PhD Thesis Approval

THESIS TITLE: User Data Sharing in Online Services

AUTHOR: Yabing Liu

Ph.D. Thesis Approved to complete all degree requirements for the Ph.D. Degree in Computer Science.

Thesis Advisor

Thesis Reader

Thesis Reader

Thesis Reader

Graduate School Approval:

Director, Graduate School

Copy Received in Graduate School Office:

Recipient's Signature

Distribution: Once completed, this form should be scanned and attached to the front of the electronic dissertation document (page 1). An electronic version of the document can then be uploaded to the Northeastern University-UMI website.
User Data Sharing in Online Services

A Dissertation Presented
by

Yabing Liu

to

The College of Computer and Information Science
in partial fulfillment of the requirements
for the degree of

Doctor of Philosophy
in

Computer Science

Northeastern University
Boston, Massachusetts

November 2016
To my family.
# Contents

List of Figures .......................................................... v

List of Tables .......................................................... viii

Acknowledgments ......................................................... x

Abstract of the Dissertation ........................................... xi

1 Introduction .......................................................... 1
   1.1 Contributions ..................................................... 3
   1.2 Outline .......................................................... 6

2 Background .......................................................... 7
   2.1 Online social networks (OSNs) ................................ 7
   2.2 Online advertising .............................................. 9
   2.3 User privacy ...................................................... 11

3 Related Work ......................................................... 14
   3.1 Online advertising .............................................. 14
   3.2 User privacy ...................................................... 16
      3.2.1 Privacy issues on OSNs ................................. 16
      3.2.2 Privacy enhancing techniques ......................... 18
      3.2.3 Detection of privacy leaks .............................. 20

4 Analyzing and Improving Facebook Privacy Settings ............... 23
   4.1 Motivation ....................................................... 23
   4.2 Methodology ..................................................... 25
      4.2.1 Approach ................................................... 25
      4.2.2 Data statistics ........................................... 27
      4.2.3 User demographics ....................................... 28
   4.3 Analysis of privacy settings .................................. 28
      4.3.1 Existing privacy settings .............................. 29
      4.3.2 Desired privacy settings ............................... 30
      4.3.3 Improving privacy tools ............................... 33
List of Figures

4.1 Screenshot of our survey. Each user was asked about 10 different uploaded photos. 26
4.2 Cumulative distribution of the fraction of each user’s content that remains at the
default privacy setting, for the five different content types. The distribution differs
across the content types; with many users only having changed the settings for a
subset of their content. 30
4.3 Cumulative distribution of the sizes of observed friend lists. 33
4.4 The 8 automatically detected groups of friends from one of the authors’ Facebook
social network. Nodes represent the author’s friends and links exist between pairs of
friends who are also friends. Nodes with the same color are automatically grouped
together by the community detection algorithm. 33
4.5 Cumulative distribution of the maximum normalized conductance values per user
in Facebook New Orleans [148]. In 40% of cases BGLL creates communities
with conductance greater than 0.2, and all methods create communities with almost
equally good conductance values (the results are similar for the other two Facebook
networks). 37
4.6 Cumulative distribution of the size of the communities per user in Facebook New
Orleans. In more than 90% of cases, BGLL creates communities with fewer than 20
members. 38
4.7 Screenshot of user interface of the application with proposed new friend lists. 39
4.8 Screenshot of (a) an existing friend list with proposed additions and (b) a newly
proposed friend list with merge friend list functionality. 40

5.1 Screenshot of Facebook’s ad creation webpage, showing the suggested bid (bot-
tom right) for the selected targeting parameters. We programmatically collect the
suggested bid information. 45
5.2 Example query targeting U.S. males between 21 and 30. Also shown is the suggested
bid response (in JSON format). 46
5.3 1,000 suggested bids for three different sets of targeting parameters with massively
different audience sizes (note the log-scale on the y-axis). Significant variance is
observed, as well as a skewed distribution, across all three countries. 47
5.4 Audience size vs. coefficient of variation (standard deviation divided by mean) of
suggested CPM bids for all 204 countries. No correlation is observed for CPM
minimum, median, or maximum. 48
5.5 CPM maximum values for 100 successive suggested bids targeting US users, queried from 2 different accounts at the same time. The prices received by the two different accounts show no correlation.

5.6 Example probability distribution function of CPM maximum values for 20,000 successive suggested bids (US 25-year-old females interested in computer programming; 179,760 users). This distribution fails statistical tests for multiple common distributions, suggesting the absence of random noise.

5.7 Diagram of Hypothesis 3: Sampling winning bids. To generate each suggested bid, Facebook selects a sample of bids from the set of all bids, and returns the median, minimum, and maximum from the sample.

5.8 (a) CPM median values and (b) CPM maximum values before, during, and after our three advertising campaigns targeting Seychelles users (the shaded region represents the time of the ad campaigns. We observed that our advertising campaigns were quickly reflected in the suggested bids.

5.9 Convergence of different numbers of collated suggested bids towards overall CPM minimum, median, and maximum values for each of the 204 countries (each country is represented by a line in each graph). The average value across all countries is shown as the dark dashed line. Very quick convergence is observed for minimum and median, as expected.

5.10 Scatterplot of CPM median versus GDP per capita for the 204 countries Facebook supports. Labeled are some of the countries with high CPM median values. All but a few countries have ad markets with very low CPMs.

5.11 CPM maximum prices for all U.S. cities with population over 100,000. The color of each city corresponds to CPM maximum price (yellow to red represents increasing prices, and the size of the circle is in proportion to the number of Facebook users in each city.

5.12 (a) CPM median versus Age (b) Audience Size versus Age for three countries. CPM prices increase significantly with age for the United States and New Zealand, and massively different user populations are observed (note the log scale on the y-axis of the lower graph).

5.13 (a) Cumulative distribution of CPM median values for 129 interest categories for three countries. (b) Cumulative distribution of audience size for three countries (United States, New Zealand, and Antigua and Barbuda).

5.14 Long-term tracking of CPM median prices over three weeks for four different sets of targeting parameters that show different properties with distinct trends over time.

