Magdeburger Journal zur Sicherheitsforschung

Gegründet 2011 | ISSN: 2192-4260
Herausgegeben von Stefan Schumacher und Jörg Sambleben
Erschienen im Magdeburger Institut für Sicherheitsforschung

This article appears in the special edition »In Depth Security – Proceedings of the DeepSec Conferences«.
Edited by Stefan Schumacher and René Pfeiffer

Social Authentication

Vulnerabilities, Mitigations, and Redesign

Marco Lancini

High-value services have introduced two-factor authentication to prevent adversaries from compromising accounts using stolen credentials. Facebook has recently released a two-factor authentication mechanism, referred to as Social Authentication (SA). We designed and implemented an automated system able to break the SA, to demonstrate the feasibility of carrying out large-scale attacks against social authentication with minimal effort on behalf of an attacker. We then revisited the SA concept and propose reSA, a two-factor authentication scheme that can be easily solved by humans but is robust against face-recognition software.

Citation: Lancini, M. (2014). Social Authentication: Vulnerabilities, Mitigations, and Redesign. *Magdeburger Journal zur Sicherheitsforschung*, 2, 476–492. Retrieved November 13, 2014, from http://www.sicherheitsforschung-magdeburg.de/publikationen.html

Version 2014/11/13 14:01

Abstract

As social networks have become an integral part of online user activity, a massive amount of personal information is readily available to such services. In an effort to hinder malicious individuals from compromising user accounts, high-value services have introduced two-factor authentication to prevent adversaries from compromising accounts using stolen credentials. Facebook has recently released a two-factor authentication mechanism, referred to as *Social Authentication (SA)*, which requires users to identify some of their friends in randomly selected photos to be allowed access to their accounts.

In this work, we first study the attack surface of social authentication, showing how any attacker can obtain the information needed to solve the challenges presented by Facebook. We implement a proof-of-concept system that utilizes widely available face recognition software and cloud services, and evaluated it using real public data collected from Facebook. We have empirically calculated the probability of an attacker obtaining the information necessary to solve SA tests when relying on publicly accessible data as well as following a more active approach to gather restricted information, and we have then designed an automated attack able to break the SA, to demonstrate the feasibility of carrying out large-scale attacks against social authentication with minimal effort on behalf of an attacker.

We then revisited the Social Authentication concept and propose *reSA*, a two-factor authentication scheme that can be easily solved by humans but is robust against face-recognition software. Our core concept is to select photos in which state-of-the-art face-recognition software detects human faces, but cannot identify them due to certain characteristics. We implemented a web application that recreates the SA mechanism and conducted a user study that sheds light on user behavior regarding photo tagging, and demonstrated the strength of our approach against automated attacks.

1 Introduction

Online social networks (OSNs) have become some of the fastest growing Web services with a massive user base and, at the same time, an appealing target for malicious activities: Twitter reports over 140 million active users that send about 340 million tweets per day (Twitter no date), while Facebook reports over one billion monthly active users as of October 2012 (Zuckerberg no date), all the while encouraging its users to share more and more information online for a richer experience.

Consequently, OSNs have attracted the interest of the research community, which has striven to understand their structure and user interconnection (Krishnamurthy et al. 2008; Tang et al. 2009) as well as the interactions among users (Viswanath et al. 2009). Research has also focused on how OSN can be attacked or misused by malicious users. In fact, such accumulated data and the interconnections between users have made OSNs an attractive target for the Internet miscreants, for which OSNs became a lucrative platform for various types of attacks ranging from spam (Stringhini et al. 2010) to personalized phishing campaigns (Jacoby 2012). Studies (Shulman 2010) have shown that traditional underground economies have shifted their focus from stolen credit card numbers to compromised social network profiles, which are sold for the highest prices. A recent study (Gao et al. 2010) reports that the vast majority of spamming accounts in OSNs are not dummy profiles created by attackers, but legitimate, existing user accounts that have been compromised. Additionally, new Facebook phishing attacks use compromised accounts to steal personal information (Jacoby 2012).

As a standard method for strengthening the security of online user accounts, high-value services such as online banking, and recently Google services, have adopted two-factor authentication, where users must present two separate pieces of evidence in order to authenticate. The two factors are such that the risk of an adversary acquiring both is very low. Typically, the two factors consist of something the user knows (*e.g.*, a password) and something the user possesses (*e.g.*, a hardware token). Physical tokens, however, are inconvenient for users, who may not always carry them, and costly for the service that deploys them.

In 2011 Facebook, in an effort to combat stolen account passwords, introduced its so-called *Social Authentication (SA)*, a second authentication factor based on user-related social information that an adversary »half way around the world« supposedly lacks and cannot easily trick the owners into divulging. Following the standard password-based authentication, if Facebook deems it necessary, users are presented with photos of 7 of their friends and are asked to identify them. SA appears to be more user-friendly and practical as (i) users are required to identify photos of people they know and (ii) they are accustomed to tagging photos of their friends, thus implicitly providing the necessary labeled dataset for Facebook to generate challenges from.

A recent study (Kim et al. 2012), provided a formal analysis of the social authentication weaknesses against attacker within the victim's social circle. We expand the threat model and demonstrate in practice that any attacker, inside and outside the victim's social circle, can carry out automated attacks against the SA mechanism in an efficient manner. Therefore we argue that Facebook should reconsider its threat model and re-evaluate this security mechanism.

This work consists of two parts. In the first part (Section 3), we identify the vulnerable nature of SA and empirically confirm a series of weaknesses that enable an adversary to carry out an effective automated attack against Facebook's SA. The key of SA is the knowledge a user has about his online social circle, whereas an attacker trying to log into the account with

stolen credentials, lacks. Facebook acknowledges that its heuristics and threat model do not cover the case of friends and family (*i.e.,* anyone inside a user's online social circle) hacking into one's account.

The intuition behind our research is that any stranger who obtains a user's password can gain enough data to defeat the SA mechanism. To this end, we initially conduct a series of experiments to validate our assumptions about the access that an adversary might have to such information. The core of this study is the design and implementation of an automated, modular attack that defeats Facebook's SA mechanism. Initially, during a preparatory reconnaissance phase, the attacker obtain a victim's list of friends and the photos accessible from his OSN profile. This includes crawling the publicly-accessible portion of the victim's social graph and (optionally) performing actions that bring him inside the restricted part of the victim's social circle, such as issuing friendship requests to his friends. The attacker can then process the collected photos using face detection and recognition software to build each friend's facial model. An attacker is highly unlikely to be familiar with the friends of a victim—at least under the threat model assumed by Facebook—and there lies the security of recognizing one's friends as a security mechanism. However, by acquiring accurate facial models of a victim's friends he is in possession of the key to solving SA challenges. When the SA test is triggered, he can lookup the identity of the depicted friends and provide an answer.

At a first glance, it might seem that our attack only affects Facebook users that leave their friends list and published photos publicly accessible. According to Dey R. et al. (Dey et al. 2012) (2012), 47% of Facebook users leave their friends list accessible by default. However, an attacker can always attempt to befriend his victims, thus gaining access to their protected information. Such actions may achieve up to a 90% success rate (Bilge et al. 2009; Boshmaf et al. 2011; Nagle and Singh 2009; Ur and Ganapathy no date). That way, the set of vulnerable users may reach 84% of the Facebook population. At the same time, our experiments show that 71% of Facebook users expose at least one publicly-accessible photo album. Similarly, an attacker has very good chances of getting access, through online friendship requests, to profiles with private photo albums. Moreover, even if user A's photos are protected from public view and A does not accept friend requests from unknown people, user B might have a photo of A in which A is tagged (*i.e.,* their face framed and labeled with his real name and Facebook ID). If user B has their photos public, A's tags are implicitly exposed to crawling. Overall, dynamics of OSNs such as Facebook, make it very hard for users to control their data (Madejski et al. 2012; Staddon and Swerdlow 2011) and thereby increase the attack surface of threats against SA. We show that anyone can gain access to crucial information for at least 42% of the tagged friends used to build SA challenges that will protect a user's profile.

Under such minimal attack-surface assumptions, we manually verify that our implemented SA breaker, powered by a face recognition module, solves 22% of the real SA tests presented by Facebook (28 out of 127 tests), in less than 60 seconds for each test. Moreover, our attack gives a significant advantage to an attacker as it solves 70% of each test (5 out of 7 pages) for 56% of the remainder tests (71 out of 99 tests). Note that we obtain this accuracy in real-world conditions by relying solely on publicly-available information, which anyone can access: We do not send friendship requests to the victims or their friends to gain access to more photos. Furthermore, our simulations demonstrate that within a maximized attack surface (*i.e.,* if a victim, or one of his friends, accepts befriend requests from an attacker, which happens in up to 90% of the cases), the success rate of our attack increases to 100%, with as little as 120 faces per victim for training, and takes about 100 seconds per test.

In the second part of our study (Section 4) we present *reSA* (short for *»Social Authentication, Revisited,«*), a design of a secure yet usable SA mechanism for social networks. *reSA* is a two-factor authentication scheme that can be easily solved by humans but is robust against face-recognition software.

Given that we have demonstrated that standard SA tests are broken (Section 3), our core concept is to select photos of poor quality, in which state-of-the-art face-recognition software detects human faces, but cannot identify them due to certain characteristics (*e.g.,* strange angles or lighting). We designed a web application that simulates the SA mechanism and we carried out a user study where we asked humans to solve SA tests with photos of mixed quality. The outcome of this user study shows that people are able to recognize their friends just as good in both standard SA tests and tests with photos of poor quality (*e.g.,* face partially visible or unrecognizable).

