Implicit passwords
using autobiographical memory recall

Under the mentorship of
Prof. Suneeta Agarwal

Group: CS-23

- Tuhin Subhra Patra 20164142
- Upamanyu Jamwal 20164169
- Rajat Dipta Biswas 20164114
- S Pranav Ganesh 20164098

Introduction

- **Goal**: Create an easy to use authentication system for users that has a high entropy and would be free from the risks of shoulder-surfing, screen-dumping and key-logging.

- Most password these days require users to remember a complicated and long alphanumeric string of characters and numbers which the human brain was not created to remember efficiently.

- This inevitably leads the user to store the password in written format somewhere for easy retrieval which defeats the purpose of a text-based single-use password.

Motivation

- Humans can remember and recall only 5 alphanumeric passwords on an average. This is due to the cognitive phenomenon called "Interference of Memory" [1], not due to ignorance.
- Memories of numbers and alphabets, which contain very limited information, are subject to the severe interference of memory which causes terrible confusion in what we remember.
- However, autobiographical memories hold emotions and are hence easier to remember for human beings. Tapping into people’s autobiographical or emotional memories can help create passwords that are more secure as well as easy to remember.

Explicit vs Implicit Passwords

Explicit

Passwords that explicitly match information from the server side to the client side to authenticate user fall under the explicit passwords umbrella. Most passwords in use today are all explicit passwords. There is a set way to authenticate the user and the user is expected to enter exactly what he/she created during the registration process.

Implicit

Implicit passwords, on the other hand, do not match the entered password explicitly. There might be different ways of authenticating the same information. Several approaches might be taken to verify if the right user is trying to access the system. Question need not to be asked explicitly for this method of authentication.

Problems with Explicit Passwords

Non-biometric

- Explicit passwords have problems such as being hard to remember, vulnerable to guessing, dictionary attack, key-logging, shoulder-surfing and social engineering. [2]

- A user may tend to choose a weak password or record his password.

Biometric

- The major problem of biometrics as an authentication scheme is the high cost of additional devices needed for identification process.
- The false-positive and false-negative rate may also be high if the devices are not robust.
- Biometric systems are vulnerable to replay attack.

Advantages of Implicit Passwords

Higher Entropy

- The number of possible uniques passwords is higher as there are more than one question.
- Some methods of authentication use “real matching” instead of actual matching. [3]
- e.g. picking a particular frame in a short video clip or choosing a particular point in a picture are inherently immune to brute force attacks as these have fine-grained options to choose.

Advantages of Implicit Passwords

Immune to Brute Force

- A conventional brute cannot be applied straightaway to this scheme as the answer to the login screen questions are not fixed but rather change with every try.
- The impersonator cannot simply try all possibilities of answering the questions.

Easy to Remember

- Implicit passwords are much easier to remember by users due to the fact that no exact sequence is to be remembered by the user.

