Nowhere to Hide: Navigating around Privacy in Online Social Networks

Mathias Humbert, Théophile Studer, Matthias Grossglauser, and Jean-Pierre Hubaux

LCA, EPFL, Lausanne, Switzerland
firstname.lastname@epfl.ch

Abstract. In this paper, we introduce a navigation privacy attack, where an external adversary attempts to find a target user by exploiting publicly visible attributes of intermediate users. If such an attack is successful, it implies that a user cannot hide simply by excluding himself from a central directory or search function. The attack exploits the fact that most attributes (such as place of residence, age, or alma mater) tend to correlate with social proximity, which can be exploited as navigational cues while crawling the network. The problem is exacerbated by privacy policies where a user who keeps his profile private remains nevertheless visible in his friends' "friend lists"; such a user is still vulnerable to our navigation attack. Experiments with Facebook and Google+ show that the majority of users can be found efficiently using our attack, if a small set of attributes are known about the target as side information. Our results suggest that, in an online social network where many users reveal a (even limited) set of attributes, it is nearly impossible for a specific user to "hide in the crowd".

1 Introduction

Over the last few years, online social networks (OSNs) have revolutionized the way people behave and interact with each other over the Internet. OSNs enable the majority of users to not just be passive consumers of the Web, but to become active producers of content, and to be storytellers of their own lives for the first time online. The other side of the coin is that privacy breaches are intrinsically bound to OSNs, and new forms of surveillance and control have emerged with OSNs. Recruiters are now known to look up Facebook profiles of job applicants, and hiring discrimination based on OSNs has become a serious threat [2, 10]. Some employers and colleges even request the Facebook passwords of job applicants and student athletes in order to get full access to their profiles [36]. OSNs have also been exploited by government agencies of authoritarian regimes to infiltrate protesters' social networks. Several Syrian activists have notably reported having been arrested and forced to reveal their Facebook passwords [35]. These practices are only the tip of the iceberg of privacy erosion caused by OSNs.

The first, straightforward method for finding an individual in an online social network is to rely on a central directory, if available. Obviously, a user u trying to keep his profile private would opt not to be listed in such a directory or, if this privacy option is not available,[1] make use of a pseudonym. The second method to reach u is to rely on the social links between users and to navigate via these links towards u. This approach works if some of u's friends show their friend lists publicly (thereby exposing u), which is the default setting in most OSNs.

In order to find a hidden user, an attacker could search the whole public social graph. However, such an exhaustive search, despite guaranteeing to find any user in the giant component,[2] would certainly be too expensive for OSNs that contain hundreds of millions users, notably because of the anti-crawling features deployed by virtually all OSNs. To reduce the search cost, the attacker can decide to crawl only a targeted subset of OSN users. In this paper, we evaluate the feasibility of such an attack for *large* networks and ultimately answer the following question: Is it possible to find a target profile by navigating a small fraction of the whole network, by relying on public attributes of queried profiles? Answering this question is crucial for privacy, because reaching the target profile or its neighborhood is *the necessary precondition* for any targeted attack such as the inference of hidden attributes (e.g., political or religious views) through other personal attributes [7, 29], or through friends' public attributes [8, 22, 33].

To the best of our knowledge, this is the first work proposing to find a target profile in an OSN by making use of social links between users. Our *navigation attack* is generic in order to apply to any attribute-enhanced OSN (such as Facebook, Google+, or Twitter). We propose a search algorithm that relies on a space of attributes and distance heuristics based on A^* search [17]. The categories of attributes and their priorities can be adapted to any kind of OSN. Given the OSN visibility, privacy policies, and the users' privacy choices, we show how the attack can be efficiently carried out, by implementing it in the two largest OSNs, Facebook and Google+. For these OSNs, building upon results on navigation and routing in social networks, the attack first relies on geographical attributes only, then making use of additional types of attributes (such education or work) as soon as it reaches the target's city. Our results demonstrate that 66.5% of Facebook users are findable by crawling a median number of users smaller than 400; and 59% of Google+ users are findable by crawling a median number of users small than 300. This shows that it is very difficult to hide in an OSN, however large it is and to prevent targeted attacks and/or to deny the existence of a profile. Moreover, targets' cities are reached in 92% and 93.5% of the cases by crawling a median number of 13 and 8 users, in Facebook and Google+, respectively. This shows the efficiency of geographic navigation in Facebook and Google+. We propose two main explanations for the failed cases. First, the users least likely to be discovered are those who have a small number of friends, or privacy-cautious friends (who do not reveal too much information), or friends whose revealed information is not similar to their own information. Second, users

[1] It is the case of Facebook since the end of 2012.

[2] This holds if the search starts from the giant component and the target is in this component too. This is a fair assumption for current OSNs; for example, in Facebook, 99.91% of users belong to the giant component [39].

living in larger cities tend to be harder than others to discover in Facebook. Although the latter reason is inherent to the structure of the OSN and to the limit we impose on the number of crawled users, the former is essentially due to the privacy settings of the targets' friends and the OSN dynamics. Our results show that homophily in social networks [5, 30] does not only allow us to infer hidden attributes of OSN users locally, but also allows us to efficiently navigate toward the target. Note that we do not assume any prior knowledge about the network structure and the users' distribution in the network. Moreover, by starting the navigation from a random user in the network, we consider the worst-case scenario for the attacker and provide a lower-bound on the attack efficiency. It is clear that the use of advanced search filters or source users closer to the target can only further benefit the attacker. We briefly show how this can dramatically reduce the search cost. Finally, we show that simple countermeasures exist and could be implemented upstream by the OSN operators.

2 Related Work and Background

We present here the most closely related work on privacy threats in OSNs, showing how our paper complements existing attacks. We also discuss the background on navigation in social networks.

