INTEGRATION OF APP LOCK AND DETECTION OF MOBILE THEFT USING USER PATTERNS

K.Khaja Mohideen[1], S. Alagesan[2]

[1]Assistant Professor, Department of Information Technology, Aalim Muhammed Salegh College of Engineering,

[1]k.khajamohideen@aalimec.ac.in

[2]Assistant Professor, Department of Information Technology, Aalim Muhammed Salegh College of Engineering,

[2]s.alagesan@aalimec.ac.in

Abstract

Many web applications provide secret questions, to reset the account password when a users login fails. Most secret questions are blank fillings and are created based on the long-term knowledge of a users personal history that may not change over months/years. Today's prevalence of smartphone has granted us new opportunities to observe and understand how the personal data collected by smartphone sensors and apps can help create, personalize secret questions without violating the user's privacy concerns.In Existing system, a set of secret questions was created based on the user's personal history such as "What is your place of birth?" which can be easily identified .It may lead to poor security and reliability. A prototype is developed on Android smartphone and evaluates security of the secret questions.The proposed system, to reveal the secret questions related to calendar, last charged time, photo taken, contacts, call logs, SMS logs and app usage history which are memorable for users and are highly robust to attacks.In Modification of the project, this application will frame a set of standard questions along with the user runtime. If a mobile is stolen or lost, they either change the SIM card or retain the same and try to access the apps. If SIM card is changed then automatically GPS, Camera, Voice recorder are initiated, So that location and audio are sent as SMS to the alternative number and mail id of the user. If unauthorized tries to access the secret apps then, the system will query the user with four random questions (two from normal &two from user runtime).If not authorized, then GPS, camera, voice recorder is initiated and sent to the original user for tracking. This application helps user to identify the lost or stolen mobile by trackingmechanism

I.INTRODUCTION

Secret questions have been widely used by many web applications as the secondary authentication method for resetting the account password when the primary credential is lost. When creating an online account, a user may be required to choose a secret question from a pre-determined list provided by the server, and set answers accordingly. The user can reset his account password by providing the correct answers to the secret questions later.

For the ease of setting and memorizing the answers, most secret questions are blank-fillings and are created based on the long-term knowledge of a user's personal history that may not change over months/years (e.g., "What's the model of your first car?"). However this may lead to poor security and reliability

The "security" of a secret question depends on the validity of a hidden assumption: *A user's long-term personal history/information is only known by the user himself*. Moreover, a stranger can figure out the answers leaked from public user profiles in online social networks

The recent prevalence of smartphone has provided a rich source of the user's personal data related to the knowledge of his *short-term* history, i.e., the data collected by the smartphone sensors and apps. Is it feasible to use the knowledge of one's *short-term* personal history (typically within one month) for creating his secret question?

In this paper, we present a *Secret-Question based Authentication* system, called "Secret-QA", taking advantage of the data of smartphone sensors and apps without violating the user privacy. Meanwhile, Specifically,

- We design a user authentication system with a set of secret questions created based on the data of users' *short-term* smartphone usage.
- We evaluated the reliability and security of the three types of secret questions (blank-filling, true/false, and multiple-choice)
- We evaluate the usability of the system, and find that the Secret-QA system is easier to use

II. BACKGROUND AND RELATED WORK

Guessing attacks by acquaintance and stranger. the answers of 33% questions can be guessed by the "significant others" who were mainly participants' spouses (77%) and close friends (17%).

On the other hand, strangers can be more sophisticated than ever to launch the guessing attacks, as they can access the user's personal history through online social networks (OSN) . Therefore, the statistical guessing has become an effective way to compromise a few personal "secret" questions [5] (e.g., "Where were you born?", "What is the name of your high school?").

Recent proposals of user authentication systems. To reduce the vulnerability to guessing attacks we use user's dynamic Internet activities for creating his secret questions, namely network activities (e.g., browsing history), physical events (e.g., planned meetings, calendar items). They emphasized that frequently-changing secret questions will be difficult for attackers to guess the answers.