Types of Passwords in Use Today

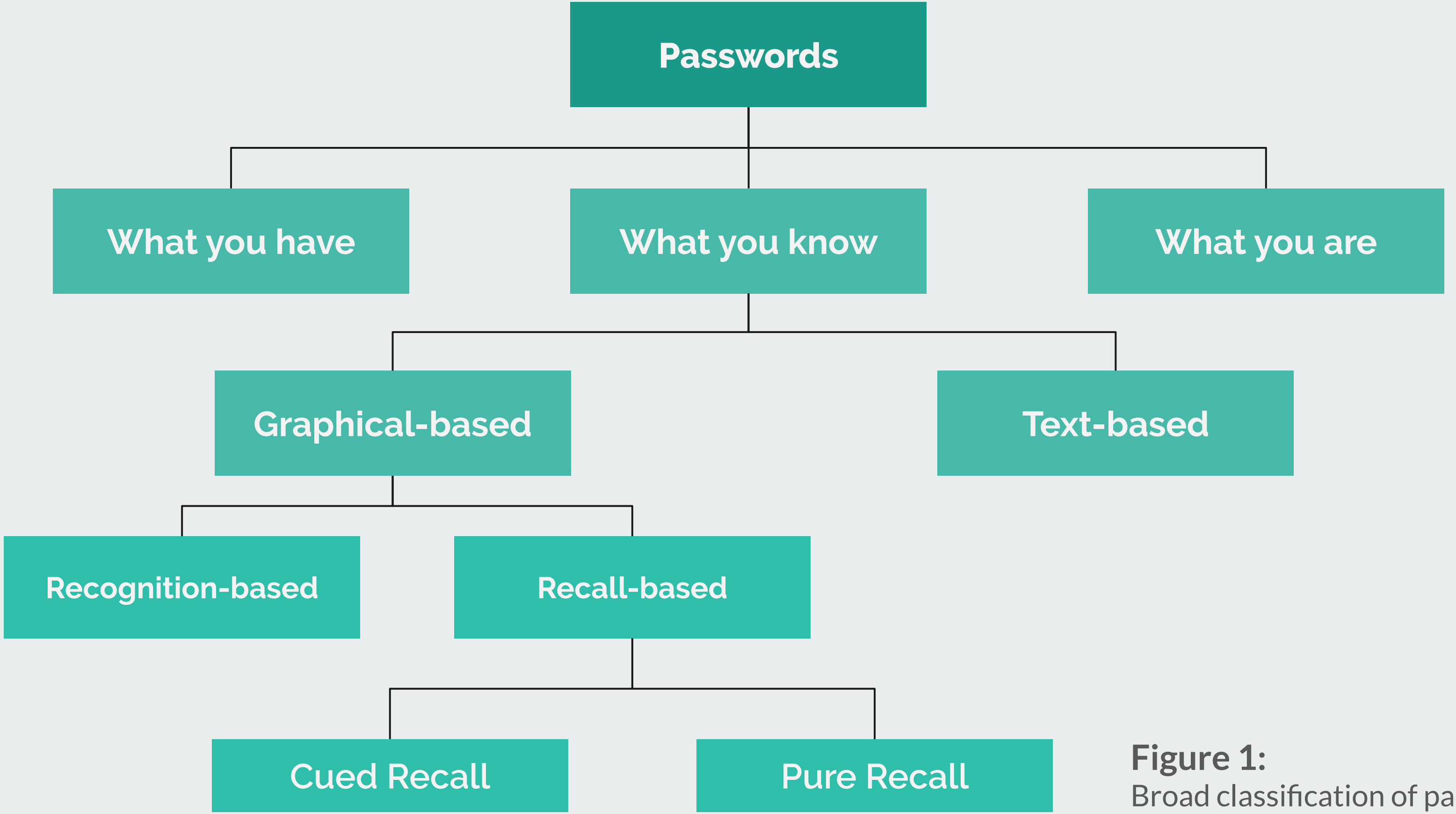


Figure 1:
Broad classification of passwords

Overview of the Authentication Process

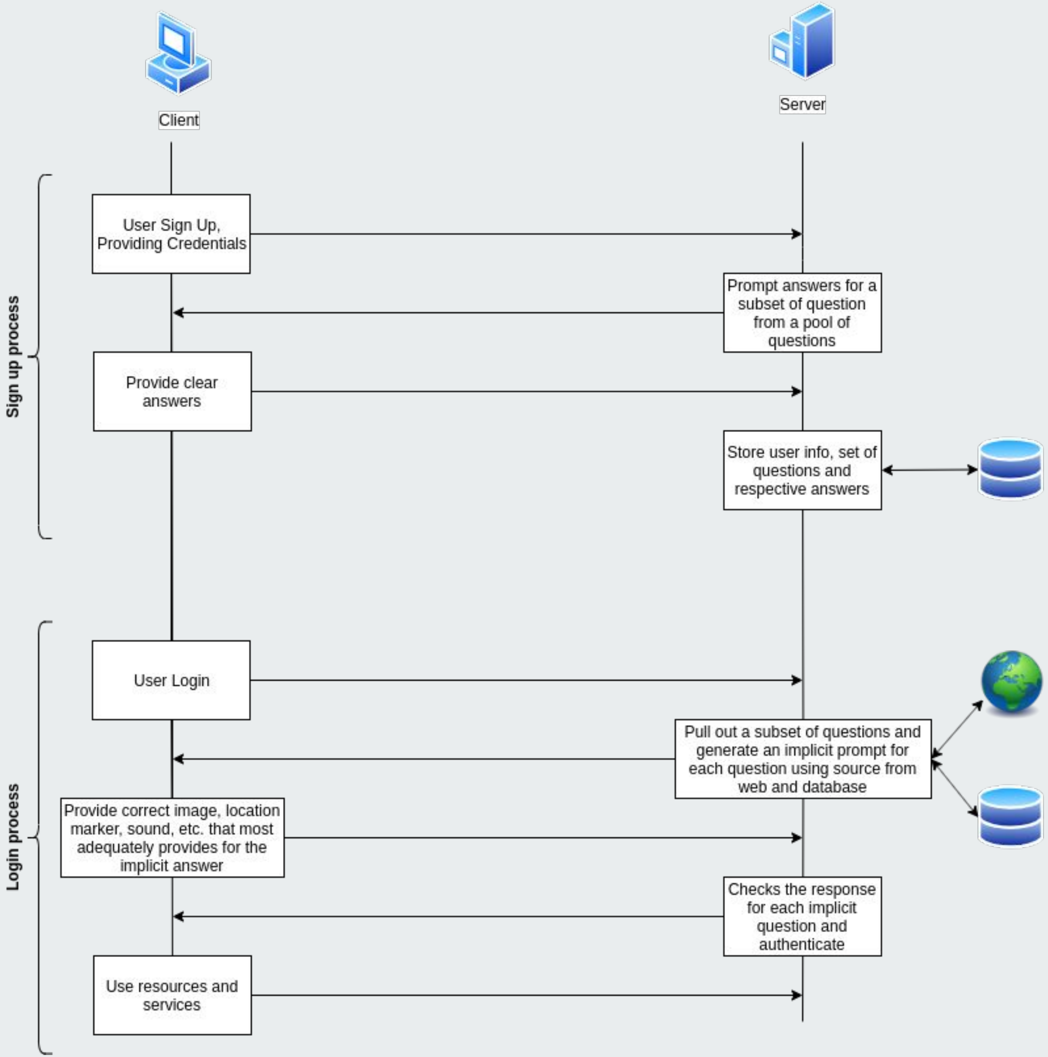

Client
Server
Sign up process
User Sign Up, Providing Credentials
Prompt answers for a subset of question from a pool of questions
Provide clear answers
Store user info, set of questions and respective answers
Login process
User Login
Pull out a subset of questions and generate an implicit prompt for each question using source from web and database
Provide correct image, location marker, sound, etc. that most adequately provides for the implicit answer
Checks the response for each implicit question and authenticate
Use resources and services

Figure 2:
The simplified flow diagram of the entire process excluding the methods employed in generation of implicit authentication questions.

Algorithm 1: **Naive Authentication Screen Generation**

```
Input Variables
    userId: Public key of user, input by user in step 1 of login
    noToEnquire: Number of questions to enquire during login
end Input Variables
procedure auth_Generate_and_Echo(userId, noToEnquire)
    currentAuthSpaces[] <- get_ImplicitPasswords_from_database(userId)
    shuffle(currentAuthSpaces);
    for (i := 1 to noToEnquire) {
        // currentAuthSpaces[i].waysToAsk are the total number of ways to ask
        // a particular authSpace
        r <- get_random_integer(1, currentAuthSpaces[i].waysToAsk);
        // ask in the rth specified format of authspace currentAuthSpaces[i].id
        echo_particular_question(currentAuthSpaces[i].id, r);
    }
```

Approach

- The central approach revolves around designing innovative questions that can have multiple answer spaces. This makes sure that there are multiple ways of extracting information implicitly, which was provided by the user during registration.
- Location information from maps and still images from videos can be used to extract information from the user as he places a marker on the map or stops a video at a certain timestamp. Image-recognition based systems are also incorporated to diversify the answer space.
- A group of images are displayed to the user and an accepted authentication requires a correct image being clicked.
- Traditional recognition-based systems are vulnerable to replay attack and mouse tracking because of the use of a fixed image as a password. However, by generating a different image set for new sessions and using image set with different tags but same implicit meaning, the situation can be improved.

Approach

- While generating the wrong options, proper care is taken so that wrong options are related to the correct option. Otherwise, the impersonator can pick the odd-one-out and guess the correct answer.
- One of the ways to ensure that we have some relation amongst all the options generated is to take a proper subset of keywords that are extracted from the answers and generate options based on them.
- Care is taken after generating the options, to check whether any incorrect option is present.
- If it is, we can fix them by extracting keywords from the generated option and ensuring that the new keywords are not a super-set of the set of keywords corresponding to the correct option.