2 Security of Social Authentication

The challenge for consumer-facing websites is to balance strong security with usability. Indeed, complicated security schemes will not achieve widespread adoption among users.

A new, emerging, approach consists in authenticating users via a method called *Social Authentication*, a type of two-factor authentication scheme that tests the user's personal social knowledge, and that only the intended user is likely to be able to answer. In particular, a so called »social CAPTCHA« is presented to authenticate a member of the web service (note that this mechanism is particularly adequate for social networks). The social CAPTCHA includes one or more challenge questions based on information available in the social network, such as the user's activities and/or connections in the social network. The social information selected for the social CAPTCHA may be determined based on affinity scores associated with the member's connections, so that the challenge question relates to information that the user is more

likely to be familiar with. A degree of difficulty of challenge questions may be determined and used for selecting the CAPTCHA based on a degree of suspicion. This approach eliminates the key issues of traditional CAPTCHAs, which are (at times) incredibly hard to decipher and, since they are only meant to defend against attacks by computers, vulnerable to human hackers. Indeed, a common type of CAPTCHA requires the user to type letters or numbers from a distorted image that is difficult for a computing algorithm to interpret but relatively easy for a human. Requiring a user to read distorted text for authentication prevents automatic systems from connecting to a website without user intervention. Moreover, existing CAPTCHA mechanisms can be defeated by a practice known as »CAPTCHA farming,« wherein an automated algorithm temporarily diverts the CAPTCHA question to a human user to solve the CAPTCHA question and then returns to its illegitimate purpose. If cheap human labor can be utilized, the existing CAPTCHA mechanisms can be rendered completely ineffective.

2.1 Facebook's Social Authentication

Facebook's Social Authentication, for which Facebook obtained a patent in September 2010 (Shepard; Jonathan et al. 2010), was announced in January 2011 (Facebook 2011a,b), and in June 2012 landed also on the mobile version of the website (Facebook 2012). To the best of our knowledge it is the first instance of a two-factor authentication scheme based on the »who you know« rationale: A user's credentials are considered authentic only if the user can correctly identify his friends.

The idea that underlies this mechanism is that the user can recognize his friends whereas a stranger cannot: Attackers halfway across the world might know a user's password, but they don't know who his friends are. Therefore, the assumption is that nobody but the actual user will possess the necessary social information to correctly pass the test. Actually, Facebook's SA is not meant to substitute a strong second factor of authentication. Instead, it is meant to be a weak form of second factor of authentication to block large-scale abuses of credentials stolen through phishing attacks (*e.g.*, casual attackers).

How Social Authentication Works SA is activated only when Facebook's security heuristics classify a login attempt as suspicious, for instance when taking place from a country or computer for the first time. Instead of showing a traditional CAPTCHA, Facebook shows the user a few pictures of his friends and asks him to name the person in those photos. More precisely, right after the standard, password-based authentication, the user is presented with a sequence of 7 pages featuring authentication challenges. As shown in Fig. 1, each challenge is comprised of 3 photos of an online friend plus a multiple-choice list of the names of 6 people from the user's social circle (*i.e.*, »suggestions«), from which he has to select the one depicted. The user is allowed to fail in 2 challenges, or skip them, but must correctly identify the people in at least 5 challenges out of 7 to pass the SA test.

Advantages and Shortcomings The major difference from the traditional two-factor authentication mechanisms (*e.g.*, confirmation codes sent via text message or hardware tokens) is that Facebook's SA is less cumbersome, especially because users have grown accustomed to tagging friends in photos. However, as presented recently by Kim et al. (Kim et al. 2012), designing a usable yet secure SA scheme is difficult in tightly-connected social graphs, not necessarily small in size, such as university networks. It is infact hard to identify the social knowledge that a user holds privately since social knowledge is inherently shared with others: many likely attackers are »insiders« in that the people who most want to intrude on your privacy are likely to be in your circle of friends.

The experimental evaluation we carried out in Section 3.3 suggests that SA carries additional implementation drawbacks. First of all, the number of friends can influence the applicability and the usability of SA. In particular, users with many friends may find it difficult to identify them, especially when there are loose or no actual relationships with such friends. A typical case is a celebrity or a public figure. Even normal users, with 190 friends on average (Facebook 2011c), might be unable to identify photos of online contacts that they do not interact with regularly. Dunbar's number (Dunbar 1998) suggests that humans can maintain a stable social relationship with at most 150 people. This limit indicates a potential obstacle in the usability of the current SA implementation, and should be taken into account in future designs.

Another parameter that influences the usability of SA is the number of photos that depict the actual user, or at least that contain objects that uniquely identify the particular user. As a matter of fact, feedback (Jacoby 2012) from users clearly expresses their frustration when challenged by Facebook to identify inanimate objects that they or their friends have erroneously tagged for fun or as part of a contest which required them to do so.

These findings have led us to demonstrate, with an automated attack, the level of risk due to the current implementation of Facebook's SA.

3 Breaking Social Authentication

We conducted experiments (as detailed in Section 3.3) were we manually inspected a set of SA challenges in order to determine the presence (or absence) of human faces in the presented photos. These experiments reveals that about 80% of the photos found in SA tests contain at least one face that can be detected by face-detection software. This rationale makes us argue that an automated system can successfully pass

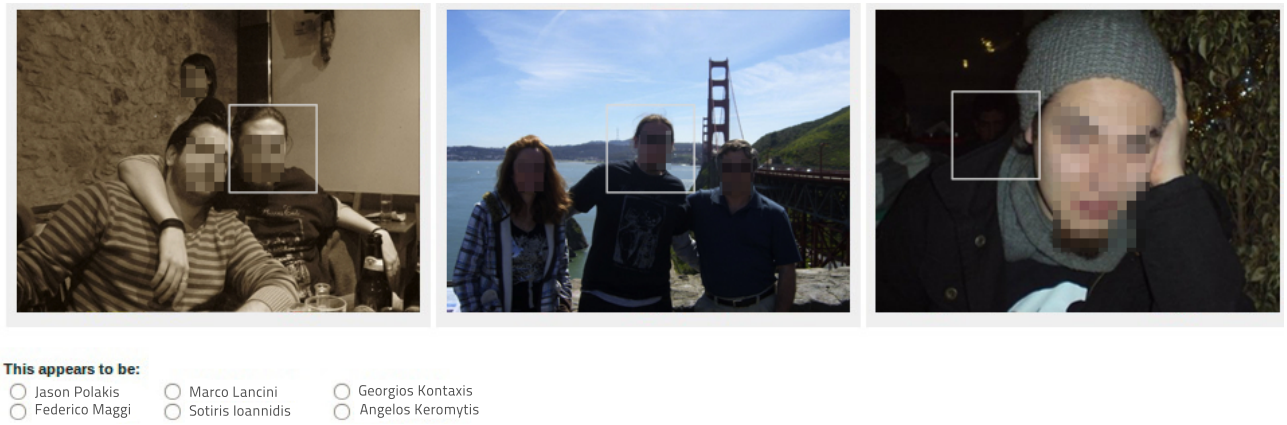


Figure 1: Example screenshot of the user interface of a Facebook SA page.

the SA mechanism. As a matter of fact, we argue that any stranger (*i.e.*, anyone not in a user's online social circle) can position himself inside the victim's social circle, thereby gaining the information necessary to defeat the SA mechanism automatically.

3.1 Threat Analysis

In this work, we refer to the people inside a user's online social circle as *friends*. Friends have access to information used by the SA mechanism. Tightly-connected social circles where a user's friends are also friends with each other are the worst scenarios for SA, as potentially any member has enough information to solve the SA for any other user in the circle. However, Facebook designed SA as a protection mechanism against strangers, who have access to none or very little information. Under this threat model, strangers are unlikely to be able to solve an SA test.

3.1.1 Attacker Models

In our attack model, the attacker has compromised the user's credentials. This can be accomplished in many ways (*e.g.*, phishing, trojan horses, key logging, social engineering) depending on the adversary's skills and determination (Dhamija et al. 2006). Note that this is not an unreasonable assumption, as it is actually the reason behind the deployment of the SA.

We then distinguish between two attacker models, a casual and a determined attacker.

Casual Attacker A *casual attacker* is interested in compromising the greatest possible number of accounts, without focusing on some particular user. This type of attacker leverages publicly-accessible information from a victim's social graph, and therefore may lack some information (*e.g.*, the victims may expose no photos to the public, there are no usable photos, no friend requests issued) and have limited access to the data needed for training a face recognition system.

Determined Attacker A *determined attacker* is more focused on a particular target and so he actively attempts to gather additional private information by infiltrating the victim's social graph through friendship requests addressed to the target himself and/or to his friends. This approach allows the attacker to have access to the majority of the victims' photos, have a better dataset for create facial models, and, accordingly, to obtain better results breaking a SA challenge.

3.1.2 Attack Surface Estimation

To assess the risk behind SA we estimated the probabilities that an attacker has to collect the information needed to carry out an attack against it. In other words, if an attacker has obtained the credentials of any Facebook user, what is the probability that he will be able to access the account? What is the probability if he also employs friend requests to access non-public information on profiles? To derive the portion of users susceptible to this threat, we built the attack tree depicted in Figure 2 as follow.

