

All Your Contacts Are Belong to Us: Automated Identity Theft Attacks on Social Networks

Leyla Bilge, Thorsten Strufe, Davide Balzarotti, Engin Kirda
EURECOM

Sophia Antipolis, France

bilge@eurecom.fr, strufe@eurecom.fr, balzarotti@eurecom.fr, kirda@eurecom.fr

ABSTRACT

Social networking sites have been increasingly gaining popularity. Well-known sites such as Facebook have been reporting growth rates as high as 3% per week [5]. Many social networking sites have millions of registered users who use these sites to share photographs, contact long-lost friends, establish new business contacts and to keep in touch. In this paper, we investigate how easy it would be for a potential attacker to launch automated crawling and identity theft attacks against a number of popular social networking sites in order to gain access to a large volume of personal user information. The first attack we present is the automated identity theft of existing user profiles and sending of friend requests to the contacts of the cloned victim. The hope, from the attacker's point of view, is that the contacted users simply trust and accept the friend request. By establishing a friendship relationship with the contacts of a victim, the attacker is able to access the sensitive personal information provided by them. In the second, more advanced attack we present, we show that it is effective and feasible to launch an automated, cross-site profile cloning attack. In this attack, we are able to automatically create a forged profile in a network where the victim is not registered yet and contact the victim's friends who are registered on both networks. Our experimental results with real users show that the automated attacks we present are effective and feasible in practice.

Categories and Subject Descriptors

D.2.0 [Software]: Software Engineering : General; H.M [Information Systems]: Miscellaneous

General Terms

Security

Keywords

Social Network Security, Identity Theft

1. INTRODUCTION

A social network is a social structure that is made up of nodes representing individuals or organizations. These nodes may be tied to each other by properties such as friendship, common values, visions, ideas, business relationships and general interests. Although the idea of social networks has been around for a long time (e.g., see [14]), social networking web sites and services are a relatively new phenomenon on the Internet. Business relationship-focused social networking sites such as XING [13] (previously known as OpenBC) and LinkedIn [6], as well as friendship-focused social networking sites such as Facebook [4], MySpace [8], StudiVZ [11] and MeinVZ [7] have been gaining popularity among Internet users. In fact, LinkedIn boasts on its web site that it has 30 million registered users. XING, a business networking site that is very popular in Switzerland, Germany and Austria, claims to have 6 million registered users. Although it has only been created four years ago, Facebook now has more than 150 million active users and is reporting growth rates of 3% per week. According to Facebook, it registers 30 billion page views per month and is the largest photo storage site on the web with over 1 billion uploaded photos [5].

Unfortunately, as the interest for a new technology grows on the Internet, miscreants are attracted as well. For example, spam was not a major problem until the end of the '90s. However, as more and more people started using e-mail, unsolicited (i.e., spam) e-mails started increasing in numbers. In fact, spam has reached such high proportions that the Spamhouse Project [12] now estimates that about 90% of the incoming e-mail traffic in North America, Europa and Australasia is spam. Also, the increase in the popularity of e-mail also resulted in an increase in the numbers of malicious e-mails (e.g., e-mails with worm attachments, phishing e-mails, scam e-mails, etc.). Today, e-mail is a popular way of spreading infections.

As the popularity of social networking sites increase, so does their attractiveness for criminals. For example, worms have recently emerged that specifically target MySpace and Facebook users [9]. These worms make use of old ideas that are applied to a new technology. Analogous to classic worms such as LoveLetter [3] that used the contacts in a victim's Outlook address book to spread, these new social networking worms use the friend lists of a victim to send a copy of themselves to other social networking users. Although such e-mail attachments may raise more suspicion now as such tricks have already been seen by many e-mail users, they are not as well-known on social networking sites. Fur-

thermore, note that incoming e-mails with attachments are often scanned for malicious content and Bayesian filters are applied to sort out unsolicited mails. In comparison, social networking sites do not usually provide filtering mechanisms or warnings for dangerous content, hence, making it easier, in principle, for a potential attacker to send malicious applications and URLs to victims.