6.1 The cumulative distribution of the uniqueness ratio of all static domain-keys in ISP traffic dataset.

6.2 The cumulative distribution of number of domains that each user-value has been sent to in ISP traffic dataset.
6.3 Overview of our proposed approach, exemplifying the four steps in identifying domain-keys that may contain PI. In the Pre-processing step, domain-keys with too few values are filtered. In the Seed rules step, different rules are applied to the values in each domain-key; domain-keys with too few matching values are filtered out in the Filtering stage. Finally, in the Expansion step, the newly discovered values are shown in **bold**, and these may be used to help refine the seed rules.

6.4 The cumulative distribution of the number of unique tuples, users, and distinct values for each domain-key discovered in *ISP traffic* dataset.

6.5 The cumulative distribution of *ratio matched values* for domain-keys in *ISP traffic* dataset for each of the eight different seed rules. For each rule, only those domain-keys that have *at least one* matching value are considered. Different rules show different properties, but most show a “knee” in the curve at some threshold.

6.6 CDF of the ratio of values matched for email domain-keys discovered via seed rule.

6.7 Heat map of PI leaked by categories of domains.
List of Tables

2.1 Facebook’s targeting parameters made available to advertisers. ................................. 10
4.1 Existing privacy settings for all content items. The different content types possess similar privacy setting distributions, and the default (All Facebook users) is selected for the plurality of the items. ................................................................. 29
4.2 Comparison of the actual privacy settings and desired privacy settings for randomly-selected photos. The shaded cells represent instances where two are the same; this only occurs in 332 (37%, ± 3.14%) photos. When the two are different, they are more often shared with more users than desired (443 photos) than fewer users than desired (132 photos). ................................................................. 31
4.3 Comparison of the actual privacy settings and desired privacy settings for photos with non-default privacy settings. The shaded cells represent instances where two are the same; this only occurs in 296 (39%, ± 3.45%) photos. When the two are different, they are shared with more users than desired (254 photos) with approximately the same frequency as fewer users than desired (218 photos). ................................................................. 32
4.4 Number of nodes, links, and average user degree in the samples of the Facebook networks used to evaluate community detection algorithms. ................................. 37
4.5 Topological characteristics of authors’ 1-hop Facebook subgraph along with number of friend lists detected by Friendlist Manager, number of friend lists finally users create on Facebook and time taken by users to use Friendlist Manager for modifying and creating these friend lists on Facebook. ................................................................. 41
5.1 Comparison of the coefficient of variation of the CPM values for the three different countries. Significant variance is observed, especially for the CPM maximum values. 47
5.2 Comparison of Facebook’s ARPU and CPM median suggested bids. We scale all values relative to the US+Canada region. We observe the same ranking of regions. 53
6.1 Examples of different types of textual PI, and the breakdown of PI along different dimensions. ....................................................................................................................... 64
6.2 High-level statistics of our two datasets. ........................................................................ 66
6.3 Examples of regular expressions used for a subset of the seed rules. Some of the regular expressions require minor post-processing, such as a the “age range” PI category, where the second element of the range must be greater than the first. 72
6.4 Examples of values that both match and do not match the seed rules. We observe that values that do not match the seed rules may still identify potential leaks, and may help to refine the seed rules. .................................................. 74

6.5 Comparison of domain-keys selected by the seeded method and ground truth in Lab traffic dataset. ................................................................. 75

6.6 Examples of both true and false positives when using the key semantic method.  . 77

6.7 Comparison between our proposed seeded method and baseline key-semantic method. The seeded method has a dramatically lower false positive rate (13.6%, when disregarding postal code) than the baseline method. ................................. 78
Acknowledgments

First and foremost, I would like to record my deepest gratitude to my advisor, Alan Mislove, for supporting me, for encouraging and helping me, for all of his comments and suggestions and advice during my graduate career. None of this research would have been possible without him. His remarkable insights have greatly inspired me and enriched my growth as a student and a researcher. I want to thank Alan for having faith in me this entire time. For that and more, I’m deeply indebted to Alan, and I am and will always be grateful.

I also want to give my deep appreciations to other members of my thesis committee: Christo Wilson, David Choffnes, Han Hee Song, for their valuable feedback and encouragement in my research and thesis. Dr. Wilson has offered insight and advice for my research, and has been of great help for all of my publications we coauthored. Dr. Choffnes has patiently went through my thesis, and provided me a lot of insightful ideas and shaping up this thesis. And Dr. Song has provided nothing less than enthusiastic support and guidance during my internship at Narus.

Also, I want to give my special thanks to my other mentors at Narus: Ignacio Bermudez, Mario Baldi, and Alok Tongaonkar.

A number of other faculty members have given me insightful advice and perspective: Balan- chander Krishnamurthy, Krishna P. Gummadi, Dave Levin, Bruce Maggs, Aaron Schulman, and Ricardo Baeza-Yates. Many thanks to their constructive comments on my work. Additionally, it was my great pleasure to work with all my collaborators: Bimal Viswanath, Mainack Mondal, Chloe Kliman-Silver, Diego Saez-Trumper, Liang Zhang and Taejoong Chung.

I would like to thank all of my friends I have made at Northeastern during this unique journey. I am grateful for my friendship with Tim, Bochao, Yingying, and Jingjing. There were hard times when I felt exhausted physically and emotionally, they pulled me up and offered me help when I needed it.

Finally, I want to give my deepest heartfelt thanks to my family. I want to thank my father, Erxi Liu, my mother, Weidong Zhang and my brother, Yaying Liu for their love and support, which has gave me courage to explore the world all these years.
Abstract of the Dissertation

User Data Sharing in Online Services

by

Yabing Liu

Doctor of Philosophy in Computer Science
Northeastern University, November 2016
Dr. Alan Mislove, Advisor

Over the last decade, online services like online social networks (OSNs) have exploded in popularity. Today, users can access these services anytime and anywhere, via both traditional web sites and mobile applications. These services differ significantly from prior systems. For example, one fundamental difference between OSN services and the “traditional” web is that these services embody users as first-class entities, meaning users are tightly linked to content as well as other users. As another example, the sharing of user personal data has emerged as a popular activity over these services, in contrast to prior systems that primarily distribute publisher-generated data. As large amounts of such user-generated data has become accessible, many researchers have leveraged it to examine user activities, patterns of interaction, and the structure of human social networks. However, a full accounting of the privacy implications of new online services—especially OSNs—for end users remains elusive. Privacy is of paramount importance on such services, as the providers monetize user data to enable their site-specific advertising services; these allow third parties (e.g., advertisers) to target users directly (by taking advantage of user-provided data). Unfortunately, there is still little quantification of user-selected privacy settings; the difficulty users face when managing their privacy; the scope, characteristics, and operation of advertising markets that provide the economic basis for online services; and the extent to which services and third parties are automatically collecting user data.