2.1 Privacy issues in OSNs

Acquisti and Gross were among the first to mention the potential risks induced by information sharing in OSNs in their seminal papers [1, 13]. They study in detail the Facebook privacy settings and data visibility, and they emphasize the potential threats caused by weak privacy settings (used by most users). In [23] and [24], Krishnamurthy and Wills study what types of information are shared with whom, by default or not, and what kind of privacy settings are available for various pieces of *personally identifiable information.* They show that, among 12 OSNs, 10 publicly reveal social links by default and 1 reveals them always (i.e., without any possibility of changing the settings). 7 reveal by default the user's location and 5 always reveal it. 8 reveal the attended schools by default and 6 the employers. These statistics are relevant for our work as they show what kind of attributes are publicly revealed, and thus can be used for the navigation.

He et al. [18] were among the first to propose inference attacks based on the users' neighborhood. They make use of Bayesian inference and multi-hop inference to predict private attributes based on the friends, and friends of friends of the targeted users. The authors apply their analytical findings to a LiveJournal dataset with hypothetical attributes. In the same vein, Lindamood et al. propose to infer political affiliation (binary attribute: liberal or conservative) based on a modified Naive Bayes classifier [27]. Their results show that simply sanitizing user attributes or links is not enough to prevent inference attacks. Johnson [20] also emphasizes that social links can leak very sensitive information about a specific Facebook user, for instance whether a certain user is homosexual or not.

Zheleva and Getoor [43] propose novel inference attacks based on social links and group memberships, which they apply in four different social networks. Another work on inference of undisclosed attributes proposes to rely on any of the user's public attributes, and on any of the aggregates of his friends' attributes [22]. Finally, Chaabane et al. [7] show how music interests can be used to infer private sensitive attributes of Facebook users. Their approach does not rely on users' social links or group memberships, but only on users' attributes.

Thomas et al. [37] examine how the lack of joint privacy controls can put a user's privacy at risk. Notably, they highlight the inherent interdependent privacy risks due to friends in Facebook, and the fact that a user had no control over his friends' friend lists. They present inference techniques that, based on wall posts and friends, present improvements compared to previous work by relying only on friends to infer private attributes. Yamada et al. [42] also emphasize the impact of conflicting privacy policies on users' privacy. They propose 3 different attacks: friend-list, profile and wall-post recovery attacks. Dey et al. [8] estimate the leakage of age information in Facebook, either by relying on the target's profile directly, or by using information released by the targets' friends.

While these previous papers exploit the notion of homophily to infer hidden attributes of a user from the visible attributes of his neighbors, our work exploits the global structure of visible attributes to navigate efficiently towards a target. While the former is a purely local operation, ours exploits a macroscopic property of the social network. It complements existing work by showing how to efficiently find anyone in an OSN, necessary condition for any targeted inference attack.

Finally, Jain and Kumaraguru propose an integrated system which uses major dimensions of a user identity (profile, content and network) to search and link a user across multiple social networks [19]. Our work notably differs in the method used to search for a user. Our navigation attack does not require the targeted user to be present in multiple OSNs, and does not assume the target profile to be known in one OSN in order to find him in another.

2.2 Navigation in Social Networks

The seminal experiment by Milgram [31] shows that any arbitrarily selected individuals can reach any other person through a short chain of acquaintances. There generally exists a short path from any individual to another, thanks to a few long-range social links. However, knowing that short chains exist does not tell us how arbitrary pairs of strangers are able to find them. Since Milgram's experiment, there have been many theoretical and experimental papers that explain how people can find short paths, and thus navigate, in social networks [26]. Travers and Milgram ask 296 arbitrarily selected individuals in the United States to generate acquaintance chains (using postal mail) to a single target person. Out of the 296 starting chains, 64 reach the target (22% of completion rate) with a mean number of intermediaries between the sources and the target of 5.2 [38]. They also show that chains converge essentially by using geographic information; but once in the target's city, they often circulate before entering the target's circle of acquaintances. Dodds et al. propose a similar social-search experimental approach except that they rely on e-mails instead of classic postal service to reach a target [9]. They show that geography clearly dominates the routing strategies of senders at early stages of the chains and is less frequently used than other characteristics (such as occupation) after the third step.

Liben-Nowell et al. study the role of geography in order to route messages in social networks and provide a theoretical model to explain path discovery [26]. To the best of our knowledge, they are the first to analyze routing in an online social network (LiveJournal). However, they limit themselves to the problem of reaching the target's city. They show that geography remains a crucial factor in online connections and is thus very helpful when trying to reach a target. Lattanzi et al. extend this one-dimensional approach based on geographical proximity to a multidimensional space of interests relying on a model of social networks called "affiliation networks" [25]. In contrast with these contributions, our work studies large OSNs that allow users to finely tune their privacy settings to protect their privacy. Our paper notably shows that privacy policies remain weak and do not protect enough the privacy-cautious users, notably against navigation attacks.

Knowing that acquaintances' and social networks show small-world properties, we now question whether current OSNs do so as well. Mislove et al. already provided a piece of the answer to that question in 2007 [32]. The considered OSNs exhibit power-law degree distributions, a densely connected core of high-degree nodes linking small groups of strongly clustered nodes and, as a result, short path lengths. A crucial step in providing evidence about the small-world characteristics of OSNs has recently been achieved with the publication of two reports by Facebook researchers on the Facebook full social graph [6, 39]. Their dataset of 721 million users shows the main small-world properties: 99.91% users belong to the largest component, the distribution of nodes degree follows a power-law distribution, and the average distance between users equals 4.7, showing that OSNs are even smaller than real-world social networks. We can thus predict that, by relying on users' attributes, most OSNs should also be navigable. However, how to efficiently navigate on them was until now an open question. Furthermore, Facebook reports considered the full social graph, with all social links, whereas the attacker assumed in this work would not have access to all those links. In this paper, we study if the public subgraph induced by the users' privacy settings on their social links is navigable by relying on publicly revealed attributes.