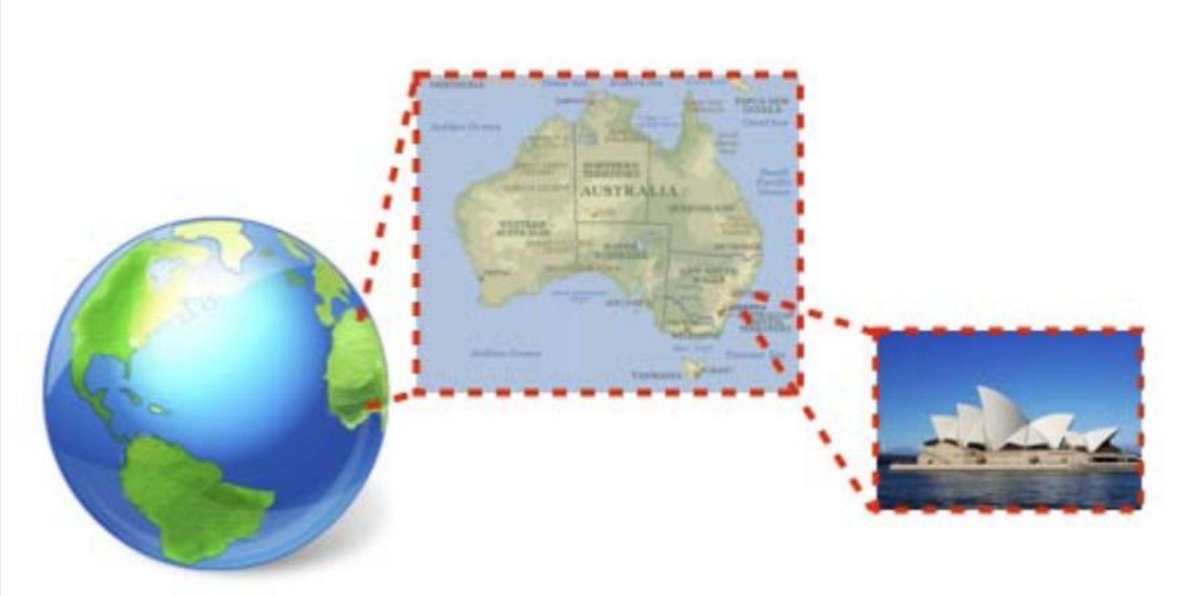


Figure 3:
Example of an Implicit Password Authentication System using marker on globe
(source: https://www.sciencedirect.com/science/article/pii/S0895717712001719)

Methods

Marker on Map

Question example:
“Favourite place to visit”, “Where was your mother born?”

Ask the user to locate a particular point on a map or a globe

Video Identification

Question example:
“Favourite pop-star or personality?”

Ask the user to stop a video at a certain timestamp, that has the frame of his favourite personality.