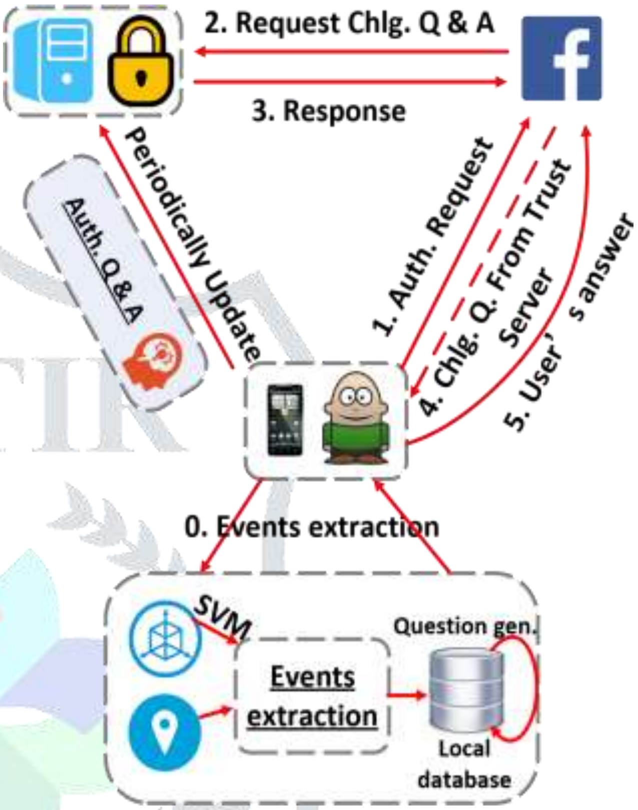

Fig. 1. System architecture of Secret-QA, for a typical user scenario of resetting the account password through answering the secret questions.

III.SYSTEM OVERVIEW

The Secret-QA system consists of two major components,namely the user-event extraction scheme and the challenge-response protocol

A. The User-event Extraction Scheme

Today's smartphones are equipped with sensors and apps which can capture various events related to a user's daily activities, e.g., the accelerometer can record the user's sports/motion status without consuming excessive battery

Secret-QA client app. we develop a Secret-QA client app called "Event Log" to extract the features for questiongeneration. the client app schedules the feature extraction

process periodically, and then features will be recorded in the local databases. Our extraction of user events are most lazily scheduled using Android Listener to save battery; meanwhile, we will pause the scheduling for some sensors after the screen is locked because no events can happen during screen-lock periods.

Secret-QA server. A trusted server is used as the auditor, which can also provide the user authentication service even if the phone is not available. As shown in block diagram of Fig 1, when authentication is needed, users' phone can generate questions with local sanitized data and send the answers/results (e.g., how many questions they answered correctly) to auditors via HTTPS channels.

B. A Three-phase Challenge Response Protocol

As shown in Figure 1 (from step 1 – 5), a service provider needs to authenticate the user's identity (typically for resetting the account password) through our trusted server.

TABLE I. Top ten categories of sensors/apps selected in Secret-QA.

1) GPS	2) Acc. (Accelerometer)
3) Calendar	4) Battery charging
5) Photo-taking	6) Contact
7) App installment	8) Call
9) SMS	10) App usage (mainly OSN apps)

The service prescribes three phases for authentication.

- Issue: the user issues an authentication request to the service provider (e.g., an OSN website, the step 1 in Figure 1), then the OSN website asks our trusted server for one or more encrypted secret questions and its answers; the questions are finally transferred to the user displaying on the smartphones (the step 2 – 3 in Figure 1).
- Challenge: the user provides answers to the challenge questions according to his/her short term memory, then sends it back to the OSN website (the step 4 in Figure 1).
- Authentication: the authentication is successful if the user's response conforms to the correct answers; otherwise, a potential attack is detected. If the times of authentication failure exceeds the threshold, our trusted server would deny to provide service for this particular user, as the in the last step in Figure 1.