3 Model

OSN Model Online social networks can be described as social links between online users who own a personal profile. Formally, an OSN can be defined as a graph $G = (V, E)$, where the vertex set, V, represents the set of users[3] and E, the edge set, their social links. Each user $u \in V$ is endowed with a set of attributes A_u that is a subset of the set A of the available attributes (gender, birthdate, education, city, ...). OSNs with symmetric social links requiring mutual consent, such

[3] In the rest of the paper, we will alternatively write *user*, *node* or *vertex* to refer to a member of the OSN.

as Facebook or LinkedIn, can be modeled as undirected graphs, whereas OSNs with asymmetric social links, such as Twitter or Google+, can be represented as directed graphs.[4]

In most OSNs, users can decide to what extent and with whom they share information by appropriately tuning their privacy settings. For instance, in Facebook users can reveal personal attributes to *friends* only, to *friends of friends*, or to *everyone* in the OSN. The same settings are generally available for their list of social links. $A^i_u = \emptyset$ denotes that a particular attribute A^i is not publicly revealed by user u. Embedding users' privacy settings on their social links into the original social graph G induces a directed public subgraph D, where directed edges are those whose tail vertices have publicly available social links. Formally, $D = (V, E_d)$, with $E_d = \{(u, v)|(u, v) \in E, \Gamma(u) \neq \emptyset\}$, where $\Gamma(u)$ represents the out-neighbors of $u \in D$. Note that we make the conservative assumption that all privacy settings except the public one (e.g., *everyone* in Facebook) are private (e.g., *friends*, *friends of friends*), as we cannot access the information if we are not part of a user's close neighborhood.

Attacker Model The attacker can be any external curious entity that wants to collect data or infer information about a target t. We assume that the attacker controls at least one node and can thus have access to information publicly visible in the OSN. In order to reach his target, the attacker will search the public subgraph D, relying on all public social links and other public personal attributes (such as place of residence and work, educational affiliations, hobbies, etc.). We assume this attacker to have prior knowledge on the values of a subset A'_t of t's personal attributes, that he will use to navigate towards the target. As the attacker will reach the target through the target's social links (friends, friends of friends, ...), he will also discover at least one friend of the target, which can be useful for friend-based inference attacks [8, 33, 42]. Finally, note that the attacker we consider in this work is passive, in that he does not subvert any user account or interact with other OSN users, e.g., to create social ties with them.

4 Approach

We present here our navigation attack and algorithm. This attack is generic in order to apply to any attribute-enhanced OSN. We suppose that the attacker cannot rely on any search directory to find the target or to jump towards any user close to the target and that the navigation's starting point is randomly selected. This helps us evaluate the feasibility of a navigation attack in the worst-case scenario, and provide an upper-bound on the number of nodes that need to be crawled before reaching a target in general. In Subsec. 6.2, we nevertheless show how the attacker can take advantage of search filters to quicken the navigation.

In the generic scenario, the attacker navigates from user to user through public social links, until he reaches the target. He makes an informed decision

[4] Note that Facebook now also allows asymmetric social links, by enabling users to become subscribers of other users.

Algorithm 1 TargetedCrawler

```
1:  $F \leftarrow s$ % Initializing the frontier with the source user
2:  $E \leftarrow \varnothing$ % The explored set is initially empty
3:  repeat
4:    if $F = \varnothing$ then
5:      Failure
6:    else
7:      Select the user $u^* \in F$ with the lowest estimated cost to the target $t$ and
        remove it from $F$
8:      $E \leftarrow u^*$
9:      if $t \in \Gamma(u^*)$ then
10:       Return $t$'s profile and the path from $s$ to $t$
11:     else
12:       for all $u \in \Gamma(u^*)$ do
13:         $c_u = d_{\mathrm{hop}}(s,u) + d_{\mathrm{rem}}(u,t)$
14:         if $u \notin F$ AND $u \notin E$ then
15:           $F \leftarrow (u, c_u)$
16:         else if $u \in F$ AND $c_u < c_u^{\mathrm{old}}$ then
17:           $c_u^{\mathrm{old}} = c_u$
18:           Replace the former parent of $u$ by $u^*$
19:         end if
20:       end for
21:     end if
22:   end if
23: until $t$ reached
```

about the next user to visit by relying on information publicly revealed by users at each hop towards the target and on his prior knowledge about the target. Whereas in Milgram's experiment every participant in the chain could rely on his own local information about his acquaintances to make a decision about the next user to select, the attacker here relies on global information bounded by the attributes publicly revealed by users on the path. Our navigation attack is represented by Algorithm 1, called TargetedCrawler. This generic algorithm relies on a heuristic model inspired by A^* search [17].

The TargetedCrawler's inputs are (i) the source user s, from which the attacker will start crawling, (ii) the target user t that he has to reach, (iii) a subset of the target's attributes $A'_t \subseteq A_t$ known a priori by the attacker, (iv) the distance functions for each attribute, and (v) the priority of the attributes. The priorities depend essentially on the OSN and on the prior knowledge about the target's attributes. For instance, we will give higher priority to profession or workplace attributes in job-oriented OSNs (such as LinkedIn), to interests in microblogging OSNs (like Twitter), or to geographical attributes for mobile OSNs. The highest- and lowest-priority attributes will be represented as A^1 and A^N, respectively. The algorithm outputs t's profile and the shortest discovered path from s to t.

The total estimated cost c_u (line 13) from the source to the target at some node u on the path is divided into (i) the cost from the source to u, $d_{\mathrm{hop}}(s,u)$

(hop distance), and the estimated remaining cost from u to the target, $d_{\text{rem}}(u,t)$, that is expressed as

$$d_{\text{rem}}(u,t) = \begin{cases} k_h d_h(A_u^h, A_t^h) & \text{if } d_j(A_u^j, A_t^j) = 0 \ \forall j < h \\ k_1 d_1(A_u^1, A_t^1) & \text{otherwise} \end{cases} \tag{1}$$

where $d_h(A_u^h, A_t^h)$ is the distance function between users u and t in the attribute h (attribute with h^{th} priority). The distance functions can be represented by (i) binary values (e.g., 0 or 1 for last names), (ii) real values (e.g., difference for ages, or geographical distance for locations), or (iii) integers based on hierarchical decompositions (e.g., half the tree distance for tree-based hierarchies). k_h is a normalization parameter translating the attribute distance into a hop distance. k_h should decrease with h, as the more attributes we share, the closer to each other we should be. With d_{rem}, the targeted crawler will reach a user sharing the same first-priority attribute as the target before considering the second-priority attribute, then reach a user sharing a second-priority attribute before considering the third-priority attribute, and so on. We conjecture that OSN users share certain categories of attributes more than others (depending on the OSN) and that these attributes affect the way users cluster different OSNs. Thus, in order to increase the search efficiency, we prioritize different categories of attributes depending on the type of OSN.