In this thesis, we aim to develop novel measurement techniques to study the emerging privacy problems in the online ecosystem. We examine the user data sharing activities and the privacy implication of such behavior by focusing separately on each of the three primary players in the ecosystem on online services: end users, site operators, and third parties such as advertisers, with an ultimate goal to improve privacy for end users. To do so, we conduct the first large-scale measurement study to quantify the magnitude of the problem Facebook users have when managing privacy. Our
analysis identifies a significant potential to improve privacy controls. To test our ideas in practice, we develop *Friendlist Manager*, a Facebook application that reduces the user burden in automatically creating and maintaining their *friend lists*, or groups of friends that can be used as privacy basis for data sharing. Second, we explore how user data is being monetized by OSNs. We leverage the *suggested bid* feature provided to advertisers on many OSN services to examine how they utilize user data for their underlying advertising market. We also demonstrate how suggested bids can be used to explore the relative value of different user demographics and the overall stability of the advertising market. Finally, we explore how user data may leak from the providers of online services to third parties. As a first step towards investigating this potential privacy leak, we present and evaluate a new methodology to automatically detect when user data is transmitted in network traffic when using online systems from both web sites and mobile applications. Taken together, the techniques in this thesis allow researchers to measure and analyze how end users manage and share their uploaded content and the privacy challenges they face on such sites, to develop new privacy-preserving tools for content sharing, to examine how user data is monetized by OSN sites for their advertising markets, as well as to automatically detect when user data is transmitted in network flows to online services and third parties.
Chapter 1

Introduction

Online services like online social networks (OSNs [142], e.g., Facebook and Twitter), online content-sharing services (e.g., Flickr and YouTube), communication services (e.g., WhatsApp and Skype) have become very popular worldwide in recent years. Many of these services provide a new way for users to connect, communicate, and share content; today, many serve as the de-facto Internet portals for millions and even billions of users. Online services such as Facebook (over 1.65 billion monthly active users in 2016) [117], Twitter (over 310 million users) [118], YouTube (over 1 billion users) [162], and LinkedIn (over 400 million professional members) [119] are examples of some of the most popular online services today.

However, these online services, especially the OSN and content-sharing services are fundamentally different from the prior systems that preceded them—the “traditional” web (e.g., the New York Times’ web site, Yahoo!, or even many e-commerce sites). We begin by discussing two primary differences between the “traditional” web and the “modern” online services in terms of how they are architected, funded, and used.

First, the “modern” online services embody users as first-class entities. Users can publish new content, build their own profile pages, add other users as “friends”, and share content with their “friends”. In the prior systems that primarily distribute publisher-generated data, users only had limited interaction in the web, such as navigating to different web pages via hyperlinks. Furthermore, users typically did not have any interaction with other users on the web sites. In contrast, on online systems, direct user–user interaction is often the entire purpose of the system. As a result, the characteristics of end users (their activity, their friendships, the structure of their social network) is one of the primary determinants in the overall characteristics of the sites themselves.

Second, the sharing of user data on OSN and content-sharing sites has fundamentally changed
CHAPTER 1. INTRODUCTION

the economic underpinnings of these services. Prior to 2005—before Facebook and Twitter—the “traditional” web was funded by advertising in two typical forms: banner advertising, which entailed embedding an advertisement into a web page that can link to the web site of the advertiser, and search advertising, which targeted the user’s search keywords or queries from search engines such as Google, Yahoo!, and Bing. However, the OSN services work on a unique, new business model based on site-specific advertising services: the services themselves are made available free of charge to end users, in exchange for users accepting that service providers monetize user-uploaded data by enabling third parties such as advertisers and trackers to target platform users. Thus, in contrast to the “traditional” web that was (and is) funded via advertising based on Internet-wide advertising networks, many OSN services provide site-specific advertising that allow advertisers to target different user demographics directly.

So far, we have described the two fundamental differences between the “traditional” web and the online services that we focus on in this thesis. Next, we turn our attention to the three main entities that are involved in the new online services—end users, online service operators, and third parties (e.g., advertisers, trackers, and network operators)—and explore how their roles have changed in these services. We argue that the rise of new online services has significant consequences for the privacy implications for end users. Hence, it becomes important to understand how these new online services work, especially since their underlying privacy implications remain elusive. We discuss each of them in more details below.

First, the online services provide a new way for end users to upload and share their own content, resulting in a fundamental shift in the role end users play on the web. Today, instead of just being content consumers, individual end users are now required to be content creators and managers. For every single piece of content shared on sites like Facebook—every wall post, photo, status update, and video—the uploader must decide which of his friends, group members, and other Facebook users should be able to access the content. The per-user average of 130 friends and 80 groups and events [43]—compounded with the average 205 pieces of content uploaded per day [43]—has turned the task of simply managing access to content into a significant mental burden for many users. The privacy policies of online services are often too complicated for users to understand [101, 134], and the privacy controls are very limited as well [85, 94]. While major privacy violations and mismatched user expectations are likely to exist, previous studies focus on inferring privacy violations indirectly, the extent to which such direct privacy violations occur has yet to be quantified.