IV. DESIGN OF CHALLENGE-RESPONSE PROTOCOL

We create three types of secret questions: A "True/false" question or "Yes/No" question ;a "multiple-choice" question or a "blank-filling" question that typically starts by "W" questions

We have two ways of creating questions in either a "Yes/No" or a "W" format: (1) a frequency based question like "Is someone (Who is) your most-frequent contact in last week?"; and (2) a non-frequency based one like "Did you (Who did you) call (Someone) last week?"

A. True/false Questions

Location (GPS) related questions. The example question related to GPS is No. 1 "Did you leave campus yesterday?". The GPS sensor captures the location information of the participants so that we could easily learn whether participants left campus far away enough with GPS coordinates recorded. It records within 500 metres range.

Motion activity (accelerometer) related questions. The example question related to accelerometer is No. 2 "Did you do running exercise for at least 10min with your phone carried yesterday?". There are many smartphone

No.	Question	Cat.
1	Did you leave campus yesterday?	GPS
2	Did you do running exercise for at least 10min yesterday?	Acc.
3	Is there an item for next week in your calendar?	Calendar
4	Did you charge your phone yesterday?	Charging
5	Did you take photos in the last three days?	Photo-taking
6	Is someone in your contact?	Contact
7	Did you install some app?	App install
8	Did you call someone last week?	Call
9	Did you call someone last two weeks?	
10	Did you call someone last month?	
11	Was someone your most frequent contact last week?	
12	Was someone your most frequent contact last two weeks?	
13	Was someone your most frequent contact last month?	
14	Did you text someone last week?	SMS
15	Did you text someone last two weeks?	
16	Did you text someone last month?	
17	Was someone your most frequent SMS contact last week?	
18	Was someone your most frequent SMS contact last two weeks?	
19	Was someone your most frequent SMS contact last month?	
20	Did you use some app last week?	App usage
21	Did you use some app last two weeks?	
22	Did you use some app last month?	
23	Was some app your most frequently used one last week?	
24	Was some app your most frequently used one last two weeks?	
25	Was some app your most frequently used one last month?	

applications that help users to monitor their running activities.

Smartphone usage (calendar, battery and camera) related questions. The questions derived from the calendar events is No. 3 “Is there an item planned for next week in your calendar?”..

B. Multiple-choice and Blank-filling Questions

We create “W” questions in the form of multiple-choice and blank-filling by simply extending the true/false questions “who did you call/text?” or “Which app did you use most frequently?”.

Answers to multiple-choice questions. For each question, there are four options (only one correct option). The correct option is randomly picked with an equal probability of being any options. For example, “Who did you call last week?” , we randomly pick a name in participant’s last week call records, and the rest three are faked by names in the contact We count the number of calls (or SMS) from/to every contact, or the number of times an app is used by a participant, for creating the frequency-based question, e.g., No. 34 “Who was your most frequent contact last week?”. If there are more than one most frequent contacts or most frequently used apps, any answer within these candidates is considered correct.

C. Definition and Thresholds of Determining A Good Question

A *good* secret question is defined as *easy-to-remember* and *hard-to-guess*, We set the threshold of easy-to-remember questions to be 80% for both true/false and multiplechoice questions

A random guessing attack has a success rate of 50% and 25% for true/false and multiplechoice (one of four options) questions, respectively. Then, we set the threshold of hard-to-guess questions to be no more than 55% (or 30%)—i.e., less than 55% (or 30%) attackers can correctly guess the answer, which is approximately to be a random guess for true/false (or multiple-choice) questions.