5 Experiments

As the current largest OSN (1.1 billion users as of March 2013), Facebook is the most representative candidate for evaluating our attack. Moreover, its privacy policies are notoriously designed to encourage public disclosure: the default policy for many important user attributes is *everybody*, i.e., full public visibility.[5] We also implemented our attack in Google+ in order to validate our findings in Facebook. This OSN is now the second largest OSN, after Facebook [40], and shares many privacy features with Facebook. It also reveals the users' social links by default but, contrary to Facebook, allows users to be not searchable by name.

5.1 Implementation in Facebook and Google+

Gathering Source-Target Pairs Before beginning the navigation attack, we had to collect source users from which to start and target users to be reached. To further evaluate the paths' symmetry, we chose to select pairs of users that would act both as source and target. In order to have representative and meaningful results, we wanted to avoid sampling biases as much as possible. Unfortunately, as Facebook and Google+ IDs are encoded over 64 bits, there is a very small probability that a randomly generated ID corresponds to an existing profile.

[5] As of this writing, this is the case for the following attributes: current city, hometown, sexual orientation, friend list, relationship status, family, education, work, activities, as well as music, books, movies, and the sports users like.

For this reason, to gather source and target profiles, we decided to sample on the Facebook directory, as in [7]. The Facebook directory[6] has a tree structure, and profiles are sorted in first-name alphabetical order. The first layer of the tree is divided into Latin characters and non-Latin characters. Then, all subsequent layers are divided by alphabetical order into at most 120 subcategories, until the fifth layer, where we can actually select users' profiles. At each layer of the directory tree, we randomly selected one branch, until we reached the last layer, where we randomly selected one profile. Unfortunately for us, Google+ does not provide such a public directory. Thus, we decided to sample source and target users by relying on a random walk method. Our method starts by walking through 50 different profiles in order to reach a random profile in the network [34]. Once we have reached this profile, we select a node with a probability inversely proportional to its (bidirectional) degree, to be added to the source-target set. This probability compensates the random-walk bias towards high-degree nodes [11]. Finally, we only retain profiles with at least two publicly accessible attributes, assuming these to be part of the attacker's prior knowledge.[7] We discuss the representativeness of our target set in Subsection 5.2.

Navigating in Facebook and Google+ Because of the very limited Facebook API, we had to implement our own crawler of users' friend lists. With the standard HTTP request to access the friend list, Facebook provides only the first 60 friends of a user. Then, it dynamically provides the rest of the friends if the Web user scrolls down the friend list's page. While the user is scrolling down, his Web browser actually sends an Ajax request to get the subsequent 60 friends in the friend list. The server replies in about 2 seconds with a JSON (JavaScript Object Notation) object that contains the next 60 friends in the list. We parsed the list of user IDs of each JSON object, as well as the additional piece of information (if any) provided right below each friend's name that would be used for the navigation. We also implemented our own crawler for Google+. We could get both of all outgoing and incoming social links with only two HTTP requests. Both requests returned a JSON object with the social links (names), and some attributes (including location, employer, education) useful for the navigation.

Several lessons can be learned from previous work on navigation in social networks: (i) Geography and occupation are the two most crucial dimensions in choosing the next hop in a chain [21]; (ii) geography tends to dominate in the early stages of routing [9]; (iii) adding non-geographic dimensions once the chain has reached a point geographically close to the target can make the routing more efficient [38, 41]; and (iv) seeking hubs (highly connected users) seems to be effective in some experiments [4, 38] and to have limited effect in others [9]. As Facebook and Google+ share many properties with real social networks, we incorporate these findings into our navigation attack in order to maximize its

[6] http://www.facebook.com/directory

[7] This does *not* mean that a target without any publicly available attributes could not be found. We need this information here to replace the prior knowledge the attacker is assumed to have.

efficiency. We select location (*current city* or *hometown*) as the first-priority attribute in Algorithm 1, and education, employer/workplace, and last name as second-priority attributes. We make this choice also because of the OSN structure and design. All aforementioned attributes are those most publicly shared by the Facebook and Google+ users. Location (*current city* or *hometown*), *education* and *work* are publicly revealed by around 35%, 30%, and 25% of the Facebook users, respectively [7, 14]. In Google+, location, education, and employer are publicly shared by 26%, 27%, and 21% of the users, respectively [28]. Moreover, all these attributes are directly available from the social links' JSON objects, thus hindering us from crawling all friends' profiles individually, and thus dramatically decreasing the number of HTTP requests and crawling time.