Fortunately, so far, social networking sites and services have been spared from large-scale, high profile attacks. Nevertheless, social networking sites are an attractive target for attackers because of the nature of the sensitive information that they contain on registered users. Typically, users enter their real e-mail addresses and provide information on their education, friends, professional background, activities they are involved in, their current relationship status and sometimes even list their previous relationships (e.g., on Facebook, one may read that Mr X. was together with Ms Y until they broke up in 2006). Hence, from the attacker's point of view, access to this type of detailed, personal information would be ideal for launching targeted, social engineering attacks, now often referred to as spear phishing [2, 1]. Furthermore, the collected e-mail addresses and personal information would be invaluable for spammers as they would 1) have access to e-mail addresses that belong to real people (i.e., one problem spammers face is that they often do not know if the e-mail addresses that they collect are indeed being used by real people or they are just secondary addresses that are not regularly read) and 2) have information about the people using these e-mail addresses allowing them to efficiently personalize their marketing activities, tailored according to the knowledge from the target's profile. Also, note that the ability to associate personal information with an e-mail address is important to be able to successfully bypass spam filters [21]. Such filters usually generate a list of "spammy" tokens versus "good" tokens after training with a large set of previously received e-mails. As a result, e-mails that contain the name of the user receiving the e-mail, or names of people that he is acquainted with tend to receive lower spam ratings than e-mails that are less personal. As a result, if the spammer is able to include some personal information in the spam that he is sending, he would be able to improve his chances of reaching the targeted user.

Typically, a prerequisite for being able to access personal information in a social networking site is to have a confirmed personal "relationship" with the person who is concerned. The default setting in Facebook is to allow all confirmed friends to have access to the personal information (e-mail address, photographs, etc.), but not to provide it to unconfirmed third parties. In LinkedIn, the contacts of a person can only be accessed if it is a confirmed business contact, and therefore he/she has already accepted a request and confirmed the relationship.

Hamiel and Moyer conducted an impersonation experiment in which they created a fake profile on LinkedIn for the well-known security expert Marcus Ranum. The authors obtained the information to create a plausible profile by manually surfing the web, visiting Ranum's personal web page, and his entry in Wikipedia [25]. By impersonating a high-profile person, the authors showed how effective an impersonation attack can be. The forged profile received many friend requests, even from one of the target's immediate family members.

In this paper, we investigate how easy it would be for a

potential attacker to launch this type of impersonation attacks in an automated fashion against a number of popular social networking sites in order to gain access to a large volume of personal user information. Unlike a Sybil attack [17] where the attacker aims to subvert a reputation system of a peer to peer or a social network by creating a large number of pseudonymous entities, the attacks presented in this paper consist of the automated identity theft of real user profiles. In the first attack we clone an already existing profile in a social network and we send friend requests to the contacts of the victim. Hence, we are able to "steal" the contacts of a user by forging his identity and creating a second, identical profile in the same social network. Having access to the contacts of a victim, therefore, means that we can access the sensitive personal information provided by these contacts. Our experimental results show that a typical user tends to accept a friend request from a forged identity who is actually already a confirmed contact in their friend list.

In the second attack we present, we show that it is effective and feasible to launch an automated, *cross-site profile cloning* attack. In this attack, we are able to automatically identify users who are registered in one social network, but who are not registered in another. We can then clone the identity of a victim in the site where he is registered, and forge it in a social networking site where he is not registered yet. After we have successfully created the forged identity, we can then automatically attempt to rebuild the social network of the victim by contacting his friends that we have identified to be registered on both social networking sites. Our experimental results suggest that this attack is especially effective because profiles in this case only exist once on the social networking site that is being targeted. As a result, the friend requests that we send look perfectly legitimate and do not raise suspicion with the users who have been contacted.

We implemented our attacks in a prototype system called iCloner (identity Cloner). iCloner consists of several components that are able to crawl popular social networking sites, collect information on users, automatically create profiles, send friend requests and personal messages. Furthermore, iCloner also supports CAPTCHA [16] analysis and breaking capabilities that make our attacks feasible against social networking sites that employ CAPTCHAs to prevent automated access.

It is important to note that the attacks we present in this paper can potentially be launched on a large scale, allowing an attacker to control hundred of thousands of cloned accounts and thus reaching millions of real user profiles. Furthermore, if the attacker has a high number of different IP addresses at his disposal (such as a botnet that consists of thousands of compromised hosts), the detection of an automated attack like the ones presented in this paper may become more difficult.