Image Identification

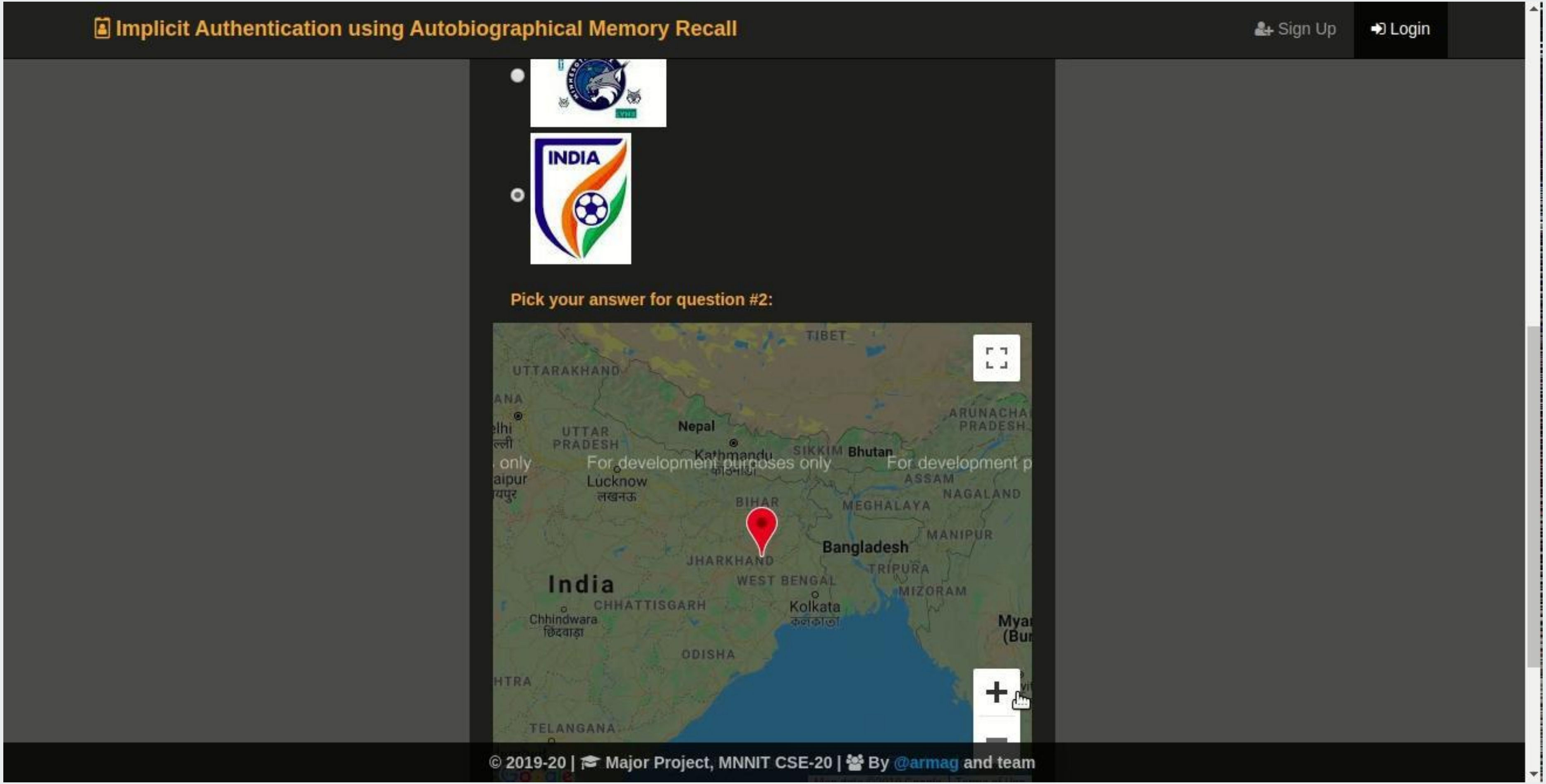
Question example:
“What is your favourite sports team?”

Provide the user with images. The user picks one out of them.

Disadvantages of Implicit passwords using autobiographical memory recall

- During registration, the user may provide answers to questions that are widely known about him/her.
- The generation of authentication spaces for each question is difficult and may need artificial intelligence or human intervention.
- Proposed system works fine with the bandwidth in modern day websites but for terminal based systems overhead is significant making text based method more feasible.
- Out of the multiple ways to enquire the same information, some ways are more secure.

Video Demo



Implicit Authentication using Autobiographical Memory Recall

Pick your answer for question #2:

© 2019-20 | Major Project, MNNIT CSE-20 | By @armag and team

The Web Implementation of the Project

1. For registration the user provides the answer to three particular questions. The user enters information that is personal to him and easy to remember. The user should not enter information that is public and known to people who the user is close to.

2. User will still require to remember things, but it will be easier since the user can relate to it. These are saved in the database and user during authentication.

3. During login, the user requests access to the system by entering his/her *username*. This may be sent as a plain text. If the user has not signed up, he would need to go through the sign up procedure.

The Web Implementation of the Project

4. The system chooses two questions from the ones answered by the user during the time of the registration.

5. For each question, the server may choose a random scenario from the authentication space that represents that question.

6. The server then asks these questions in an implicit manner masking the actual question asked during registration. Only the user can figure out the question from the options provided.