We propose relying on two different types of distance function to evaluate the similarity between two locations. The first metric is computed as half the tree distance, where the tree is defined by a discrete geographical hierarchy: $d_1(A_u^1, A_t^1)$ is equal to 3, 2, 1, or 0, if user u shares a continent, a country, a region/state or a city, respectively, with the target t. $d_1(A_u^1, A_t^1) = 4$ if u and t are from different continents. The second distance metric relies on the real geographical distances between two locations and $d_1(A_u^1, A_t^1)$ is then defined as

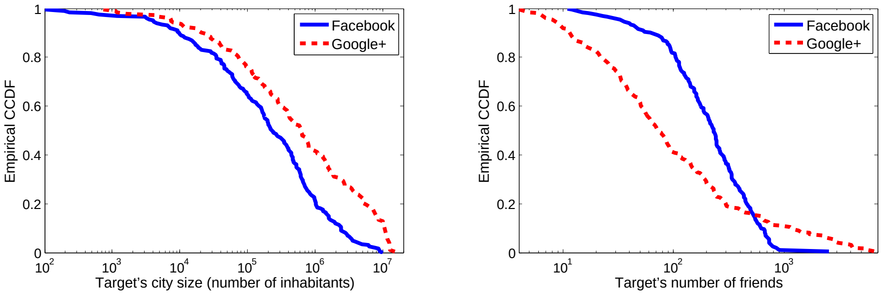
$$d_1(A_u^1, A_t^1) = \max(0, \log(d_{\text{geo}}(u, t)/\alpha)) \tag{2}$$

where the logarithm is base-10, d_{geo} is the great-circle distance (in km), and α is a normalization constant set to 1 km. We notice that this distance is very close to the discrete-hierarchy distance (first metric). In order to infer detailed geographical information from any location attribute, we relied on GeoNames[8], a Web service with a database containing over 10 million geographical names. More precisely, we used GeoNames (i) to find the region, country and continent associated with a city in the first distance metric and (ii) to compute the distance between two locations in the second metric. k_1 is set to 2 to get a maximal (theoretical) hop distance of around 8.

We give all non-geographical attributes second priority. We make these design choices mainly because we can only access a single attribute in the Facebook users' friend lists (below each friend's name). These structural constraints, imposed by the OSN architecture, lead us to trade off some of Algorithm 1's steps against efficiency. Moreover, we make use of a binary distance function for these second-priority attributes (0 if two attributes match, 1 otherwise) because (i) we believe it is more efficient to directly select users based on whether they share the same attribute with the target once we have reached the same city, and (ii) it is particularly complex to build more elaborate distance functions for last names, employers, high schools or universities. k_2 can be set to any number strictly smaller than 2; we chose $k_2 = 1$.

For simplicity, we verify whether we have reached the target profile by checking his ID or alias, which both uniquely identify users. An attacker who is not supposed to know such identifiers will have to check the target's first and last names that, in addition to the location, should uniquely identify most of the

[8] http://www.geonames.org/



(a) (b)

Fig. 1. Empirical complementary cumulative distributions of (a) the targets' city sizes, and (b) the targets' degrees.

people. In case there are multiple matching targets, the attacker could, for instance, just check the profile pictures of these few potential targets in order to select the correct target. Facial recognition could be further used to automatize the targets' check for targets making use of pseudonyms.[9]

5.2 Dataset Description

We ran our experiments on Facebook from April to November 2012, not too intensively, with a crawler having a behavior similar to an energetic human user, in order to avoid overloading the system. Despite this, we attempted to reach 200 targets, collecting approximately 393k different friend lists. We also targeted 200 different users in Google+, during Spring 2013, collecting 398k friend lists. For the Google+ crawler, we took similar precautions as for Facebook.

In both Facebook and Google+, we gathered targets in 42 different countries spread over all continents. North America encompasses 33.5% of the targets in Facebook and 44% in Google+, Asia 26% in Facebook and 31% in Google+, Europe 18% and 15%, South America 13.5% and 8%, Africa 7.5% and 1%, and Oceania 1.5% and 1%. The continent distribution is quite close to the actual distribution of users' continents, except for North America that is a bit overrepresented with respect to Europe and Asia. USA represents 26% of the targets in Facebook, followed by Indonesia, Brazil, and India, with 9.5%, 8.5%, and 8%, respectively. Almost the same sequence appears in Google+, with USA representing 38% of the targets, India 13%, Brazil 4%, and Indonesia 4%.

Regarding the targets' cities, we can notice in Figure 1(a) that the populations' distributions of Facebook and Google+ follow a similar shape, Google+'s targets living in cities with slightly more inhabitants than Facebook's. The average and the median city populations are equal to 870k and 233k, respectively, in Facebook, and to 2.6M and 440k, respectively, in Google+.

[9] Face recognition has been shown to be very accurate and efficient for subject reidentification in OSNs [3].

Table 1. Success rates and numbers of crawled nodes for all continents.

	Facebook			Google+		
Continent	% success	# nodes: mean	median	% success	# nodes: mean	median
North America	71.6	1,065	467	67.1	668	272
Asia	51.9	1,061	658	49.2	565	179
Europe	86.1	513	144	53.3	348	72
South America	59.3	1,275	445	56.3	667	628
Africa	60	1,500	1,608	67	805	100
Oceania	66.7	2,270	553	100	92	14

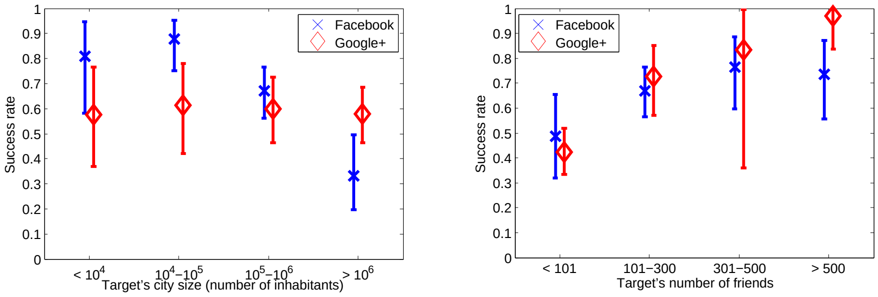
Regarding the targets' degrees (friends' or social links' numbers), we clearly notice a phase transition in the degree distribution (Fig. 1(b)) in Facebook, which is very similar to the one shown in [39]. Moreover, the average target has 291 friends, which is fairly close to the global average that was around 278 in April 2012 according to [16]. The targets' degree distribution is more scattered in Google+, with more targets having degrees smaller than 100 and greater than 1000. The median number of social links is equal to 71, smaller than Facebook, but its average is 424, greater than Facebook. It is hard to link these numbers with other studies, as Google+ is a recent OSN evolving rapidly [28]. The geographical distance between sources and targets is quite uniformly distributed between 450 km (shortest distance) and 18,962 km (longest distance) in Facebook, and between 285 km and 15,814 km in Google+.