The contributions of this paper are the following:

- We show that it is feasible in practice to launch automated attacks against five popular social networking sites. In particular, we present two automated identity theft attacks: Profile cloning and cross-site profile cloning.
- Even though some of the sites employ CAPTCHAs to prevent automated access, in some cases, there is

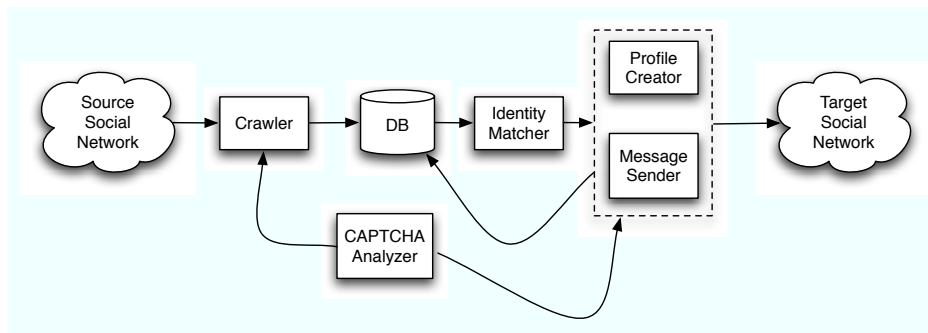


Figure 1: An architectural overview of iCloner

significant room for improvement to make these CAPTCHAs more difficult to break.

- We present experimental results with real users and show that the attacks we present are feasible in practice. Our results confirm empirically, as one would expect, that most social network users are not cautious when accepting friend requests or clicking on links that are sent to them. In fact, many are even willing to accept friend requests from people they do not know.
- We make suggestions on how social networking sites can improve their security, and therefore, better protect the privacy of their users.

The rest of the paper is structured as follows. In Section 2, we give a brief overview of iCloner, our prototype attack system. In Section 3, we describe the two cloning attacks that we used in our experiments to gain access to the victims' contacts. In Section 4, we discuss the results of our experiments. In Section 5, we discuss how social networking sites can improve their security. In Section 6, we discuss related work and conclude the paper in Section 7.

2. ICLONER OVERVIEW

In this section, we give a brief overview on the architecture of iCloner.

2.1 Architecture of the system

Our prototype attack system consists of four main components: The *crawler* component is responsible for crawling the target social networking site and collect information on users that have chosen to make their profiles public. The personal information of these users, therefore, can be accessed by all members of the social network. In some networks, such as Facebook, the default setting does not allow anyone to see any other person's personal information unless they are friends. However, by default, friend lists are public information on Facebook. In contrast, one needs to be friends (i.e., business colleagues) with a person on LinkedIn to be able to see their contacts. Our crawler component is able to crawl through StudiVZ, MeinVZ, Facebook and XING and collect information on contact lists and profiles if these are accessible to the public. The crawler also keeps track of which user profiles could not be retrieved (because of more restrictive user access settings).

The *identity matcher* analyzes the information in the database and tries to identify profiles in different social networks

that correspond to the same person. The *profile creator* component can then use this information to create accounts in a social network where the victims have not registered yet, or to duplicate an existing profile inside the same network.

Finally, the *message sender* component is responsible to login into the created accounts and automatically send friend requests to the people that are known to be friends with the victim. Depending on the social networking site that is being targeted, CAPTCHAs might need to be solved to create accounts, to send friend requests, and sometimes even to access a user's profile (if a user sends many requests, a social networking site might request to verify that the user is a real person and not a script).

The CAPTCHAs are analyzed by the *CAPTCHA analysis* component in our system. In particular, we have analyzed the CAPTCHAs that are displayed by StudiVZ, MeinVZ, and Facebook and have designed techniques to break these CAPTCHAs with a success rate that makes automated attacks feasible in practice. Note that we have not encountered CAPTCHAs on LinkedIn, and did not need to solve CAPTCHAs for our experiments with XING.

Figure 1 gives an architectural overview of iCloner and depicts the dependencies between the various components.

2.2 Breaking CAPTCHAs

A CAPTCHA [16] (Completely Automated Public Turing test to tell Computers and Humans Apart) is a type of challenge-response test that is commonly used to determine whether or not the user of a certain application is a human being. Therefore, the key feature of any CAPTCHA algorithm is the ability to generate tests that are at the same time easily solvable by humans, but very hard to solve for a computer application. For instance, since most of the CAPTCHAs are based on the ability to recognize a text in presence of noise, a good CAPTCHA should be resistant against Optical Character Recognition [24] (OCR) techniques.

Just like many other online web services, in social networks, CAPTCHAs are usually employed to prevent automated programs from accessing and abusing the provided services. For example, without CAPTCHAs, it would be trivial for miscreants to crawl the social network in order to automatically collect personal information, and spam the registered users.

Even though breaking CAPTCHAs is not in the focus of this paper, in order to automate our attacks we had to develop a number of CAPTCHA breaking techniques based on a set of open source tools and custom-developed scripts. We used ImageMagick [19] for image filtering, Tesseract [27]