No.	Question	Cat.
1	Did you leave campus yesterday?	GPS
2	Did you do running exercise for at least 10min yesterday?	Acc.
3	Is there an item for next week in your calendar?	Calendar
4	Did you charge your phone yesterday?	Charging
5	Did you take photos in the last three days?	Photo-taking
6	Is someone in your contact?	Contact
7	Did you install some app?	App install.
8	Did you call someone last week?	Call
9	Did you call someone last two weeks?	
10	Did you call someone last month?	
11	Was someone your most frequent contact last week?	
12	Was someone your most frequent contact last two weeks?	
13	Was someone your most frequent contact last month?	
14	Did you text someone last week?	SMS
15	Did you text someone last two weeks?	
16	Did you text someone last month?	
17	Was someone your most frequent SMS contact last week?	
18	Was someone your most frequent SMS contact last two weeks?	
19	Was someone your most frequent SMS contact last month?	
20	Did you use some app last week?	App usage
21	Did you use some app last two weeks?	
22	Did you use some app last month?	
23	Was some app your most frequently used one last week?	
24	Was some app your most frequently used one last two weeks?	
25	Was some app your most frequently used one last month?	

TABLE II.True-false questions created in the experiment.Questions in boldface are *good* ones.

a questionnaire to indicate their experience of using OSNs, password recovery methods, and smartphones. Results show that many participants have less experience on smartphones' sensors; however, almost all students whose major is computer science are familiar with the concepts above. Hence, in our study, we use these groups of students as representatives of other populations for the following two reasons: (1) The scope of this work is to study whether using smartphone sensor/app data is helpful for secret-question based secondary authentication (2) Young people like students have the necessary experience on setting and answering secret questions (or completing this experiment), and they use smartphones and online tools (e.g. OSNs, search engines) every day.

Accordingly, participants in our experiment meet the following requirements.

1) Participants should be undergraduates or graduates , and should not be full-time employed

2) Participants should have used Android smartphone for at least one year;

V.EVALUATION AND EXPERIMENTSRESULTS

A. Experiment Setup

3) Participants should access at least onceper week to one of the well-known OSNs

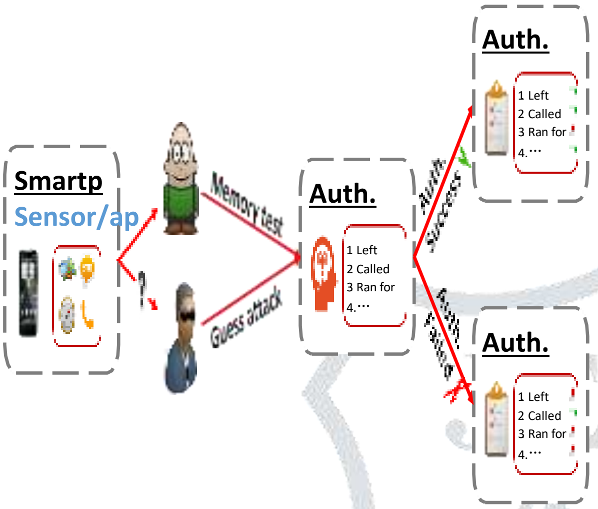

Fig. 2. Three-phase experiment procedure.

Only part of participants agreed to install it and proceed the experiments, and the participants are divided into two groups: Group *A* denoted the set of Android smartphone users that were willing to grant full permissions to install the client software; and Group *B* denoted the set of other participants that did not install the client software because of privacy concerns.

To eliminate hints and prevent collusion as much as possible during the tests, we enforced the following rules

- Each question could be answered only once via our custom built web-questionnaire interface

- Participants would not know the next question before finishing the current one.

- If a question would be asked in more than one type, then this question would appear in the order of "blank-filling", "multiple choice" and then "true/false".

B. Overall Experiment Results and Definition

Our results show that the secret questions related to motion sensors, calendar, app installment, and one question related to call have the best performance, most of which have a high reliability over 90%, while the success rate of guessing attacks is as low as that of a random guess.

1) Participant Recruitment: Many students in a university were recruited, Each participant was first asked

Experiment Modes. We have five different experiment modes: one in the memory test, and four under the threat models.

1) *MT* represents memory test, in which participants tried to recall answers of the questionnaires we generated;

2) *A(ON)* and *A(OFF)* represent the attacks from acquaintances with and without the help of online tools.

3) *S(ON)* and *S(OFF)* represent the attacks from strangers with and without help of online

tools.