6 Results

In this section, we will first exhibit the results of our generic navigation attack, showing its success rate and efficiency. We will also provide some explanations for the failed cases. We will then mention how, by using some search filters, we can drastically reduce the crawling effort.

6.1 General Results

Our objective is *not* to launch a brute-force attack by crawling millions of nodes, which would demand a lot of resources. We rather aim to develop an algorithm that can reach a specific target in the network in a limited amount of time. For this reason, we decided to stop the attack after a certain number of crawled nodes, even if the frontier F is not empty. We choose a limit of 4,000 users, which takes about 14 hours in Facebook (much slower than in Google+). We assume this is the maximum bearable time for an attacker attempting to reach someone in Facebook and, for consistency, we keep the same limit with Google+. Despite this limit, our attack successfully reaches its target in 66.5% of the cases in Facebook, and 59% of the cases in Google+. Using the Clopper-Pearson interval in order to evaluate the confidence interval for this success rate, we find that 95% of the users are reachable with a success rate in the intervals $[59.5\%, 73\%]$



(a) (b)

Fig. 2. Success rates (and their 95% confidence intervals) with respect to (a) the target's city size, and (b) his number of friends.

and $[52\%, 66\%]$ for Facebook and Google+, respectively. The Clopper-Pearson interval is an exact method for calculating binomial confidence intervals. It is quite conservative, thus the interval above might be wider than necessary in order to achieve 95% confidence. Table 1 shows the success rates, average and median numbers of crawled nodes, for each continent.

We notice that the North American targets are reached quite successfully in both OSNs, whereas reaching Asian users is more challenging. We also note that European targets are reached very successfully in Facebook but not in Google+. Figure 2 helps us understand these discrepancies. In particular, Figure 2(a) shows that in Facebook the success rate drops with the size of the target's city, but not in Google+. We note in Figure 2(b) that the success rate increases with the target's number of friends, especially in Google+. Lower success rates in Facebook can be explained by comparing the average numbers of inhabitants of the continents. We find that European and North American city populations have averages far below 1M (217k and 449k, respectively), whereas Asia, South America and Africa have average city sizes close to or above 1M (925k, 1.83M, and 2.46M, respectively). This lower success rate is certainly due to the fact that, in large cities, our algorithm has to crawl more nodes in order to cover all the users living in these cities. Our 4,000-node limit is certainly too low for such cities. However, this does not seem to explain the difference in success rates in Google+. This is probably due to the fact that Google+ is more recent and smaller than Facebook, there are less people publicizing the same city, hence fewer people to potentially crawl. The number of friends of the targets seems to have the highest impact on the success rate in Google+. For instance, the median number of friends in Europe is equal to 33, whereas it is equal to 81 in North America. This is certainly due to the young age of Google+, and lower rate of adoption by European users. We must also mention that source users have no effect on the success rate: all crawls successfully navigate out of the source neighborhood, and the large majority of them (92% in Facebook and 93.5% in Google+) reach the target's city.

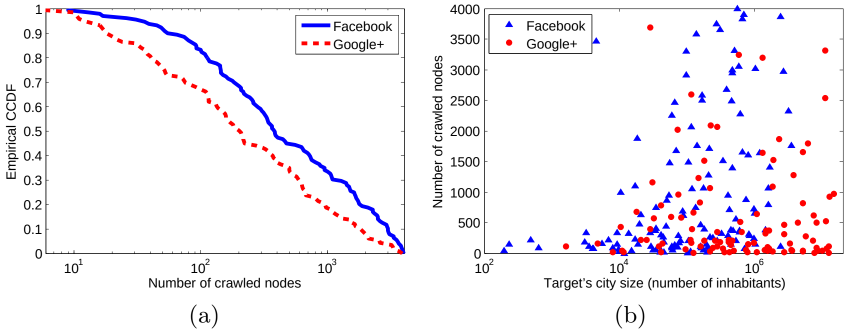
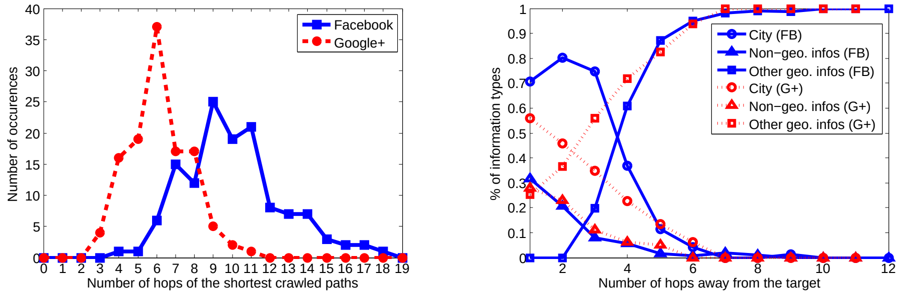


Fig. 3. (a) Empirical CCDF of the number of crawled nodes in successful cases, (b) number of crawled nodes with respect to the target's city size (number of inhabitants).

We evaluate the nodes' efficiency by looking at the number of nodes crawled in our searches. Crawling a node in our experiment means crawling a user's friend list, not his personal profile. On average, 983 and 591 nodes needed to be crawled before a target could be reached, in Facebook and Google+, respectively. Half of the targets were attained in 380 and 291 or fewer nodes in Facebook and Google+, respectively. European targets were especially rapidly reached, after 513 and 348 nodes on average, half of the targets being found after less than 144 and 72 crawled nodes in Facebook and Google+, respectively. We see in Figure 3(b) that the number of crawled nodes is (positively) correlated to the target's city size. This is again due to the fact that more nodes will be seen in larger cities, thus the target is reached after a higher expected number of crawled nodes. Moreover, for all failed and successful cases, on average 44 and 28 nodes had to be crawled before we reached a user in the target's city, and in half of the searches we found a user living in the target's city in less than 13 and 8 crawled nodes, in Facebook and Google+, respectively. This shows that our search algorithm makes use of long-range social links to efficiently reach the target's city, and that the most challenging part of the search is the navigation within the target's city, when we have to narrow down the search using second-priority attributes.