Second, as mentioned before, even though users can specify the privacy settings when sharing their content with other users or applications, they have few options to keep their content private
CHAPTER 1. INTRODUCTION

from the service providers. As a consequence, providers of OSN services are monetizing the user data for their targeted ad services. In essence, the sites are funded differently from prior systems (e.g., Google), where the ad networks were forced to infer user information from cookies, browsing history, and search terms. Instead, these new OSN-based ad services are provided with demographic information directly by end users, which makes them popular with advertisers, as the advertisers can target users directly (via profile attributes). OSN-based ad services are emerging to be a significant fraction of the online ad market; Facebook alone has achieved over $17 billion in ad revenue worldwide in 2015 and is still increasing [49]. However, there has been little academic study of the ad networks that enable such services, and online services have released very little data about their ad markets. Additionally, users have little visibility into what web sites do with their data for advertising as well as how much their personal data contributes towards the online services’ ad revenue.

Third, recent studies have shown that online services may deliberately expose user private data to third parties without the user’s understanding or control [67, 77, 97, 106]. Other third parties, such as Internet Service Providers (ISPs [72]), may get access to end user-uploaded data as it transits across their networks. Today, users may access these online services not only through the web sites on desktop and laptop devices, but also from both mobile browsers and dedicated applications. Online services are known to acquire many attributes from their users, including their profile information(e.g., their name, address, birth date, hometown, education and professional history, etc), uploaded content, activities, and interests. Similarly, mobile applications collect additional data about users, such as their unique device ID, location, contacts, call logs, and calendar entries. As a result, when we consider how online services are collecting data on users, we need to examine how users interact with these services via both the (desktop and mobile) web sites as well as via dedicated mobile applications. Thus, user privacy has emerged as an increasingly important issue in the age of online services. Overall, it is crucial to understand the extent to which the online services and third parties are automatically collecting end users’ personal data, to fully understand the scope of the privacy implications, and to help develop approaches to prevent privacy leaks.

1.1 Contributions

Given the motivation above, the overall thesis behind this work is

Using measurement of real-world users and systems, we aim to analyze the data sharing behavior of the three primary actors in online systems: end users, site operators, and third parties, to examine
CHAPTER 1. INTRODUCTION

The privacy implications when they share, leverage, and transmit user data, and to explore how we can improve privacy for end users.

In general, privacy may have different meanings in different context, and thus we have different ways to measure and quantify it. The definition of privacy we use in this thesis is “selective revelation of information about oneself” [74], and we are primarily interested when such information is shared, monetized, and transmitted in online systems. We give more concrete definitions of privacy in each context when we evaluate it in each chapter.

Below, we briefly discuss each of the three major contributions of this thesis.

The first contribution is that we examine how the role of end users has changed with online services, and the implications this has had on their privacy. The unique business model of online services assumes that users understand the tradeoff between the use of the services and data privacy. Unfortunately, there is little quantification of the incidence of incorrect privacy settings or the difficulty users face when managing their privacy. Under the circumstances, our research takes the first steps towards addressing the problem by analyzing the current state of privacy affairs on Facebook. In particular, we center our analysis around two questions: How close are the desired privacy settings to the actual settings that users have? In addition, is there potential to aid users in selecting the correct privacy settings for their content?

To answer these questions, we design a large-scale measurement study to quantify the magnitude of the problem of managing privacy. To examine whether users’ desired privacy settings differ from their existing settings, we deploy a survey. It is implemented as a Facebook application, which queries Facebook to select content to ask the user about, as well as collects the current privacy settings for the user-uploaded content. We find that the privacy settings match users’ expectations only 37% of the time, and when not matching, users almost always expose content to more users than expected. To explore the potential to assist users in selecting appropriate privacy settings, we demonstrate the usage of the Friend Lists feature by leveraging the structure of the social network. Facebook allows the user to group friends into specific friend lists, aggregating the interactions and content from all friends in each friend list. To this end, we develop Friendlist Manager, a Facebook application that can analyze a user’s 1-hop social network, detect new friend lists, as well as maintain the existing ones over time. The application can help users automate the creation and maintenance of meaningful friend lists, as a basis for privacy-sensitive content sharing. We made Friendlist Manager free for users to run, and as-of April 2015, over 5,000 Facebook users have used it, and over 2,800 of them have created or modified at least one friend list with it.
The second contribution of this thesis is that we examine how the role of site operators has changed with online services. Specifically, we look at how site operators leverage the user-uploaded data for their targeted ad services. Surprisingly, there has been little academic study of online ad services, and online sites have released very little data about their ad markets. Thus, our goal is to bring visibility to OSN ad markets, focusing on Facebook (currently the largest OSN ad market). We aim to examine how Facebook monetizes user data for its ad services: How are the users with different demographics valued? Moreover, how stable are these values in the ad market over time?

To implement the study, we develop techniques that allow researchers to measure and understand OSN ad markets. We explore the suggested bid—a common feature of ad services that suggests prices to bid for a given target demographic. We demonstrate that the (undocumented) feature that suggests bids to advertisers is most likely calculated via sampling recent winning bids. Through the exploration of suggested bid data for different demographics, we find dramatic differences in prices paid across different user interests and locations. Finally, we show that the ad market shows long-term variability, suggesting that OSN ad services have yet to mature. Overall, we present the first mechanism to quantify the relative value of different user demographics towards the OSNs. As more online services develop ad markets, our approach can be used to measure these markets as well.

The third and final contribution of this thesis is that we examine how the role and behavior of third parties has changed with online services and the rise of mobile applications. As discussed, some third parties in the network may get access to user-uploaded data as it transits across their networks. Thus, we examine what kinds of user data third parties in the network have access to. In previous studies, to examine how different information flows to third parties, researchers have created fake accounts and uploaded fake information on OSNs [74, 75, 97] and on mobile services [76, 77] and traced all requests the browser made. Some others have developed client-side techniques such as browser extensions or add-ons, such as Adblock Plus [8], NoScript [115], which rely on standard blacklist/whitelist mechanisms [9, 24] of privacy protection. In contrast with these approaches that rely on “rooting” users’ devices or instrumenting applications or browsers, we instead aim at a solution requiring access only to the network itself, with the benefits of lower barriers to deployment and higher coverage of users. However, because we do not assume any privileged access to devices, we are unable to gain visibility into HTTPS traffic, which means we focus on the plaintext traffic.