Applications

Platform that supports rich multimedia over the network

- Authentication in smartphone apps and file locks
- Web login in browsers

Platform that does not support rich multimedia

- As a fallback method for a text password (user has to use a different device once)
- As a password recovery option (user has to use a different device once)

Modules and Roles of Members

Research & Designing Authentication Spaces	**S Pranav Ganesh** ● Get information of present day implicit password schemes
Database Management and its Interaction with Backend	**Rajat D Biswas** ● Decide what and how to store in the database
Generation of Authentication spaces	**Tuhin S Patra** ● Ask questions in such a way that it is difficult for an impersonator but easy for the actual user
Frontend UI	**Upamanyu Jamwal** ● Decide on the flow of the user interaction

Conclusion

- The proposed system is successful in preventing dictionary attacks, key-logging, screen-dumping, guessing vulnerability, shoulder-surfing and social engineering .
- This could easily be incorporated into existing web server authentications since the bandwidth consumed in modern day websites is enough to support such a scheme including pictures and other multimedia with overhead being insignificant.
- The strength of implicit passwords lies in creating a good authentication space with a sufficiently large collection of images to avoid short repetitions. Compared to other methods our scheme may require human interaction and careful selection of images and “clickable” regions. It may also need user training. Once this is done, the password scheme can be more robust.

Future Work

- For field deployment the server must provide a pool of 50-100 question during sign-up out of which the user will answer 10-20 question. During sign-in 6-8 question are to be asked for authentication via some of the implicit methods available for each one of them.
- Other innovative methods of asking users questions can be devised and implemented. Finding new questions with multiple answer spaces is also rudimentary.

Future Work

- This password method would only be prevalent if developers adopt it. We will develop an easy to implement API that future developers can use. This will replace traditional text based passwords and make it easier for end-users.
- Authentication for mobile devices is very important these days. Hence apps for popular mobile operating systems need to be developed.
- Use artificial intelligence to help extract information about the video frame chosen or the keywords associated with an image to help automate the generation of more innovative and effective authentication spaces.

References

[1]T. Denning, K. Bowers, M. van Dijk, and A. Juels, Exploring implicit memory for painless password recovery, in Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, CHI ’11, New York, NY, USA, 2011, ACM, pp. 2615–2618.

[2]S. Almuairfi, P. Veeraraghavan, and N. Chilamkurti, A novel image-based implicit password authentication system (IPAS) for mobile and non-mobile devices, Mathematical and Computer Modelling, 58 (2013), pp. 108 – 116. Financial IT Security and 2010 International Symposium on Computational Electronics.

[3]C. Castelluccia, M. Duermuth, M. Golla, and F. Deniz, Towards Implicit Visual Memory-Based Authentication, in Network and Distributed System Security Symposium (NDSS), San Diego, United States, Feb. 2017, ISOC.

References

[4] S. Almuairfi, P. Veeraraghavan, and N. Chilamkurti, Implicit password authentication system, in 2011 IEEE Workshops of International Conference on Advanced Information Networking and Applications, March 2011, pp. 430-435.

[5] A. R. Gadekar and P. S. Shendekar, Implicit Password Authentication System, vol. 4, June 2013, pp. 77-81. https://pdfs.semanticscholar.org/6dfe/35e77773cf7b62c8fb0e3e021b4713373fda.pdf.

[6] G. Maps, Geocoding API, 2011. https://developers.google.com/maps/documentation/geocoding/intro.

[7] H. Gao, L. Ma, W. Jia, and F. Ye, Multiple password interference in graphical passwords, International Journal of Information and Computer Security, 5 (2012), pp. 11-27.

References

[8] P. D. Kulkarni, C. S. Satsangia, and S. Easo, Authentication system for banking using implicit password, in International Journal of Engineering Research and Development, vol. 3, August 2012, pp. 58–61. http://www.ijerd.com/paper/ vol3-issue1/I03015861.pdf.

[9] L. S. Packiam, Implicit pass point scheme for password authentication, Artificial Intelligent Systems and Machine Learning, 5 (2013).

[10] E. Shi, Y. Niu, M. Jakobsson, and R. Chow, Implicit authentication through learning user behavior, in Information Security, M. Burmester, G. Tsudik, S. Magliveras, and I. Ili ́c, eds., Berlin, Heidelberg, 2011, Springer Berlin Heidelberg, pp. 99– 113.

THANK **YOU**