From each subgraph crawled during a successful attack, we reconstructed the shortest discovered path from the source to the target. Figure 4(a) illustrates the distribution of the shortest discovered path lengths. We notice that it goes from 4 to 18 hops in Facebook, with most of shortest paths being between 9 and 11-hops long. This is around twice the distance found in [6] with the knowledge of the full social graph. The shortest paths are between 3 and 11 hops in Google+, most of them being 6 hops long. This result is similar to the diameter obtained in [12], where 90% of the pairs were separated by a distance of 5, 6 or 7 hops.

We show in Figure 4(b) the evolution of the information that displayed by the nodes on the shortest path (SP). It shows that the city is especially useful 3, 2, and 1 hop(s) before the target, for both OSNs. At 4 (and more) hops from the



(a) (b)

Fig. 4. (a) Histograms of the shortest discovered path lengths within the crawled subgraphs, and (b) evolutions of the information types used to navigate towards the target.

target, other (non-local) geographical attributes are used to navigate towards the target. We also note that the crawler starts using other types of attributes (education, work, or last name) 4 hops before the target (certainly once we have reached the target's city) and their influence is increasing while getting closer to the target. At the latest hop before the target, the city is represented in 70% of cases in Facebook and 56% in Google+, non-geographical information representing around 30% of cases in both OSNs. This shows that geographical information remains crucial, but also that other types of information can still be useful when we get close to the target, as it was already mentioned in [38]. Finally, we note that 25% of the targets in Google+ were found from a last hop sharing no similar attributes with the target. These targets were reached from a last user who is geographically close (at a median distance of 32 km) but does not share the same location.

6.2 Jumping towards the Target

Facebook provides an additional feature in order to help people find their acquaintances in the network: It allows users to apply search filters on location, education or workplace. We did not want to rely extensively on this feature for our navigation attack because we wanted to keep it generic and applicable to other OSNs. However, we show here that the attacker can take advantage of Facebook's search filters to facilitate his attack.

We search for the last names and the cities of the targets using the Facebook search filters, and then crawl the friend lists of the users found by the search directory. We search for last names because users sharing same last names are more likely to be relatives, thus to be friends. Our targets can also appear in the users found by the search filters, as we chose targets that are in the Facebook directory for our experiments. Searching for the last names and the cities of our targets, we directly find the targets in 49.5% of the search results. As targets are assumed to not be in the directory, we remove them from the list of users to be crawled. At least 10 users satisfying the search criteria are found in 30% of the filtered searches, and the search requests output no user in 15% of the cases. By crawling only the friend lists of users found by our filtered search, we reach the targets with a success rate of 16.5%.

7 Countermeasures

Countermeasures should logically be developed and implemented by the OSN operators themselves. An obvious solution, already advanced in [37], is to set the visibility policy as the intersection of visibility policies selected by all users involved in the published information. Although it is difficult to force a friend to change his privacy settings on his personal attributes, it is possible to enforce his social links' privacy policy. Choosing the intersection of both users' policies on social links would mean that a user electing to reveal his social links to his friends, or friends of friends only, would automatically enforce non-public social links for his own friends. It would prevent any curious stranger from accessing his profile by using his friends' friend lists. OSN operators could also prevent anyone from publicly showing his social links, as it is the case in LinkedIn. They could at least design non-public default privacy settings on social links. Detailed formal requirements to protect multilateral privacy are presented in [15].

If the OSN operators themselves do not re-design their privacy policies, the users could also take action. The first option is to change the default privacy settings on social links to more restrictive settings. For this option though, users must collectively deviate from the default policy in order for it to be efficient. Finally, if more users decided to hide their personal attributes (such as city, education, ...), the attacker's ability to navigate efficiently in the social graph would decrease, thus reducing the threat presented in this paper.

8 Conclusion

We believe our navigation attack to be the first to rely on social links to find a target's profile. We describe a search algorithm that relies on public attributes of users and distance heuristics, and that discovers 66.5% and 59% of the targeted users, in a median number of crawled nodes smaller than 400 and 300, in Facebook and Google+, respectively. Moreover, the targets' cities are reached in more than 90% of the cases, in a median number of 13 and 8 crawled nodes, respectively, showing the efficiency of geographic navigation in these OSNs. The navigation within the targets' cities, which relies on more attributes, is less efficient and successful. One important reason for the failed cases is the privacy behaviors of the target's friends: the more friends with public attributes and social links, the more likely the target is to be found.

In future work, we plan to propose other search algorithms, especially for once we have reached the target's city. We also plan to apply our navigation attack to other OSNs, and build a theoretical model to support our experimental findings.

Acknowledgements

We would like to thank Abdelberi Chaabane, Mohamed Ali Kaafar, Roger Michoud and Kevin Fahy for their help in the implementation of the crawlers. We are also very grateful to Jens Grossklags for helpful preliminary discussions, as well as Nevena Vratonjic, Igor Bilogrevic and Kévin Huguenin for their feedback.