We aim to discover the extent to which these services are automatically collecting user-uploaded data, focusing on personal information (PI), which is “information about user’s demographics or other identifiable information, including personally identifiable information (PII), but not necessarily lead to distinguish or trace an individual’s identify.”
CHAPTER 1. INTRODUCTION

We examine this situation by first underscoring the difficulty of the problem of locating user PI in network traffic. We demonstrate that only a very small fraction of protocol fields convey user PI, making our endeavor akin to “finding a needle in the haystack”. Then, we develop a novel technique that automatically detects user PI traveling through the network as it is collected by services accessed via browsers or mobile applications. Specifically, we propose a novel method based on the semantics of the values to locate user PI of different types, including, but not limited to, users’ names, genders, email addresses, ages, geo-locations, cities, postal codes, and phone numbers. The method includes techniques to filter out potential “containers” that do not actually carry user PI and extend the set of containers initially found with additional ones. We evaluate the false positive/negative rates of our proposed method on real ISP traffic traces and show examples of interesting findings, including which kinds of web sites or applications are more likely to transmit PI and which types of user PI are most commonly collected. Our methodology can be applied to examine network flows to other third parties in online systems as well.

Taken together, our research makes three significant contributions to the study of user data sharing activities and the privacy implications for such activities in online services.

1.2 Outline

The outline of this thesis is organized as follows. Chapter 2 presents a brief introduction of the background information and Chapter 3 discusses related studies. To begin our analysis, in Chapter 4, we conduct the first large-scale measurement study to quantify the privacy problem that users experience on Facebook, and present the new application Friendlist Manager to help improve privacy tools. In Chapter 5, we examine how online service providers leverage the collected user data for their ad services. In Chapter 6, we explore what kinds of user PI online sites and network operators can get access to. Finally, Chapter 7 concludes with a summary of our contributions, the implications of our work, and future research directions.
Chapter 2

Background

In this chapter, we first give an overview of online social networks (OSNs), and introduce general background information about the Facebook [41] social network, as we focus on this web site in our research. We describe the web site and its functionality as it existed when our study was deployed, pointing out instances where functionality has changed over time. Then we present how online advertising works in the traditional advertising networks, as well as the largest OSN advertising network as-of 2016, which is Facebook. Finally, we discuss user privacy issues, including the definition of user personal information (PI), helping us better understand and identify them from network traffic; followed by the privacy leak problem in online services.

2.1 Online social networks (OSNs)

In the early 2000s, as more users became connected to the Internet, OSNs began to grow in great popularity. In general, OSN sites can be classified along two axes. First, different networks allow users to share different types of content, such as photos, videos, messages, demographics, links, and notes. Second, the OSN sites today can be commonly accessed from different places, from not only desktop browsers, but also mobile browsers, and mobile applications.

Since new OSN services allow users to upload new content, the first classification method is based on the different kinds of content they serve. Some examples of these systems are (1) multimedia sharing sites, such as photos on Flickr and Picasa, and videos on YouTube, (2) professional sites, such as profiles of business and professionals on LinkedIn, and teachers on Classroom 2.0, (3) and general sites for building social connections, such as Facebook, Google+, and Twitter. Though Twitter is
CHAPTER 2. BACKGROUND

more correctly called a microblogging site, other sites such as Facebook have more general types of content, including profiles, blogs, photos, videos, music, links, notes, and applications.

In the second classification approach, we look at how OSN services are designed to be accessed. One group of traditional OSNs such as Facebook and Twitter can be accessed via web sites on desktop computers, and due to the prevalence of mobile devices, they have evolved to allow access from mobile devices. Thus, users can access them through mobile browsers and mobile applications as well. A significant number of user visits occur on mobile devices—91% for Facebook [45], and 83% for Twitter in 2016 [144]. As of April 2016, over half (54%) of Facebook users log in only from a mobile device, and in total, Facebook has acquired 1.51 billion mobile active users [48]. As this research will show, the study of the Twitter ecosystem demonstrates the increasing trend of Twitter users’ activities on mobile devices. In fact, by the end of 2013, over half of all public tweets were created on mobile platforms [90].

Another new group of OSN services has been designed and developed to be primarily used via dedicated applications on mobile devices (tablets and smartphones). Some examples of these mobile applications are Foursquare, a location-based application, Instagram and Pinterest for sharing photos or videos, WhatsApp for messaging, Snapchat for sending time-expiring photos and messages, and WeChat for text and voice messaging. These mobile applications are all available to download in the Apple App Store [12], Google Play Store [65], or Windows Phone Store [155].

In this thesis, we focus on Facebook in particular. Facebook is the largest and most mature OSN site, and as a result, it has been the subject of the greatest amount of debate over the privacy implications of online services. Below, we will discuss the Facebook social network in detail.

The Facebook social network  Facebook [41] is an online social networking service that allows users to upload and share a variety of content types, including text messages, images, videos, news articles, etc. After users register, they can set up their personal profiles that include basic information such as name, birthday, marital status, work and education history, interests, contact information, family and relationships, and life events. Users can friend other users, with a maximum limit of 5,000 total friends. Users can upload different types of content (photos, videos, statuses, links, notes, life events), which are shown on their Timeline page. Users can view their friends’ posts on the News Feed page, and they can like, comment on the posts or share them with others. Users can also send messages to their friends, join groups, and use applications.

Facebook was launched in February 2004, and has evolved into the largest OSN site in the world. The number of users has increased to hundreds of millions in 2008 to billions in 2012. Facebook
CHAPTER 2. BACKGROUND

has over 1.65 billion monthly active users as-of 2016 [117]. On average, Facebook has 300 million uploaded photos per day, and it receives 8 billion daily video views as-of 2016.

Facebook’s web site and features change frequently. When we did a measurement study on the Facebook privacy problem [89] in 2011, Facebook allowed users to manage the privacy settings of uploaded content (photos, videos, statuses, links, and notes) using five different granularities: Only Me, Specific People, Friends Only, Friends of Friends, and Everyone. Specific People allows users to share content by explicitly choosing friends or pre-created Friend Lists [50]. Users create a friend list, add a subset of their friends to it, name it, and can then select the list as a basis for privacy control. Friend lists are private to the user who creates them unless the user explicitly chooses to display them as part of his profile.