Friends list Initially, any attacker requires access to the victim's friends list. According to Dey et al. (Dey et al. 2012) $\mathrm{P}(F) = 47\%$ [1] of the user's have their friends list public (as of March 2012). If that is not the case, a determined attacker can try to befriend his victim. Studies have shown (Bilge et al. 2009; Boshmaf et al. 2011; Nagle and Singh 2009; Ur and Ganapathy no date) that a very large fraction of users tends to accept friend requests and have reported percentages with a 60-90% chance of succeeding (in our analysis we use 70%, lower than what the most recent studies report). Therefore, he has a combined 84% chance of success so far, versus 47% for the casual attacker.

Photos Ideally the attacker gains access to all the photos of all the friends of a victim. Then with a probability of 1 he can solve any SA test. In reality, he is able to access only a subset of the photos from all or a

1 From hereinafter we use $\mathrm{P}(E)$ to indicate the estimator of the probability of event E. We use the empirical frequency as the estimator.

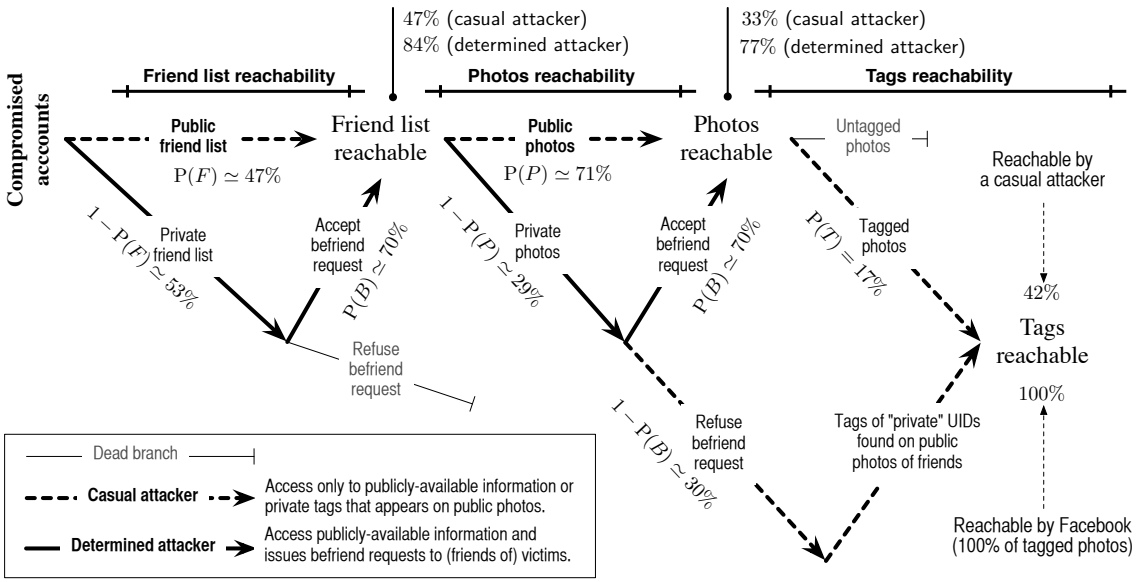


Figure 2: Attack tree to estimate the vulnerable Facebook population. Not all the branches are complete, as we consider only the events that are relevant to the case study.

subset of the friends of a victim. Our study of $236,752$ Facebook users revealed that $P(P) = 71\%$ of them exposed at least one public photo album. Again we assume that a determined attacker can try to befriend the friends of his victim to gain access to their private photos with a chance of $P(B) \simeq 70\%$ to succeed, which is a conservative average compared to previous studies. At the end of this step, the determined attacker has on average at least one photo for 77% of the friends of his victim while a casual attacker has that for 33%. This is versus Facebook, which has that for 100% of the friends with uploaded photos.

Tags The next step is to extract labeled frames (tags) of people's faces from the above set of photos to compile $\langle \texttt{uid}, \texttt{face} \rangle$ tuples used by Facebook to generate SA tests and by the attacker to train facial models so as to respond to those tests. By analyzing $16,141,426$ photos from out dataset, corresponding to the 33% of friends' photos for the casual attacker, we found that 17% of these photos contain tags (hence usable for generating SA tests), yet only the 3% contain tags about the owner of the photo. This means that by crawling a profile and accessing its photos it is more likely to get tags of friends of that profile than of that profile itself. The astute reader notices that Facebook also has to focus on that 17% of photos containing tags to generate SA tests: Facebook will utilize the 17% containing tags of all the photos uploaded by a user's friends and therefore generate SA tests based on 100% of the friends for whom tags are available, whereas an attacker usually has access to less than that. In the extreme case, having access to a single friend who has tagged photos of all the other friends of the target user (*e.g.*, he is the »photographer« of the group), the attacker will acquire at least one tag of each friend of the user and will be able to train a face recognition system for 100% of the subjects that might appear in an SA test. In practice, by collecting the tags from the photos in our dataset we were able to gather $\langle \texttt{uid}, \texttt{face} \rangle$ tuples for 42% of the people in the friend lists of the respective users. Therefore, assuming that all of a user's friends have tagged photos of them on Facebook, a casual attacker is able to acquire this sensitive information for 42% of the tagged friends used by Facebook to generate SA tests. As we show in Section 3.3.2, with only that amount of data, we manage to automatically solve 22% of the real SA tests presented to us by Facebook, and gain a significant advantage for an additional 56% with answers to more than half the parts of each test. We cannot calculate the corresponding percentage for the determined attacker without crawling private photos. However, we simulate this scenario in Section 3.3.3 and find that we are able to pass the SA tests on average with as little as 10 faces per friend.

Faces Finally, from the tagged photos, the attacker has to keep those that actually feature a human face and discard any photos that do not contain any tag information as they are of no use for building a dataset of labeled faces. We found that 80% of the tagged photos in our dataset contain human faces that can be detected by face-detection software, and Facebook seems to follow the same practice; therefore, the advantage for either side is equal.

Overall, our initial investigation reveals that up to 84% of Facebook users are exposed to the crawling of their friends and their photos. They are, thus, exposed to attacks against the information used to protect them through the SA mechanism. A casual attacker can access $\langle \texttt{uid}, \texttt{face} \rangle$ tuples of at least 42% of the tagged friends used to generate social authentication tests for a given user. Such information is considered sensitive, known only to the user and the

user's circle, and its secrecy provides the strength to this mechanism.

3.2 System Overview

To prove our hypothesis, we built an automated system that can carry on the attack in an automated fashion. The attack consists of four preparation steps (Steps 1-4), which the attacker runs offline, and one execution step (Step 5), which the attacker runs in real-time when presented with the SA test. Figure 3 presents an overview of the attack.

Step 1: Crawling Friend List Given the victim's *UID* [2], a crawler module retrieves the UIDs and names of the victim's friends and inserts them in the system's database.

As explained in Section 3.1.2, casual attackers can access the friend list when this is publicly available (47% of the users), whereas determined attackers can reach about 84% of the friend lists by issuing befriend requests to their victims.

Step 2: Issuing Friend Requests A determined attacker can use social-engineering techniques to obtain more informations than those publicly available. He can use legitimate-looking, dummy profiles (*i.e., fake accounts*) to send friendship requests to all of the victim's friends. As shown in Figure 2, this step can expand the attack surface by greatly increasing the number of photos that will be reachable in Step 3. Indeed, while only a small portion of people let their album freely accessible also by strangers (*i.e.,* non-friends), almost all the users of Facebook keeps the default privacy settings regarding photo albums, that make them accessible only to their friends.

Step 3: Photo Collection The URLs of all the photos contained in the albums of the target's friends are collected using the same approach described in Step 1. The collected URLs are then processed by a module that performs the actual download of the photos, which are stored, together with their metadata (URL, UID of the owner, tags and their coordinates), into a distributed filesystem.

Step 4: Modeling Each downloaded photo that comes from the previous phase is then analyzed to find faces, each of which is subsequently labeled with the UID of its nearest tag. To avoid association errors, a face is matched to a tag only if the euclidean distance between the face's center and the tag's center turns out to be lower than a given threshold. Unlabeled faces and tags with no face are useless, thus they are discarded. This labeled dataset contains, for each friend of the victim, a set of his faces, which are normalized and used to create a facial model that, in turn, is used to train a face-recognition classifier.

For the modeling phase we can rely on two different approaches: a *custom solution* (based on the `OpenCV`[3] toolkit), which is more versatile toward the selection of algorithm parameters; and a *cloud-based service* (namely face.com), which offers better accuracy for the face-recognition task.

Step 5: Name Lookup After completing the preparation steps (1-4), an attacker can proceed with the actual attack. When Facebook challenges the system with a SA test, the system extracts all the significant information (*i.e.,* photos and suggested names) and then proceed to solve each challenge. The 3 photos belonging to a SA page are analyzed by a face detection classifier that identifies all the faces contained in them, which are then processed to extract their principal components. Those components are submitted to the classifier previously created, which attempts to identify the depicted person and select the correct name. This process is repeated for each one of the 7 pages of a SA test.

3.3 Experimental Evaluation

Here we evaluate the nature of Facebook's SA mechanism and our efforts to build an automated attack against Facebook's SA.

We first assess the quality of our dataset of Facebook users, which we consider a representative sample of the population of the online social network. Next, we evaluate the accuracy and efficiency of our attack.

3.3.1 Overall Dataset

Our dataset contains data about real Facebook users, including their UIDs, photos, tags, and friendship relationships, as summarized in Table 1. Note that we have not attempted to compromise or otherwise damage the users or their accounts and that we collected our dataset as a *casual attacker* would do.

