providers. In such cases, even if a device is stolen, the attacker will not be able to access the data without being able to compromise the system and the software, which is nontrivial. Also, data that are more than "x" days old may be deleted automatically, limiting the amount of data that might be compromised even if an attacker can break the software.

While many other alternative solutions are possible, we would like to stress that in-depth discussion of possible solutions to address the privacy concerns that are common across various dynamic human-behavior based KBA schemes is beyond the scope of our current work, which we aim to investigate in our future work.

Conclusion

In this paper, we investigate the effect of challenge question style on users' performance for a location-based authentication system where the system generates challenge questions by exploiting users' recent location information captured by smartphones. We implemented the location tracking application based on Wi-Fi AP information and conducted two real-life studies for 60 days in total. Our findings suggest that the style of challenge questions can have a significant effect on user performance. To account for variability in individual user's recall ability, we also presented Bayesian classifier based authentication models that learn each user's response patterns and subsequently leverage that model to identify legitimate users with high accuracy while reducing the success rate of adversaries. To conclude, based on our findings, we strongly believe that effective design of challenge questions can significantly improve the usability of dynamic knowledge-based authentication systems and improve the overall system security.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

YA, MK and AB proposed the idea of authenticating users leveraging spatio-temporal behavior data. YA designed, developed and implemented the location-based authentication system and conducted the experiments under the supervision of MK. YA and MK wrote the manuscript. MK designed the study and drafted the manuscript. AB provided implementation guidance for the proposed approach. YA and SK implemented the location extraction component algorithm. RJ helped with the implementation of the location extraction component. NN participated in the experimental validation of the model for authentication. All authors read and approved the final manuscript.

Acknowledgment

This work is supported by the National Science Foundation under Grant No. CNS-1251962. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agency.

Author details

[1]Department of Computer Science and Engineering, University of Connecticut, CT, Storrs, USA. [2]Ameresco, Framingham, MA, USA. [3]Department of Computer Science, Salem State University, Salem, MA, USA.

Received: 22 December 2014 Accepted: 4 May 2015
Published online: 24 June 2015

References

1. Schechter S, Reeder RW (2009) $1 + 1 =$ you: Measuring the comprehensibility of metaphors for configuring backup authentication. In: Proceedings of the 5th Symposium on Usable Privacy and Security. ACM, New York, NY, USA. p 9
2. Chen Y, Liginlal D (2007) Bayesian Networks for Knowledge-Based Authentication. IEEE Trans Knowl Data Eng 19(5):695–710. IEEE Educational Activities Department, Piscataway, NJ, USA
3. Chen Y, Liginlal D (2008) A maximum entropy approach to feature selection in knowledge-based authentication. Decision Support Systems 46(1):388–398
4. Jakobsson M (2012) The Death of the Internet. John Wiley & Sons
5. Rabkin A (2008) Personal knowledge questions for fallback authentication: Security questions in the era of facebook. In: Proceedings of the 4th Symposium on Usable Privacy and Security. ACM, New York, NY, USA. pp 13–23
6. Schechter S, Brush AB, Egelman S (2009) It's no secret. measuring the security and reliability of authentication via "secret" questions. In: Proceedings of the 2009 30th IEEE Symposium on Security and Privacy. IEEE Computer Society, Washington, DC, USA. pp 375–390
7. O'Gorman L, Bagga A, Bentley J (2004) Call center customer verification by query-directed passwords. In: Financial Cryptography. Springer, Berlin Heidelberg. pp 54–67
8. Asgharpour F, Jakobsson M (2007) Adaptive challenge questions algorithm in password reset/recovery. First International Workshop on Security for Spontaneous Interaction (IWIISI '07), Innsbruck, Austria, (2007) 7:6
9. Nosseir A, Connor R, Dunlop MD (2005) Internet authentication based on personal history-A Feasibility Test. In: Proceedings of Customer Focused Mobile Services Workshop at WWW2005. ACM Press, New York, NY, USA
10. Nishigaki M, Koike M (2007) A user authentication based on personal history-a user authentication system using e-mail history. J Syst Cybern Inform 5(2):18–23
11. Nosseir A, Connor R, Revie C, Terzis S (2006) Question-based authentication using context data. In: Proceedings of the 4th Nordic Conference on Human-computer Interaction: Changing Roles. ACM, New York, NY, USA. pp 429–432
12. Das S, Hayashi E, Hong JI (2013) Exploring capturable everyday memory for autobiographical authentication. In: Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing. ACM, New York, NY, USA. pp 211–220
13. Zviran M, Haga WJ (1990) User authentication by cognitive passwords: an empirical assessment. In: Information Technology, 1990.'Next Decade in Information Technology', Proceedings of the 5th Jerusalem Conference on (Cat. No. 90TH0326-9). IEEE Computer Society Press, Los Alamitos, CA, USA. pp 137–144
14. Podd J, Bunnell J, Henderson R (1996) Cost-effective computer security: Cognitive and associative passwords. In: Proceedings of the 6th Australian Conference on Computer-Human Interaction (OZCHI '96). IEEE Computer Society, Washington, DC, USA. pp 304–305
15. Nosseir A, Terzis S (2010) A study in authentication via electronic personal history questions. In: Proceedings of the 12th International Conference on Enterprise Information Systems (ICEIS'10). HCI, Funchal, Madeira, Portugal Vol. 5. pp 63–70
16. Ullah A, Xiao H, Barker T, Lilley M (2014) Evaluating security and usability of profile based challenge questions authentication in online examinations. J Internet Serv Appl 5(1):2
17. Gupta P, Wee TK, Ramasubbu N, Lo D, Gao D, Balan RK (2012) Human: Creating memorable fingerprints of mobile users. In: Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference On. IEEE, Washington, DC, USA. pp 479–482
18. Xu K, Yao D, Pérez-Quinones MA, Link C, Scott Geller E (2014) Role-playing game for studying user behaviors in security: A case study on email secrecy. In: Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom), 2014 International Conference On. IEEE. pp 18–26
19. Choi BJ, Sun K, Choi S (2014) Cloud-based user authentication with Geo-temporal queries on smartphones. In: Proceedings of the 2nd International Workshop on Security in Cloud Computing. ACM, New York, NY, USA. pp 19–26
20. GSam Battery Monitor Application. https://play.google.com/store/apps/details?id=com.gsamlabs.bbm&hl=en
21. Ester M, Kriegel HP, Sander J, Xu X (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: KDD Vol. 96. pp 226–231
22. The Google Maps Geolocation API. https://developers.google.com/maps/documentation/business/geolocation/
23. The Haversine Formula. http://en.wikipedia.org/wiki/Haversine_formula
24. Supported Place Types in the Google Places API. https://developers.google.com/places/documentation/supported_types
25. Heiman GW (2010) Basic Statistics for the Behavioral Sciences. Cengage Learning, Belmont, CA, USA
26. Thorpe J, MacRae B, Salehi-Abari A (2013) Usability and security evaluation of GeoPass: a geographic location-password scheme. In: Proceedings of the Ninth Symposium on Usable Privacy and Security. ACM, New York, NY, USA. p 14
27. Hamilton LC (2012) Statistics with STATA: Version 12. Duxbury Press, Boston, MA, USA