The default or recommended privacy setting for all content was Everyone, meaning users share their content with all the Facebook users [1] if they decline to modify their privacy settings. Facebook has since simplified their privacy setting options, presenting only Friends and Everyone settings by default. The other options are still available via Custom settings. Since May 2014, for newly registered users, the default setting for sharing content has switched to Friends, instead of Public [107].

Facebook provides different software development kits (SDKs) which have a library of powerful features that enable developers to integrate with Facebook APIs [42], including Atlas API, Graph API, Marketing API, for building applications, accessing pages, users, posts, groups, events or managing ad campaigns. Please note that we will use some of these APIs in our research study in Chapter 4.

2.2 Online advertising

Online advertising has become the economic underpinning of much of the Web. There are two main different types of online advertising: advertising networks and OSN advertising. In our research study in Chapter 5, we focus on the OSN advertising, which is fundamentally different to the traditional advertising networks. We will introduce each of them in more details below.

Advertising networks Large advertising networks (e.g., Google, Yahoo!, Bing’s ad network) serve ads for millions of web sites, and as a result, many sites provide free services supported by advertising.

However, Google’s ad network, by default, knows relatively little about the user. Instead, the network must track users using cookies and other techniques, extracting more information about
CHAPTER 2. BACKGROUND

<table>
<thead>
<tr>
<th>Basic Fields</th>
<th>Parameters/Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Location</td>
<td>Country, State, City, Postal code</td>
</tr>
<tr>
<td>Gender</td>
<td>Male, Female, All</td>
</tr>
<tr>
<td>Age</td>
<td>Range (from 13–65)</td>
</tr>
<tr>
<td>Precise Interest</td>
<td>Travel, Science, Music, ...</td>
</tr>
<tr>
<td>Broad Category</td>
<td>Cooking, Gardening, iPhone 5, ...</td>
</tr>
<tr>
<td>Interested In</td>
<td>Male, Female, All</td>
</tr>
<tr>
<td>Relationship Status</td>
<td>All, Single, In a relationship, Married, Engaged, Not specified</td>
</tr>
<tr>
<td>Language</td>
<td>English, Spanish, French, ...</td>
</tr>
<tr>
<td>Education</td>
<td>Anyone, In high school, In College, College Grad</td>
</tr>
<tr>
<td>Workplaces</td>
<td>Google, Facebook, AT&amp;T, ...</td>
</tr>
</tbody>
</table>

Table 2.1: Facebook’s targeting parameters made available to advertisers.

users through data mining. For example, Google provides a significant number of services (e.g., email, calendar, etc) to users, which enables them to gather additional information about a user’s interests and demographic features.

Many of these ad services are implemented as *auctions*, with individual advertisers bidding on specific keywords, pages, or search terms. The ad network selects the winning bidders and presents their ads to the users. Many traditional ad networks, such as Google and Yahoo!, use Generalized Second Price (GSP) auctions [34].

Advertisers typically bid using either CPM (Cost Per Mille, the cost of 1,000 ad impressions) or CPC (Cost Per Click). To support both bidding mechanisms at once, ad networks will typically record each advertiser’s click-through rate (CTR, the fraction of impressions that result in a click). And they provide a way to compute an estimated CPM bid given a CPC bid (i.e., the estimated CPM bid is simply the CPC bid multiplied by the advertiser’s CTR).

These auctions are extremely popular with advertisers. Google—far and away the largest ad network on the web—alone earned over $67 billion in ad revenue in 2015 [62]. Yahoo!’s worldwide net digital ad revenue achieved up to $3.28 billion in 2015.

**OSN advertising** With the rise of OSN services, a new type of ad network has emerged: closed-site ad services run by OSNs such as Facebook. Unlike prior systems, where the ad network was forced to infer user information from cookies, browsing history, and search terms, OSNs are provided demographic information directly by the users themselves. As a result, advertisers are able to target users directly (via profile attributes), rather than targeting keywords or search terms.

In different online advertising platforms, the underlying auction mechanism may vary: Facebook
CHAPTER 2. BACKGROUND

uses Vickrey–Clarke–Groves (VCG) auctions [149]. The differences between GSP (used by Google and Yahoo!) and VCG mechanisms primarily lie in how they calculate the price that a winning bidder should pay.

Advertisers can place ads on Facebook by creating campaigns; each campaign consists of a specific ad, a target demographic, a CPC or CPM bid, and a budget. After an advertiser creates a campaign, it must be approved by Facebook (this process typically takes less than one day). Once the ad campaign is active, the advertiser participates in the ad auctions that occur whenever a user in the target demographic is shown an ad. An advertiser can view their campaign’s status on Facebook’s web site to see details on the number of impressions, clicks, and unique users, as well as the overall cost. They can pause or cancel their campaign at any time.

Unlike traditional ad networks, OSNs such as Facebook have significant data about each user, including their personal information (demographics, interests, educational history, relationship status, etc.), identities of friends, and their activity on the OSN. Advertisers can target any combination of these parameters, and are only required to specify at least one country. An overview of the currently available targeting parameters [47] is shown in Table 2.1. Facebook provides different granularities of location information in different countries: the US includes ZIP code, most Western European countries include city/town, and most developing countries only allow targeting at the country level.

The OSN-based ad services carry a significant number of ads. Based on the the only ground truth data released—10-K annual reports—Facebook had $17.08 billion in ad revenue in 2015, which makes up to 95% of the total annual revenue. Facebook’s revenue in 2015 increased $5.46 billion, or 44% compared to 2014, and this is primarily due to an increase in ad revenue [46]. In 2015, Twitter had a total revenue of over $2.22 billion, among which 89% ($1.99 billion) comes from the ad revenue, and showed an increase of 59% compared to 2014 [145].

2.3 User privacy

With the growth in popularity of online services, and a corresponding growth in the ad networks that power these services, a number of privacy concerns have been raised about these services. In this section, we first introduce the definitions of user personal information (PI) and personally identifiable information (PII), followed by discussion on the privacy leak issues on the OSN sites, as well as potential leaks on mobile devices.
CHAPTER 2. BACKGROUND

**Personal information (PI)**  One common definition used in most previous studies is *Personally Identifiable Information* (PII): “information which can be used to distinguish or trace an individual’s identity either alone or when combined with other information that is linkable to a specific individual” [76].