Through public crawling and issuing friendship requests from fake profiles (steps 1, 2 and 3 in Fig. 3) we collected data regarding $236,752$ distinct Facebook users. 71% $(167,359)$ of them have at least one publicly-accessible album and thus we refer to these users as public UIDs (or *public users*). The remaining 29% of UIDs $(69,393)$ keep their albums private and we refer to them as *private users*). We found that 38% of the private users $(26,232$ or 11% of the total users) are still reachable because their friends have tagged them in one of the photos in their own profile (to which we have access). We refer to these UIDs as semi-public UIDs (or *semi-public users*). Data about the remaining 62% of UIDs $(43,161$ or 18% of the total users) is not obtainable because these users keep their albums private, and their faces are not found in any of the public photos of their friends.

The public UIDs lead us to $805,930$ public albums,

2 A UID is a unique string assigned to each user of Facebook

3 http://opencv.org/

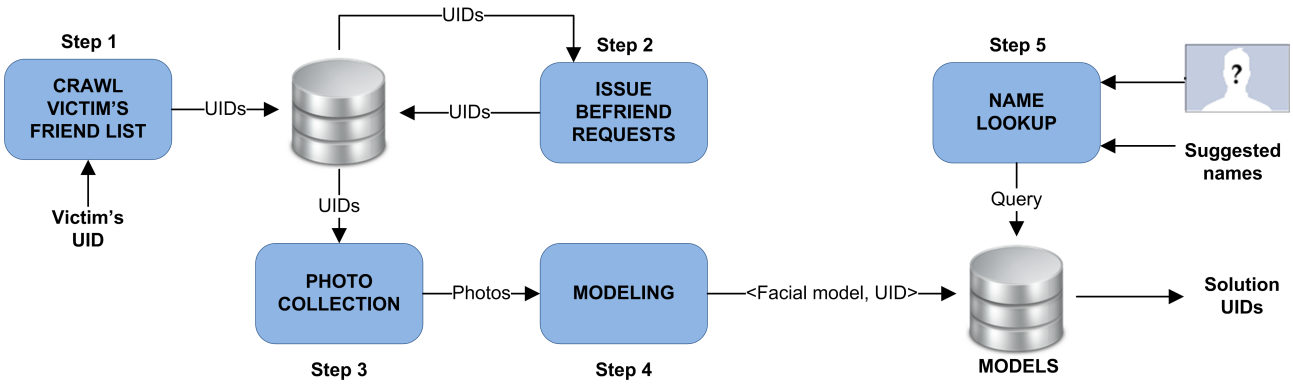


Figure 3: Overview of our automated attack. It consists in five steps. In Step 1 we retrieve the victim's friend list using his or her UID. Then, in Step 2 (optional), we send befriend requests, so that we have more photos to collect in Step 3 and to extract faces from and build face classifiers in Step 4. In Step 5, given a photo, we query the models to retrieve the corresponding UID and thus match a name to a face.

	TOTAL	PUBLIC	PRIVATE
UIDs	236,752	167,359	69,393
Not tagged	116,164	73,003	43,161
Tagged	120,588	94,356	26,232
Mean tags per UID:		19.39	10.58
Tags	2,107,032	1,829,485	277,547
Photos	16,141,426	16,141,426	(not collected)
Albums	805,930	805,930	(not collected)

Table 1: Summary of the collected dataset.

totaling $16,141,426$ photos and $2,107,032$ tags[4] that point to $1,877,726$ distinct UIDs. It is therefore evident that people exposing (or making otherwise available) their photos are not only revealing information about themselves but also about their friends. This presents a subtle threat against these friends who cannot control the leakage of their names and faces. Albeit this dataset only covers a very small portion of the immense Facebook user base, we consider it adequate enough to carry out thorough evaluation experiments and validate our approach.

Social Authentication Tests From our manual inspection of 127 instances of real SA tests ($2,667$ photos), we have noticed that Facebook's selection process is quite precise, despite some inaccuracies that lead to SA tests where some photos contain no face. Overall, 84% of these $2,667$ photos contained at least one human-recognizable face, and about 80% of them contained at least one face such that an advanced face detection software can discern (in this test, we used face.com).

To validate our argument on the use of face detection filtering, we repeated the same manual inspection on a different set of $3,486$ photos drawn at random from our dataset of $16,141,426$ photos. We then cropped these images around the tags; hence, we generated a SA dataset in the same manner that Facebook would if it naively relied only on people's tagging activity. Only 69% ($< 84\%$) of these photos contain at least one recognizable human face, thus the baseline number of faces per tag is lower in general than in the photos found in the real SA tests. This confirms our hypothesis that Facebook employs filtering procedures to make sure each SA test page shows the face of the person in question in at least one photo.

3.3.2 Breaking SA: Casual Attacker

In the following experiment we assume the role of a casual attacker, with limited access to tag data for the training of a face recognition system. At the same time we attempt to solve real Facebook SA tests using the following methodology.

We have created 11 dummy accounts that play the role of victims in our experimental scenario, where we assumed the role of the attacker. In this scenario, the attacker knows the password for the accounts, but lacks the social information to solve the SA challenges presented by Facebook. As a matter of fact, we did actually lack the social information even though we owned the victim accounts, as the friends were random strangers which we had befriended.

Then, we employ a graphical Web browser scripted

4 On 11 April 2012, our crawler had collected $2,107,032$ of such tags, although the crawler's queue contains $7,714,548$ distinct tags.

via Selenium[5] to log into these accounts in an automated fashion. To trigger the SA mechanism we employ Tor[6], which allows us to take advantage of the geographic dispersion of its exit nodes, thus appearing to be logging in from remote location in a very short time. By periodically selecting a different exit node, as well as modifying our user-agent identifier, we can arbitrarily trigger the SA mechanism. Once we are presented with an SA test, we iterate its pages and download the presented photos and suggested names, essentially taking a snapshot of the test for our experiments. We are then able to take the same test offline as many times necessary. Note that this is done for evaluation purposes and that the same system in production would take the test once and online.

The gathered dataset (summarized in Table 2) is composed as follows:

Testing Dataset With the aforementioned procedure we collected 127 distinct SA tests, comprising 7 pages that incorporate 3 distinct, tagged photos (of the same victim) and 6 suggested names, totaling $2,667$ tagged photos and $5,335$ (684 distinct) suggested names. We map these names to the corresponding UIDs at the time of collection, as sometimes people change their screen name on Facebook.

Training Dataset From our dataset, we extracted the photos and associated tag information of the $1,131$ distinct UIDs of the users that are friends with the aforementioned 11 fake profiles, and thus that are likely to contain labeled faces to train our classifier. This selection lead us to $17,808$ distinct photos.

We then tried breaking the real SA tests using face.com (*i.e.*, the *existing cloud-based service* of Step 4). Note that we manually inspected all the outcomes proposed by the module by showing to a volunteer a selection of photos of the Facebook UID guessed by our attack, so to be sure about the correctness of the answer. Figure 4 presents the outcome of the tests. Overall we are able to solve 22% of the tests ($28/127$) with people recognized in 5-7 of the 7 test pages and significantly improve the power of an attacker for 56% of the tests ($71/127$) where people were recognized in 3-4 of the 7 test pages. At the same time, it took 44 seconds on average with a standard deviation of 4 seconds to process the photos for a complete test (21 photos). Note that the time allowed by Facebook is 300 seconds.

We further analyzed the photos from the pages of the SA tests that failed to produce any recognized individual. In about 25% of the photos face.com was unable to detect a human face. We manually inspected these photos and confirmed that either a human was shown without his face being clearly visible or no human was present at all. We argue that humans will also have a hard time recognizing these individuals unless they are very close to them so that they can identify them by their clothes, posture or the event. Moreover, in 50% of the photos face.com was able to detect a human face but marked it as unrecognizable. This indicates that it is either a poor quality photo (*e.g.*, low light conditions, blurred) or the subject is wearing sunglasses or is turned away from the camera. Finally, in the last 25% of the photos a face was detected but did not match any of the faces in our training set. Indeed, for 87 of the 684 UIDs we did not have any useful training data. We may have had data but they were discarded as non-fit during training so the training set for them was empty. The 87 UIDs were involved in 96% of the SA tests.

Overall, the accuracy of our automated SA breaker significantly aids an attacker in possession of a victim's password. A total stranger, the threat assumed by Facebook, would have to guess the correct individual for at least 5 of the 7 pages with 6 options per page to choose from. Therefore, the probability [7] of successfully solving an SA test with no other information is $(\frac{1}{6})^5 = O(10^{-4})$, assuming photos of the same user do not appear in different pages during the test. At the same time, we have managed to solve SA tests without guessing, using our system, in more than 22% of the tests and reduce the need to guess to only 1–2 (of the 5) pages for 56% of the tests, thus having a probability of $O(10^{-1})$ to $O(10^{-2})$ to solve those SA tests correctly. Overall in 78% of the real social authentication tests presented by Facebook we managed to either defeat the tests or offer a significant advantage in solving them.

3.3.3 Breaking SA: Determined Attacker

In this section we use simulation to play the role of a *determined attacker*, who has access to the majority of the victims' photos. We created an automatic procedure that constructs synthetic instances of SA tests. This automatic procedure follows the same algorithm that Facebook uses to build the real SA tests. Obviously, our procedure keep tracks of the »unknown« subject (*i.e.*, the ground truth) so that we can automatically verify that the outcome of our attack is correct.

For this experiment we implemented a custom face recognition software. This was done for two reasons. First, because we needed something very flexible to use, that allowed us to perform as many offline experiments as needed for the experiments of the determined attacker. Second, we wanted to show that even off-the-shelf algorithms were enough to break the SA test, at least in ideal conditions.