C. True/false Questions

1) Performance of Sensor Data:

Location (GPS). We can conclude that participants can easily recall their location with a high accuracy rate of 91.7% in the memory test. However, its resiliance to attack under A(ON) and A(OFF) is low; for example, the percentage of A(OFF) is 83.3%, which implies a very high success rate of the acquaintance guessing attack. In the meanwhile, we observe that online tools provide little help when answering a question like “Did you leave campus yesterday?”.

As known, most of participants may turn off the GPS sensor to save battery life. Hence, it is not recommended for real-world deployment due to energy consumption. Alternatively, GPS can be replaced by a location based service using a Wi-Fi or cellular positioning system in the real-world deployment.

D. Multiple-choice Questions

Multiple-choice questions focus on the following five categories: contact, app installment, app usage, call and SMS.

Multiple-choice questions on contacts are vulnerable to attackers with online tools. Questions related to call and SMS are terribly vulnerable to acquaintance attacks

App installment and usage. Results show that the multiple-choice question related to the app installment data is both secure and reliable to serve as a good secret question. In contrast, the questions related to the app usage fail to maintain a high reliability or a low resilience to the guessing attack.

E. Authentication by Combining Multiple Lightweight Questions

However, it is feasible to combine multiple lightweight (e.g., true-false and/or multiple choice) questions sequentially to lower the success rate for an attacker. The reduction of attacker's success rate depends on how many lightweight questions we want to combine.

TABLE V. A combination of four lightweight questions.

No.	Question
2	Did you do running exercise for at least 10min yesterday?
3	Is an item planned for next week in your calendar?
7	Did you install some app?
27	Which app did you install in your phone? (multiple choice)

VI. DISCUSSION

Our results prove that questions related to motion sensors, calendar, app installment, and part of legacy app usage history (e.g., phone calls) have the best performance. Hence, we discuss the overarching issues related to real-world deployment and experimental details here.

A. Feedback on Battery Life

In experiment many participants complains that EventLog consumes much battery. We analyze the reasons blaming for high battery consumption as the following:

- The EventLog app requires a coarse-grained GPS service, and such a service consumes

excessive battery.

- The EventLog app polls every 30 seconds to track the app on-screen using Android API, and such behaviors demand a continuous workload of the CPU, and thereforethe battery runs out quickly.

The first challenge is also studied, in which WiFi (cellular) positioning system is adopted to replace GPS service to reduce the high battery overhead. As our approach only studies whether the user has left the campus, location information from WiFi or cellular positioning system is of sufficient accuracy. Researchers are free to add more location aware questions by adopting other positioning systems.

B. System Usability and Overhead

We further asked the participants to evaluate the system usability in the following three aspects.

1)
SecretQA The deployment cost. most users
smartpn considerclient app easy to install and
backend use on their nes, because our client app
operatio is mostly running Note that the
based au EventLog client app requires users' n
only for the client setting and the secret-
question
hentication.

2) Overhead. the overheads are acceptable for some users in our EventLog app with battery optimization. In future work, we will try to adopt WiFi or cellular location based service instead of GPS to further improve the battery life. Besides, the HTTPS traffic cost is almost negligible because our system will train and classify the motionrelated events locally.

3) Comparison with conventional secret-question based authentication schemes. When comparing, most users consider that it is easier to memorize the answers under Secret-QA and it has a better security against the guessing attacks due to the dynamic generation of the questions regarding to short-term user events.

C. Vulnerability to Statistical Guessing

The statistical guessing attack aims at identifying the most popular answers to each question and trying each one until no more guesses are allowed. Our research findings indicate that three categories of questions related to battery charging, photo-taking, and app usage, are statisticallyguessable

- Questions derived from battery charging data should be reconsidered, as the similar result of statistics and feedbacks implying that emerging adults are likely to charge their phones every day.
- It is possible for an attacker to crack questions like "Did you take any photos using camera in the last three days?" by just answering "yes" though, the answer may change due to demographic factors and users' behaviors.
- In terms of app usage, the top 10% popular apps can cover more than 50% of answers: a mobile client of OSN ranks first, with a percentage of 31.1%. Legacy apps come to the second place. The third, and the forth ones are the browsers and instant messaging apps.