However, one goal of this thesis is to quantify how much user data is being collected by online services and third parties. To captures different kinds of information that online services can collect on users, we define a new non-standard notion of *Personal Information* (PI), which is “information about user’s demographics or other identifiable information, including PII, but not necessarily lead to distinguish or trace an individual’s identify”.

There are many different kinds of PI, like personal identifiers (name, date of birth, social security number), users’ current location, device identifiers (IP address, unique cell phone ID), contact information (email and physical addresses, telephone numbers), social links (friends, interests), uploaded content (photos, videos), and online activities (search history, web site visits, games), among others.

User PII can be considered as a subset of all user PI, or a combination of some user PI. Since we do not know if the data that has been and is being gathered can be translated to PII, instead, we focus on a much broader range of user PI in this thesis.

**Privacy leak**  Given the increasing important role of online services, enhancing users’ privacy protection is a critical issue. To achieve the end goal of protecting user data, in this thesis, we take a first step towards identifying how much user data is being collected by online services and third parties. The next step is how to identify and prevent the potential privacy leaks. Now, we turn our attention to different kinds of privacy leak problems.

As discussed above, some third-party servers, such as ISPs, are widely used to transit user data across their networks. User data can also be written and hosted by third-party entities other than service providers, and for this reason, privacy can be undermined by third parties. Another kind of privacy leak occurs due to application bugs and vulnerabilities that expose the user’s personal data to attackers, or other applications [128, 153]. Beyond the benign applications, a malicious privacy leak may happen when online services expose user data to third parties without the user’s permission or control [67, 77, 97, 106].

The causes for concerns about potential leak of user data are based on the growth and aggregation of information tracking resulting from users’ activities on online services. When it comes to third parties, such as aggregators and applications, users cannot understand the potential flows of user data
between third parties and OSN sites, which means they are unaware of the breadth of the privacy leak problem, including actual and potential leaks. Current privacy protection measures block privacy leaks via stricter privacy settings [17, 131], customized access control [38, 100], as well as client-side tools, such as browser extensions or add-ons (e.g., Adblock Plus [8, 10], and NoTrace [116]). These client-side tools rely on blacklist/whitelist mechanisms of privacy protection.

**Mobile privacy**  In the context of mobile environments, there can be new privacy leak vectors, because applications have access to potentially much more information than web-based services, including some information related to the mobile devices, such as their unique device identifiers, real-time location, contacts, call logs, and calendar entries [77, 110].

Typically, mobile devices have one or more unique device identifiers (iOS *IFA ID* for iPhone, *Android ID* and *Android Advertiser ID* for Android phone, *DeviceUniqueId* for Windows Phone, and ICCID, IMEI, IMSI, MAC address). Please note that mobile applications on iPhone no longer have access to *UDID* starting since May 2013 for privacy reasons [13]. If these unique identifiers are leaked to a third party via an app, it could be used to track the actions of a user using a device across different applications. Recent studies [77, 128] have shown that the most commonly leaked information from all three OSes: iOS, Android, and Windows Phone devices is device identifiers, likely used by advertising and analytic services.

Another significant concern is about the “checking in” feature at a location (GPS latitude and longitude, zip code), which reveals much about the user, their presence, their location, and the current timestamp, and has been found to be leaked both from iOS and Android applications [67, 77, 128]. Another group of commonly leaked PI are user identifiers (e.g., name, gender, date of birth, and e-mail address).

Even worse, the sensitive user login credentials such as usernames and passwords are sent in plaintext flows on multiple mobile applications [128]. Documented bugs [146] have leaked plain text chat logs and passwords for hundreds of millions of users. As a consequence, the combination of unique identifiers of devices, location information, and privacy leak of other user PI all conspire against protection of a user’s privacy.
Chapter 3

Related Work

In this chapter, we describe prior studies related to the topics presented in this thesis. As we cover a number of different topics, the related work has been grouped into several sections detailing (1) work that tries to understand online advertising networks, (2) work that examines user privacy issues on OSN sites, privacy enhancing techniques, as well as privacy leak problem.

3.1 Online advertising

We now turn our attention to the studies on online advertising networks and evaluating the value of users to OSN sites, and surprisingly, there is little research work on OSN ad auctions.

Auction mechanisms Researchers have studied existing Web-search-based advertising networks (e.g., Google’s ad market) [52, 138, 159], prediction markets run by Google, Ford and others [28], and used “estimated prices” [54] from Google’s Traffic Estimator Tool [64] (similar to Facebook’s “suggested bids”) as a mechanism for understanding the network. The “estimated prices” provided potential advertisers with a guide to the auction prices that they would expect to pay for different keywords in different locations. They collected projections for 139 keywords in 195 geographic areas defined by Google to closely resemble metropolitan statistical area, under a key assumption that on average, the relative price estimates reflect the relative values of the keywords in the market.

Noti et al. [111] demonstrated that bidders with some explicit knowledge during an initial learning phase can bid with better valuations than those without such knowledge; this implies that suggested bids can provide useful guidance for bidders. A related study [122] has developed a new analytical model of GSP auctions in order to predict the number of clicks and the total price the advertiser can
CHAPTER 3. RELATED WORK

expect, using the advertiser’s bid and the distribution of the number of opponents and their relative weighted bids.

Researchers have also examined and improved auction mechanisms, including better CTR estimates [29, 53], usage of reserve prices in ad auctions [121, 141], and optimization in multiplicative bidding [18]. Using a sample from a week’s worth of data across all keywords on Bing, other work [21] showed how to optimize linear combinations of the stakeholder utilities, showing that these can be tackled through a GSP auction with a per-click reserve price. There has also been much work proposing new models for conducting online auctions [59, 60, 103, 159].

Lately, Xia et al. [156] performed a study on the bid suggestion from LinkedIn, which is similar to our study on the suggested bid feature on Facebook [91]. They examined how much is the attention of a LinkedIn user worth, using the collected suggested bid data. They found that the (suggested) bids are generally stable over time; they negatively correlate with GDP and income; the bids for users from different industries vary a lot. Their conclusions are somehow different from ours, probably due to the differences among two OSN sites.