The following experiment provides, as a matter of fact, insight concerning the number of faces per user needed to train a classifier to successfully solve the SA tests.

We created simulated SA tests using the following methodology. We train our system using a training

5 http://seleniumhq.org

6 http://www.torproject.org

7 Calculated using the binomial probability formula used to find probabilities for a series of Bernoulli trials.

	TRAINING	TESTING
Real SA tests	-	127
Photos	17,808	2,667
UIDs	1,131	5,335
Distinct UIDs	1,131	684

Table 2: Human-verified real SA dataset.

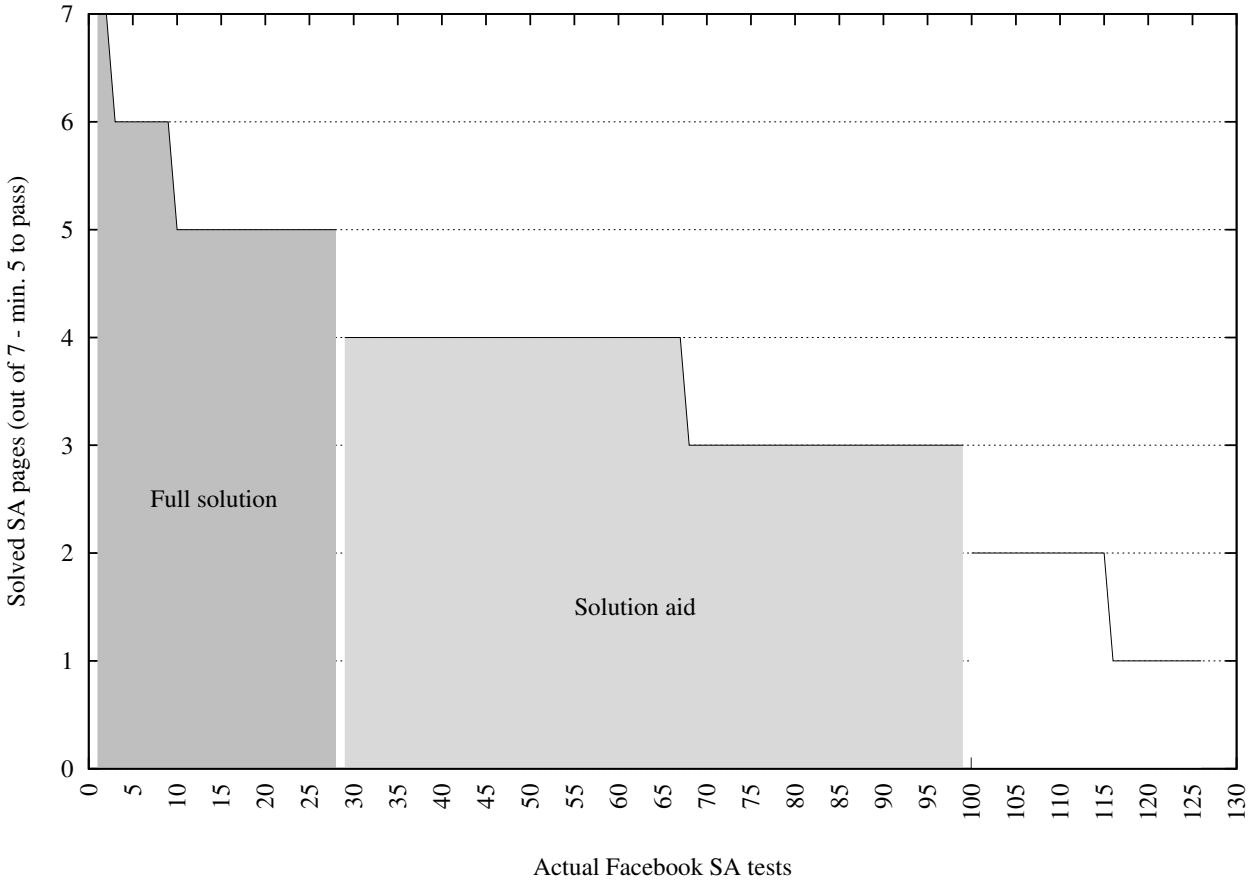


Figure 4: Efficiency of automated SA breaker against actual Facebook tests

set of $K = 10, 20, \ldots, 120$ faces per UID. We extract the faces automatically, without manual intervention, using a face detection algorithm. We then generate 30 SA tests, in which the photos are selected randomly from the pool of public photos we have for each person, from which we exclude the ones used for the training. For each page and number of faces K in the training set, we record the output of the name-lookup step (Step 5), that is the prediction of the classifier, and the CPU-time required. Figure 5 shows the number of pages solved correctly out of 7, and Figure 6 shows the CPU-time required to solve the full test (7 pages).

For an SA test to be solved successfully, Facebook requires that 5 out of 7 challenges are solved correctly. Our results show that our attack is always successful (*i.e.*, at least 5 pages solved over 7) on average, even when a scarce number of faces is available. Clearly, having an ample training dataset such as $K > 100$ ensures a more robust outcome (*i.e.*, 7 pages solved over 7). Thus, our attack is very accurate.

As summarized in Figure 6, our attack is also efficient because the time required for both »on the fly« training—on the K faces of the 6 suggested users—and testing remains within the 5-minute timeout imposed by Facebook to solve a SA test. An attacker may choose to implement the training phase offline using faces of all the victim's friends. This choice would be mandatory if Facebook, or any other Web site employing SA, decided to increase the number of suggested names, or remove them completely, such that »on the fly« training becomes too expensive.

This evaluation reveals that our attack is effective even with off-the-shelf face-recognition software and can break SA tests when supplied with the necessary training data.

3.4 Facebook Response

At the end of our experiments we notified the Facebook Security Team about our results. Though they

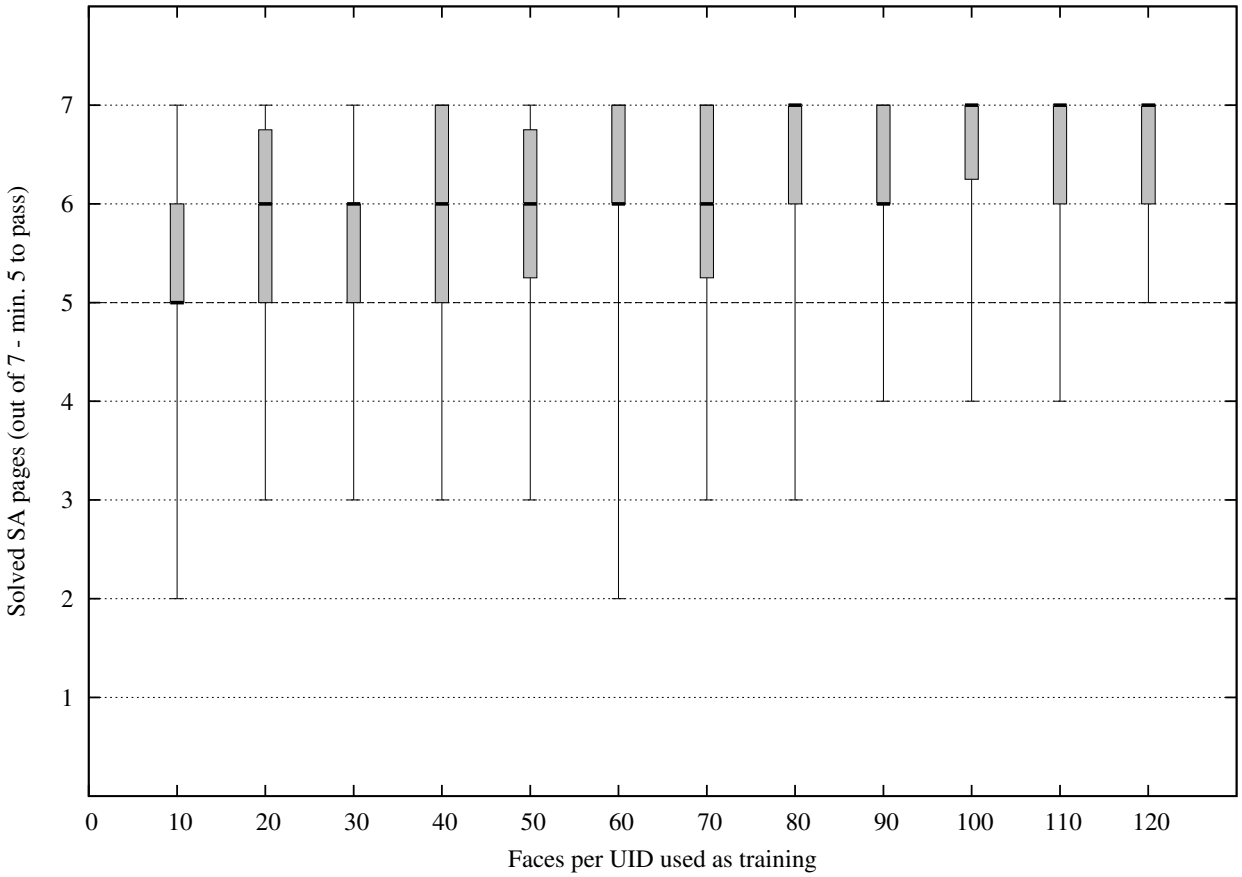


Figure 5: Percentage of successfully-passed tests as a function of the size of the training set. For each iteration, 30 randomly-generated offline SA tests were used.

acknowledged our results and thanked us for sharing them, they slightly disagree with our conclusions. Indeed, Facebook's intention is to deploy SA to raise the bar in large-scale phishing attacks, as SA is neither designed for small-scale or targeted attacks nor can substitute a »strong« two-factor authentication mechanism. However, such a strong two-factor authentication mechanism is still missing, and, even if there were, there is the problem of educating users in its adoption.