VII. CONCLUSION

In this paper, we present a *Secret-Question based Authentication* system, called "Secret-QA", and conduct a user study to understand how much the personal data collected by smartphone sensors and apps can help improve the security of secret questions without violating the users' privacy. We create a set of questions based on the data related to sensors and apps, which reflect the users' short-term activities and smartphone usage. We measure the reliability of these questions by asking participants to answer these question, as well as launching the acquaintance/stranger guessing attacks with and without help of online tools. In our experiment, the secret questions related to motion sensors, calendar, app installment, and part of legacy apps (call) have the best performance in terms of memorability and the attack resilience

REFERENCES

1. R. Reeder and S. Schechter, "When the password doesn't work: Secondary authentication for websites," *S & P., IEEE*, vol. 9, no. 2, pp. 43–49,March 2011.
2. S. Schechter, C. Herley, and M. Mitzenmacher, "Popularity is everything: A new approach to protecting passwords from statistical-guessing attacks," in *USENIX Hot topics in security*, 2010, pp. 1–8.
3. J. C. Read and B. Cassidy, "Designing textual password systems for children," in *IDC.*, ser. IDC'12. New York, NY, USA: ACM, 2012, pp. 200–203.
4. H. Kim, J. Tang, and R. Anderson, "Social authentication: harder than it looks," in *Financial Cryptography and Data Security*. Springer, 2012,pp. 1–15.
5. S. Hemminki, P. Nurmi, and S. Tarkoma, "Accelerometer-based transportation mode detection on smartphones," in *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*, ser. SenSys '13. New York, NY,USA: ACM, 2013, pp. 13:1–13:14. [Online]. Available: http://doi.acm.org/10.1145/2517351.2517367
6. "libsvm on android," *GitHub*, 2015. [Online].Available: https://github.com/cnbuff410/Libsvm-androidjni
7. "Sensor event listener on android," *AndroidDeveloper*, 2015. [Online]. Available: http://developer.android.com/reference/android/hardware/SensorEventListener.html
8. J. Clark and P. van Oorschot, "Sok: Ssl and https: Revisiting past challenges and evaluating certificate trust model enhancements," in *Security and Privacy (SP), 2013 IEEE Symposium on*, May 2013, pp. 511–525.
9. "Android api reference about location criteria,"2013. [Online]. Available: http://developer.android.com/reference/android
10. M. Oner, J. A. Pulcifer-Stump, P. Seeling, and T. Kaya, "Towards the run and walk activity classification through step
11. detection-an android application," in *EMBC*. IEEE, 2012, pp. 1980–1983.

12. H. Falaki, R. Mahajan, S. Kandula, D. Lymberopoulos, R. Govindan, and D. Estrin, "Diversity in smartphone usage," in *MobiSys*. New York, NY, USA: ACM, 2010, pp. 179–194.

13. "Top 15 most popular social networking sites until march 2014," *eBizMBA*, 2013. [Online]. Available: http://www.ebizmba.com/articles/social-networking-websites

14. C. Wang, Q. Wang, K. Ren, and W. Lou, "Privacy-preserving public auditing for data storage security in cloud computing in INFOCOM,2010 proceedings IEEE, March 2010 pp.1-9

15. S. Yu, C. Wang, K. Ren, and W. Lou, "Achieving secure, scalable, and fine-grained data access control in cloud computing," in *INFOCOM, 2010 Proceedings IEEE*, March 2010, pp. 1–9.

16. R. Faragher and P. Duffett-Smith, "Measurements of the effects of multipath interference on timing accuracy in a cellular radio positioning system," *Radar, Sonar Navigation, IET*, vol. 4, no. 6, pp. 818–824, December 2010.

17. "Android service api introduction," *Google Android API*, 2014. [Online]. Available: http://developer.android.com/ reference/android/app/Service.html