References

1. Acquisti, A., Gross, R.: Imagined communities: Awareness, information sharing, and privacy on the facebook. In: PET (2006)
2. Acquisti, A.: An experiment in hiring discrimination via online social networks. Berkeley (April 2012)
3. Acquisti, A., Gross, R., Stutzman, F.: Faces of facebook: Privacy in the age of augmented reality. BlackHat USA (2011)
4. Adamic, L., Lukose, R., Puniyani, A., Huberman, B.: Search in power-law networks. Physical review E 64 (2001)
5. Aiello, L.M., Barrat, A., Schifanella, R., Cattuto, C., Markines, B., Menczer, F.: Friendship prediction and homophily in social media. ACM Transactions on the Web (TWEB) 6 (2012)
6. Backstrom, L., Boldi, P., Rosa, M., Ugander, J., Vigna, S.: Four degrees of separation. In: Proceedings of the 3rd Annual ACM Web Science Conference (2011)
7. Chaabane, A., Acs, G., Kaafar, M.: You are what you like! Information leakage through users' interests. In: NDSS (2012)
8. Dey, R., Tang, C., Ross, K., Saxena, N.: Estimating age privacy leakage in online social networks. In: INFOCOM (2012)
9. Dodds, P., Muhamad, R., Watts, D.: An experimental study of search in global social networks. Science 301, 827–829 (2003)
10. Finder, A.: For some, online persona undermines a résumé. The NY Times (2006)
11. Gjoka, M., Kurant, M., Butts, C.T., Markopoulou, A.: A walk in Facebook: Uniform sampling of users in online social networks. Tech. rep., UC Irvine (2011)
12. Gong, N.Z., Xu, W., Huang, L., Mittal, P., Stefanov, E., Sekar, V., Song, D.: Evolution of social-attribute networks: measurements, modeling, and implications using Google+. In: IMC (2012)
13. Gross, R., Acquisti, A.: Information revelation and privacy in online social networks. In: WPES (2005)
14. Gundecha, P., Barbier, G., Liu, H.: Exploiting vulnerability to secure user privacy on a social networking site. In: KDD (2011)
15. Gurses, S.: Multilateral privacy requirements analysis in online social network services. Ph.D. thesis, KU Leuven (2010)
16. Hachman, M.: Facebook now totals 901 million users, profits slip. http://www.pcmag.com/article2/0,2817,2403410,00.asp (April 2012)
17. Hart, P., Nilsson, N., Raphael, B.: A formal basis for the heuristic determination of minimum cost paths. Systems Science and Cybernetics, IEEE Transactions on 4, 100–107 (1968)
18. He, J., Chu, W., Liu, Z.: Inferring privacy information from social networks. Intelligence and Security Informatics pp. 154–165 (2006)
19. Jain, P., Kumaraguru, P.: Finding nemo: Searching and resolving identities of users across online social networks. arXiv preprint arXiv:1212.6147 (2012)

20. Johnson, C.: Project Gaydar: An MIT experiment raises new questions about online privacy. Boston Globe (2009)
21. Killworth, P., Bernard, H.: The reversal small-world experiment. Social networks 1, 159–192 (1979)
22. Kótyuk, G., Buttyán, L.: A machine learning based approach for predicting undisclosed attributes in social networks. In: SESOC (2012)
23. Krishnamurthy, B., Wills, C.: Characterizing privacy in online social networks. In: WOSN (2008)
24. Krishnamurthy, B., Wills, C.: On the leakage of personally identifiable information via online social networks. In: WOSN (2009)
25. Lattanzi, S., Panconesi, A., Sivakumar, D.: Milgram-routing in social networks. In: Proceedings of the 20th international conference on World wide web (2011)
26. Liben-Nowell, D., Novak, J., Kumar, R., Raghavan, P., Tomkins, A.: Geographic routing in social networks. Proceedings of the National Academy of Sciences of the United States of America 102 (2005)
27. Lindamood, J., Heatherly, R., Kantarcioglu, M., Thuraisingham, B.: Inferring private information using social network data. In: WWW (2009)
28. Magno, G., Comarela, G., Saez-Trumper, D., Cha, M., Almeida, V.: New kid on the block: Exploring the Google+ social graph. In: IMC (2012)
29. Mao, H., Shuai, X., Kapadia, A.: Loose tweets: an analysis of privacy leaks on twitter. In: WPES (2011)
30. McPherson, M., Smith-Lovin, L., Cook, J.M.: Birds of a feather: Homophily in social networks. Annual review of sociology pp. 415–444 (2001)
31. Milgram, S.: The small world problem. Psychology today 2, 60–67 (1967)
32. Mislove, A., Marcon, M., Gummadi, K., Druschel, P., Bhattacharjee, B.: Measurement and analysis of online social networks. In: IMC (2007)
33. Mislove, A., Viswanath, B., Gummadi, K., Druschel, P.: You are who you know: inferring user profiles in online social networks. In: WSDM (2010)
34. Mohaisen, A., Yun, A., Kim, Y.: Measuring the mixing time of social graphs. In: IMC (2010)
35. Preston, J.: Seeking to disrupt protesters, Syria cracks down on social media. http://www.nytimes.com/2011/05/23/world/middleeast/23facebook.html?_r=1 (May 2011)
36. Sullivan, B.: Govt. agencies, colleges demand applicants' facebook passwords. http://redtape.msnbc.msn.com/_news/2012/03/06/10585353-govt-agencies-colleges-demand-applicants-facebook-passwords?chromedomain=usnews (2012)
37. Thomas, K., Grier, C., Nicol, D.: unFriendly: multi-party privacy risks in social networks. In: PETS (2010)
38. Travers, J., Milgram, S.: An experimental study of the small world problem. Sociometry pp. 425–443 (1969)
39. Ugander, J., Karrer, B., Backstrom, L., Marlow, C.: The anatomy of the Facebook social graph. Tech. rep. (2011)
40. Watkins, T.: Suddenly, Google Plus is outpacing Twitter to become the world's second largest social netwo. Business Insider (2013), http://www.businessinsider.com/google-plus-is-outpacing-twitter-2013-5
41. Watts, D., Dodds, P., Newman, M.: Identity and search in social networks. Science 296, 1302–1305 (2002)
42. Yamada, A., Kim, T., Perrig, A.: Exploiting privacy policy conflicts in online social networks. Tech. rep. (2012)
43. Zheleva, E., Getoor, L.: To join or not to join: the illusion of privacy in social networks with mixed public and private user profiles. In: WWW (2009)