4 Social Authentication Revisited

Kim. H. and collaborators in (Kim et al. 2012) study and show the inherent difficulty of implementing a secure SA mechanism. Our work concentrates more on the practical aspects of the risks associated with SA and shows that publicly-available information (*e.g.,* photos, tags, friend list) gives a significant advantage even to casual attackers. Designing effective and usable CAPTCHAs (Bursztein et al. 2010) is indeed as hard as designing effective and usable authentication schemes that exploit social knowledge (Kim et al. 2012). The downside of CAPTCHAs is that they are either too easy for machines or too difficult for humans. This study and previous work show that the main weakness of social-based authentication schemes is that the knowledge needed to solve them is too public: Ironically, the purpose of social networks and the nature of human beings is to share knowledge. On the opposite side, the main strength of good CAPTCHAs (*e.g.,* reCAPTCHA (von Ahn et al. 2008)) is that they are based on an undisclosed ground truth (*e.g.,* random text, audio or video) that is difficult for machines to interpret (*e.g.,* old, noisy recorded conversations). However, we believe that SA tests could be more secure yet still solvable by humans.

In this section we present *reSA*, short for »*Social Authentication, Revisited,*« a design of a secure yet usable SA mechanism for social networks: reSA is a two-factor authentication scheme that can be easily solved by humans but is robust against face-recognition software.

4.1 Study Framework

To receive feedback from human participants on our efforts to enhance the quality of this authentication mechanism, we carried out a user study. As the goal of our research is to evaluate our implementation of a modified SA photo selection scheme for a social networking service, we opt for a diverse set of participants that cannot be found within an academic institution. For that reason, we also explored the possibilities of reaching human subjects through the Amazon

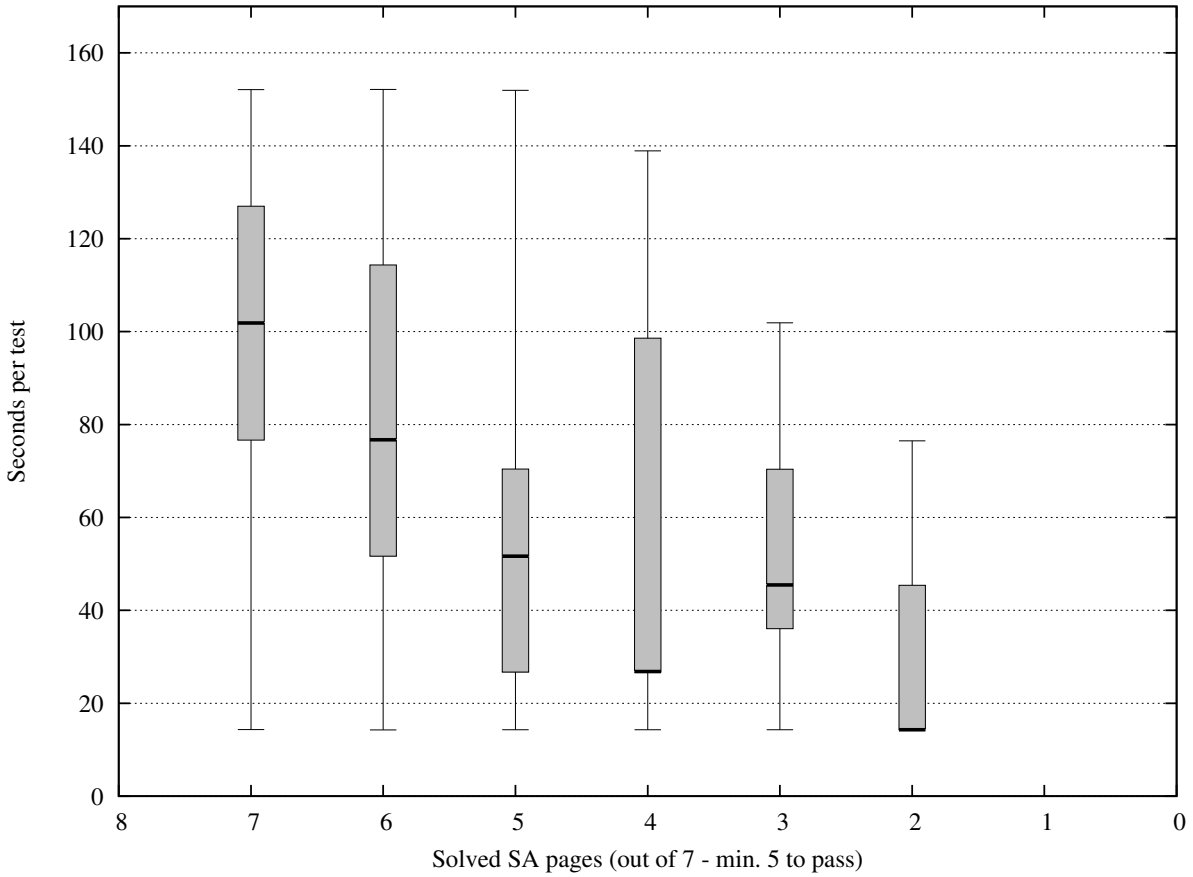


Figure 6: Time required to lookup photos from SA tests in the face recognition system.

Mechanical Turk service [8] and ResearchMatch [9].

Next, we developed a Facebook application to facilitate the efficient interaction between the experiments driving our study and the participants. We opted for a Facebook application because, first of all, they are deployed within a sandbox run by Facebook itself and are, thus, governed by a series of permissions that clearly state and, subsequently, enforce their capabilities and access to user data. Secondly, as we are using Facebook's SA as an example case for improving this type of security mechanism, it was important to integrate our work as close to the mechanics of the service itself as possible. Finally, as we require participants to grant us access to some of the data in their profile (*e.g.,* their social graph), a Facebook application enables direct access. This is also in accordance with our efforts to respect user privacy and minimize collection of potentially sensitive information.

4.2 System Overview

As we wanted to gather statistical data on the ability of humans to solve SA tests with photos of mixed quality, we designed a web application that simulates the SA mechanism. Three preparation steps are needed by the application to be ready to generate tests for a user, as specified in Figure 7.

Step 1: Application Installation The first time a user visit the homepage of *reSA*, he has the possibility to install our application by granting it access to his Facebook account. If the user accepts to grant these permissions, he is informed that the preparation phase is started. He can close the tab with our application and wait for a confirmation email that informs him that his data is ready. The backend of our framework can then start to process the user.

Step 2: Photo Collection Having access to user's information, the system can skim the list of his friends and, for each one of them, collect all the photos (alongside with the corresponding meta-data) in which they are tagged.

Step 3: Tags Processing The tags collected in the previous step are then processed by a face-recognition software in order to obtain some attributes that help us categorize the faces represented in them.

Each photo is submitted to face.com to identify any existing faces, that are then labeled with the UID of their nearest Facebook tag. Subsequently, we assign the `»confidence«` and `»recognizable«` attributes—contained in the face.com API response—to the tags matched with a face (more on this attributes in Section 4.3).

8 https://www.mturk.com/mturk/

9 https://www.researchmatch.org/

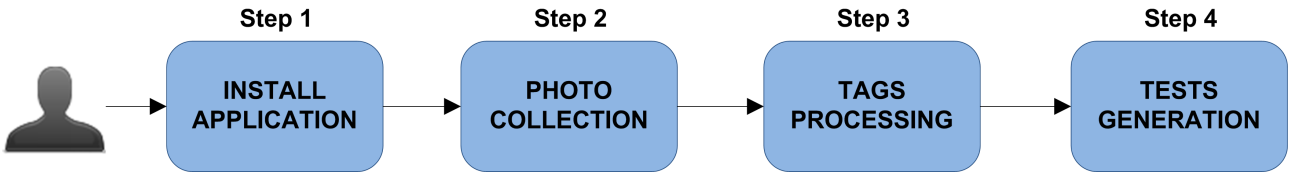


Figure 7: Overview of our study framework. The preparation phase consists of three steps. In Step 1 the user install our Facebook application. Then, in Step 2, we collect all the photos in which the user's friends are tagged. In Step 3 all the collected tags are processed by a face-recognition software. Finally, after all these steps, the application is ready to generate tests for the user.

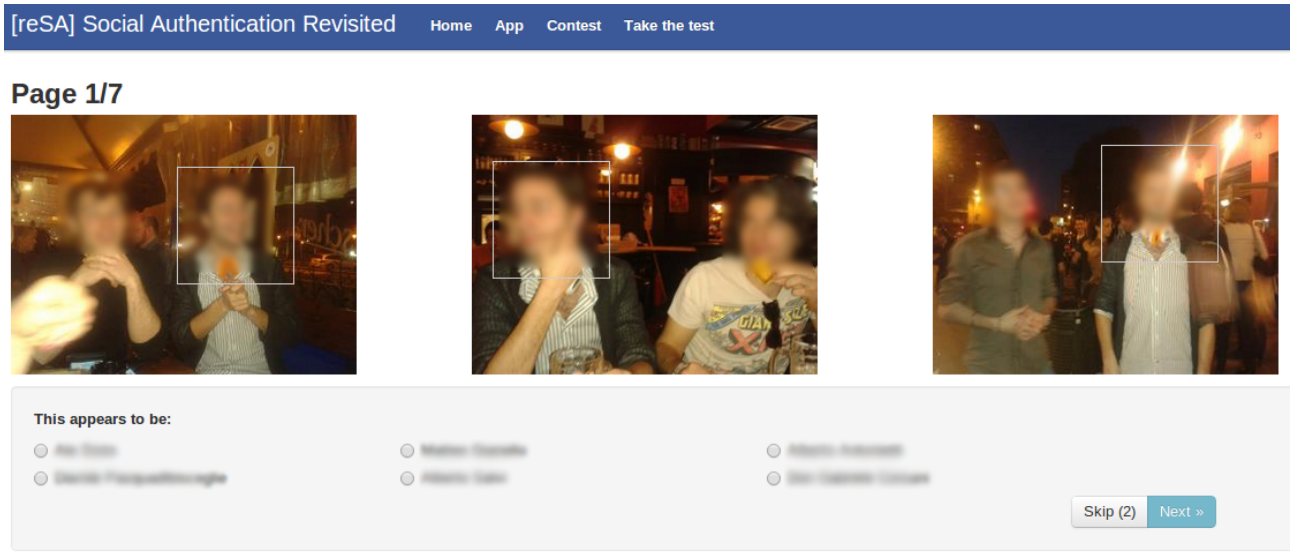


Figure 8: reSA: Test page. Faces and names have been blurred for privacy reasons.

Step 4: Tests Generation When the preparation phase is complete, the user can come back to our application and start to partecipate in the study by taking Social Authentication tests.

Each test is created on-request and is composed by 7 pages, each page containing 3 photos of a user's friend, and 6 suggestions (Figure 8). After selecting the depicted friend, the user is informed if he successfully identified the friend, and is asked to answer some questions that describe the photographs (Figure 9).

4.3 Experimental Evaluation

Here we measure the effectiveness of photo-based social authentication mechanisms. We first explain the photo selection mechanism and the content of our survey. Then we present our dataset and discuss the study results and the outcome that derive.

4.3.1 Photo Selection

We consider two orthogonal characteristics as the criteria for selecting photos to present in SA tests. While we base our selection process on criteria based on the internal implementation of the face.com face recognition algorithm, the insight behind them is rudimentary, and can be extracted from alternate face recognition software as well.

- *Confidence:* when detecting faces, face.com returns its level of confidence that the tagged area is actually a face. Thus, photos that are assigned a confidence level lower than 50% have a high probability of being false-positives, and not containing a face.
- *Recognizable:* not all tags are suitable candidates for training (or recognizing) a classifier for a specific user. Face.com returns a boolean field to indicate this; True for tags that can be recognized or are suitable to be used as part of a training set, and False otherwise.

Eligibility categories Based on the aforementioned selection criteria, we create three categories of photos which we use in our user study. Thus, we are able to evaluate the ability people have in identifying the face of their friends, even under conditions when state-of-the-art face detection algorithms fail to.

1. *Simple category:* here we assign photos that contain user tags that are are most likely to contain a human face. The goal of this category is to provide a base for comparison between the existing SA mechanism, and our revisited approach. According to the study conducted in Section 3, the photos presented in SA tests by Facebook are more likely to contain human faces than randomly-selected photos. As such, we select

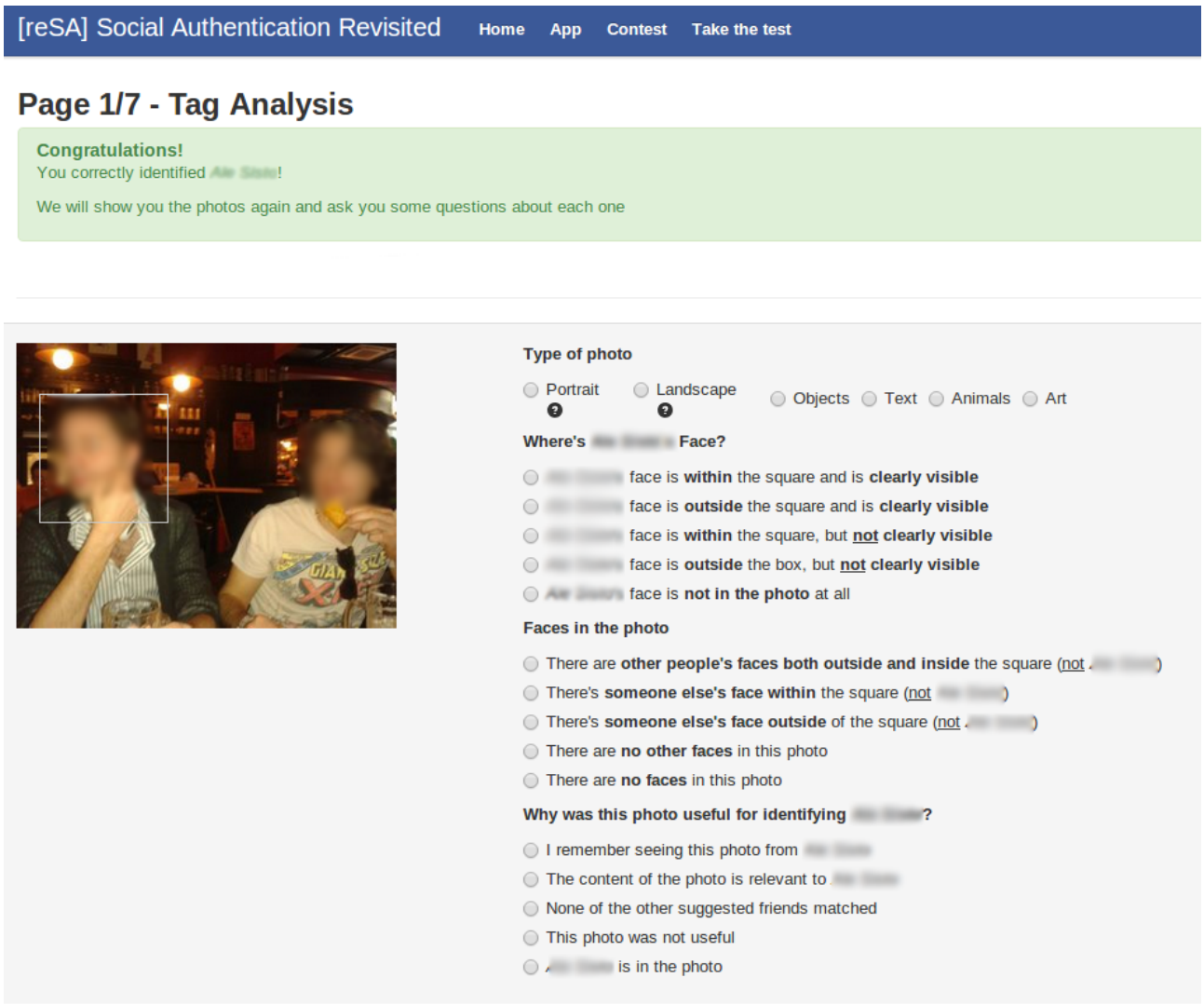


Figure 9: reSA: Questionnaire page. Faces and names have been blurred for privacy reasons.

photos in which face.com has detected faces with high confidence (80%) and has classified them as good candidates for training/recognition (an example is shown in Figure 10(a)).

2. *Medium category:* this category contains the photos used in our revisited approach to SA. The insight behind our photo selection is to choose photos that most likely contain a human face, but are not good candidates for face training/recognition algorithms. Thus, we aim to select photos that will contain tags that users will be able to identify, but software will fail to do so. If such photos are used for training, they will be of no use (as if no training took place) when attempting to recognize a good photo of the same person. Also, if used for recognition (after a classifier has been trained using good photos) it will yield no match. As such, we select photos in which face.com has detected faces with high confidence (80%) but has classified them as bad candidates for training/recognition (an example is shown in Figure 10(b)).
3. *Difficult category:* here we select photos in which face.com returns a low confidence score regarding faces being present (an example is shown in Figure 10(c)). This category is to measure how effective people are at recognizing their friends even if their face is not visible in the photo. This could be due to their posture, their clothes, visible objects etc.

4.3.2 Overall Dataset

Our study involved 141 users—120 males and 21 females, respectively —from 14 different countries, with a predominance from Italy (96 people) and Greece (16 people). The full list of countries is presented in Table 3. The majority of people that partecipated to our experiment has an age comprised from 20 and 40 years (91 people between 20 and 30 years, and 23 within 30 and 40). These users have, on average, 344 friends each, including 211 eligible for the simple type of tests, 172 for the medium category and 182 suitable to be used to construct tests of the difficult kind.

The 141 users lead us to collect a total of $4,457,829$ photos and $5,087,034$ tags. Among these tags,



(a) Simple (b) Medium (c) Difficult

Figure 10: Sample photo from each category.

COUNTRY	NUMBER
Italy	96
Greece	16
Spain	6
United Kingdom	6
Germany	3
United States	3
Colombia	2
France	2
India	2
Czech Republic	1
Dominican Republic	1
Syria	1
Turkey	1
Ukraine	1

Table 3: Distribution of users by country of origin.

$2,066,386$ can be used for the simple category, while the medium and difficult ones may use only $593,479$ and $820,947$ tags, respectively. We found that $1,606,222$ tags doesn't satisfy any selection criteria among the ones we described previously, so they are useless for our study. A summary is given in Table 4.

4.3.3 Study results

During the period in which the experiment lasted, our users took a total number of $1,027$ distinct Social Authentication tests, for an average of 7 tests taken by each user. As summarized in Table 5, both the categories of simple and medium difficulty obtained great results from users, with a success rate that span across 98% and 99%, respectively. Indeed, we collected 358 simple tests, of which 352 completed correctly and 6 failed (*i.e.*, the users passed less than the required 5 of 7 pages). Likewise, we had 341 medium tests, 338 successfully taken and only 3 failed. As we expected, users encountered more problems in solving the difficult kind of tests: among the 328 tests taken of this category, 269 were passed and as many as 59 tests were failed, for a success rate that decreases until 82%.

The outcome of this user study shows that people are able to recognize their friends just as good in both standard SA tests and tests with photos of poor quality. Given that we have demonstrated that standard SA tests are broken (Section 3), we can propose the use of tests with photos of poor quality as that will increase security without affecting usability.

5 Conclusions

In this study we pointed out the security weaknesses of using social authentication as part of a two-factor authentication scheme, focusing on Facebook's deployment. We have empirically calculated the probability of an attacker obtaining the information necessary to solve Social Authentication tests when relying on publicly accessible data as well as following a more active approach to gather restricted information. We found that if an attacker manages to acquire the first factor (password), he can access, on average, 42% of the data used to generate the second factor, thus, gaining the ability to identify randomly selected photos of the victim's friends. Given that informa-

TYPE	TOTAL	MEAN
Photo	4,457,829	31,615
Tags	5,087,034	36,078
Simple	2,066,386	14,655
Medium	593,479	4,209
Difficult	820,947	5,822
Useless	1,606,222	11,391

Table 4: Summary of the collected user data. The mean here computed refers to the number of tags can be used to generate tests for a given user.

TYPE	TOTAL	CORRECT	WRONG	SUCCESS	MEAN
Simple	358	352	6	98.32%	2.54
Medium	341	338	3	99.12%	2.42
Difficult	328	269	59	82.01%	2.33
Total	1027	959	68	93.38%	7.28

Table 5: Summary of the collected Social Authentication tests. The mean here computed refers to the number of tests taken on average by each user.

tion, we managed to solve 22% of the real Facebook SA tests presented to us during our experiments and gain a significant advantage to an additional 56% of the tests with answers for more than half of pages of each test. We have designed an automated attack able to break the Social Authentication, to demonstrate the feasibility of carrying out large-scale attacks against Social Authentication with minimal effort on behalf of an attacker. Our experimental evaluation has shown that widely available face recognition software and services can be effectively utilized to break Social Authentication tests with high accuracy. Overall we argue that Facebook should reconsider its threat model and re-evaluate the security measures taken against it.

We then evaluated both the security and usability level of face-based social authentication systems with *reSA*, a web application that simulates the Social Authentication mechanism. We carried out a user study, where we asked humans to solve SA tests and answer to a survey that helped us better understand tagging behaviors on Facebook. We found that people are able to recognize their friends just as good in both standard SA tests and tests with photos of poor quality, so we propose the use of tests with photos of poor quality as that will increase security without affecting usability.

6 About the Author

Marco Lancini has obtained a M.Sc. degree in Engineering of Computing Systems at Politecnico di Milano in 2013, where he was a member of the Computer Security Group, under advice from prof. Stefano Zanero.

Since then he is a Security Researcher and Consultant at CEFRIEL (ICT Center of Excellence For Research, Innovation, Education and Industrial Labs partnership), where he works across several aspects of computer security. His principal research interests are mobile security, privacy, and web applications' security.

He can be reached at marco.lancini@mail.polimi.it.

Acknowledgments The work here presented is an extract of Marco Lancini's M.Sc. Thesis[10] (Lancini 2013), that is the result of the collaboration between *Politecnico di Milano* [11], *Columbia University* [12] and *FORTH* [13], within the *SysSec EU Network of Excellence* [14].

References

Bilge, L., Strufe, T., Balzarotti, D. & Kirda, E. (2009). All your contacts are belong to us: automated identity theft attacks on social networks. In *Proceedings of the 18th International Conference on World Wide Web*. ACM.

Boshmaf, Y., Muslukhov, I., Beznosov, K. & Ripeanu, M. (2011). The socialbot network: when bots socialize for fame and money. In *Proceedings of the Annual Computer Security Applications Conference*. ACM.

Bursztein, E., Bethard, S., Fabry, C., Mitchell, J. C. & Jurafsky, D. (2010). How Good Are Humans at Solving CAPTCHAs? A Large Scale Evaluation. In *Proceedings of the 2010 IEEE Symposium on Security and Privacy*. IEEE.

10 Full text available at: http://hdl.handle.net/10589/78569

11 http://www.polimi.it

12 http://www.columbia.edu

13 http://www.forth.gr

14 http://www.syssec-project.eu

Dey, R., Jelveh, Z. & Ross, K. (2012). Facebook Users Have Become Much More Private: A Large-Scale Study. In *Proceedings of the 4th IEEE International Workshop on Security and Social Networking*. IEEE.

Dhamija, R., Tygar, J. D. & Hearst, M. (2006). Why phishing works. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM.

Dunbar, R. (1998). *Grooming, Gossip, and the Evolution of Language*. Harvard University Press.

Facebook. (2011a, January). A Continued Commitment to Security. Retrieved from https://blog.facebook.com/blog.php?post=486790652130

Facebook. (2011b, August). A Guide to Facebook Security. Retrieved from https://www.facebook.com/note.php?note_id=10150261846610766

Facebook. (2011c, November). Anatomy of Facebook. Retrieved from https://www.facebook.com/notes/facebook-data-team/anatomy-of-facebook/10150388519243859

Facebook. (2012, June). A Few Updates to Make Your Mobile Experience More Safe and Secure. Retrieved from http://bit.ly/fb-mobile-update

Gao, H., Hu, J., Wilson, C., Li, Z., Chen, Y. & Zhao, B. Y. (2010). Detecting and characterizing social spam campaigns. In *Proceedings of the 10th Annual Conference on Internet Measurement*. ACM.

Jacoby, D. (2012, January). Facebook Security Phishing Attack In The Wild. Retrieved from http://www.securelist.com/en/blog/208193325/Facebook_Security_Phishing_Attack_In_The_Wild

Kim, H., Tang, J. & Anderson, R. (2012). Social Authentication: harder than it looks. In *Proceedings of the 2012 Cryptography and Data Security conference*. Springer.

Krishnamurthy, B., Gill, P. & Arlitt, M. (2008). A few chirps about twitter. In *WOSN '08: Proceedings of the first workshop on Online social networks'* (Pages 19–24). ACM.

Lancini, M. (2013, April). *Social Authentication: Vulnerabilities, Mitigations, and Redesign. Master's Thesis, Department of Electronic and Information, Politecnico di Milano, Italy*.

Lancini, M. (2014). Social Authentication: Vulnerabilities, Mitigations, and Redesign. *Magdeburger Journal zur Sicherheitsforschung*, 2, 476–492. Retrieved November 13, 2014, from http://www.sicherheitsforschung-magdeburg.de/publikationen.html

Madejski, M., Johnson, M. & Bellovin, S. M. (2012). A Study of Privacy Settings Errors in an Online Social Network. In *Proceedings of the 4th IEEE International Workshop on Security and Social Networking*. IEEE.

Nagle, F. & Singh, L. (2009). Can Friends Be Trusted? Exploring Privacy in Online Social Networks. In *Proceedings of the 2009 International Conference on Advances in Social Network Analysis and Mining*. IEEE.

Shepard; Jonathan, L., Chen, W., Perry, T. & Popov, L. (2010, September 9). *Using Social Information For Authenticating A User Session*. US 20100229223. US. Retrieved from http://www.patentlens.net/patentlens/patent/US_2010_0229223_A1

Shulman, A. (2010). The underground credentials market. *Computer Fraud & Security*, (3).

Staddon, J. & Swerdlow, A. (2011). Public vs. publicized: content use trends and privacy expectations. In *Proceedings of the 6th USENIX Conference on Hot Topics in Security*. USENIX.

Stringhini, G., Kruegel, C. & Vigna, G. (2010). Detecting spammers on social networks. In *Proceedings of the 26th Annual Computer Security Applications Conference* (Pages 1–9). ACSAC '10'. Austin, Texas: ACM. doi:10.1145/1920261.1920263

Tang, J., Musolesi, M., Mascolo, C. & Latora, V. (2009). Temporal distance metrics for social network analysis. In *WOSN '09: Proceedings of the 2nd ACM workshop on Online social networks'* (Pages 31–36). ACM.

Twitter. (nodate). Twitter turns six. http://blog.twitter.com/2012/03/twitter-turns-six.html.

Ur, B. E. & Ganapathy, V. (nodate). Evaluating Attack Amplification in Online Social Networks. In *Proceedings of the 2009 Web 2.0 Security and Privacy Workshop*.

Viswanath, B., Mislove, A., Cha, M. & Gummadi, K. P. (2009). On the evolution of user interaction in Facebook. In *WOSN '09: Proceedings of the 2nd ACM workshop on Online social networks'* (Pages 37–42). ACM.

von Ahn, L., Maurer, B., Mcmillen, C., Abraham, D. & Blum, M. (2008, August). reCAPTCHA: Human-Based Character Recognition via Web Security Measures. *Science*, *321*(5895).

Zuckerberg, M. (nodate). One Billion People on Facebook. http://newsroom.fb.com/News/457/One-Billion-People-on-Facebook.