Our study [91] have done more than just evaluating user value, we also provided another methodology for the advertiser to estimate the price they need to pay when targeting users with different demographics in OSNs, when the information of opponents are not available. Furthermore, we provide a way for advertisers to estimate and compare the prices for targeting different demographics of users in an automatic way; previously, advertisers needed to manually select a combination of interests, age, gender, and other options to obtain Facebook’s suggestions.

Estimating the value of users in OSNs  Prior work [56] has shown that the contribution of users to advertising revenue is skewed (20% of users accounting for 80% of revenue), which provides support for our observations that the OSN ad market prices vary widely across different user interests and locations. Another study [84] has utilized AdReveal, a browser based tool to provide measurements of 139K online display ads and analysis of 103K Web pages. They demonstrated that up to 65% of ad categories received by users are behaviorally targeted using users’ online interests. Our study has shown similar result: targeting users with specified interests is more expensive, implying more competition among advertisers.

Much of this work is orthogonal to ours, as we present a mechanism for measuring the OSN ad network itself. Others explored ways of identifying influential users in OSNs [5, 15]; our work is complementary to these. There are nascent systems for leveraging unique features of OSNs for advertising (e.g., adby.me [78] allows users to create their own ads, shown to their friends).
Finally, a few companies (e.g., AdParlor) used Facebook’s suggested bid data to provide clients with information on the ad market. Unfortunately, there are few published details of how their analysis is conducted.

Additionally, Saeztrumper et al. [136] proposed a new framework to measure the value of users on Facebook by considering both direct impressions caused by the users’ actions and indirect impressions caused by actions induced on friends and other users. They showed that the amount of user’ actions is related with the monetary value that they produce. Another recent study [22] shows that a small percentage of active users represent 50% of the user generated content (UGC) on two different OSNs: Facebook and Twitter.

3.2 User privacy

Privacy is an emerging challenge in OSNs, and a number of researchers have examined different aspects of the privacy problem. In this section, we review a series of related work about user privacy on Facebook and other OSN sites, privacy enhancing tools, as well as detection of privacy leaks.

3.2.1 Privacy issues on OSNs

Now we first start by detailing the academic literature on user privacy on Facebook as well as other sites.

Facebook privacy issues  A series of studies [32, 58, 125, 161] surveyed users’ awareness, attitudes, and privacy concerns towards profile and content visibility on Facebook. Some studies [33, 161] showed that Facebook users reveal a lot of private information to people they do not necessarily know, and they are not very well aware of privacy options or who can actually view their contents. Other work [2, 57] has shown that only a minority of users change the default privacy preferences on Facebook, which is due to that most users were not aware of the privacy settings available. Debatin et al. [32] illustrated that young adults need to be educated about risks to their privacy in a way that they actually alter their behavior.

In addition, most users aren’t aware of the risks of not protecting their personal information they share. It is obvious that Facebook’ privacy policy and the terms of use are largely not known or understood by Facebook users. Stutzman et al. [132] studied how user privacy and disclosure behavior changed over seven years (2005 to 2011) from more than 5,000 Facebook users. They
CHAPTER 3. RELATED WORK

found that Facebook’s privacy settings changed significantly over the last decade. The ever-changing privacy settings can be one significant reason why users largely do not know or understand them.

There are several closely related papers which have measured the privacy considerations of different kinds of information, such as News Feed [68], photos with faces [71, 86], tagged photos [14], and basic profile information [101]. All of these papers demonstrate the importance of the ease of information access in alleviating users’ privacy concerns. Ilia et al. [71] proposed a system which determines which faces the user does not have the permission to view, in order to improve the current photo-level access control and protect the restricted photos on Facebook.

Madejski et al. [101] show that privacy settings for uploaded content are often incorrect, failing to match users’ expectations. Johnson and Madejski et al. [73, 102] extended their study to examine more aspects of privacy issues, including privacy concerns, users’ friend network compositions, the sensitivity of posted content, and their privacy-preserving strategies. They observed that 16.5% of participants had at least one post that they were uncomfortable sharing with a specific friend. They concluded that the current privacy controls allow users to effectively manage the outsider threat, but they are unsuitable for mitigating concerns over the insider threat.

There are two primary distinctions between their work and ours. First, they rely on text analysis to select content that is potentially privacy-sensitive; doing so locates additional privacy violations but prevents an overall estimate of the fraction of content that has incorrect settings. Second, we directly compare the user survey results to the in-use settings, instead of relying on inferring the existing privacy setting through fake accounts.

Privacy issues on other sites Similarly, there are privacy issues on Twitter [96, 99], which comes from sensitive tweets posted by users themselves and retweets of other users’ private tweets. Mao et al. [99] proposed a system to alert users to avoid posting a sensitive tweet. Meeder et al. [96] found that if a private tweet is retweeted, potentially damaging information is public or viewable to a wider audience than intended. Although Twitter and some Twitter agents do either prevent a user from retweeting something private or warn a user before, it is still possible to unofficially retweet (by copying and pasting the tweet, and signifying that the tweet come from a specific user).

Researchers have investigated the privacy model of all existing OSNs, demonstrating that sites often leak numerous types of privacy information [4, 76, 88]. A number of papers have reported that users have trouble with existing extensive privacy controls, and are not utilizing them to customize their accessibility [32, 85, 94, 134]. Tufekci et al. [140] found that there was no relationship between
CHAPTER 3. RELATED WORK

users’ privacy concerns and their level of disclosure on OSN sites. However, this study focused on only the profile attributes, rather than concerns over shared content, such as photos, and statuses.

Several studies [95, 157, 163] focused on the inferring users’ personal information by correlating the public data from one or multiple OSN sites. Zheleva et al. [163] applied link-based and group-based classification to study privacy implications in social networks, to infer the information of private-profile users. Xia et al. [157] proposed Tessellation, a framework to correlate user identity to its online behavior, using various OSN identifiers extracted from the mobile traffic from OSN sites. They found that up to 50% of the traffic from a cellular service provider (CSP) can be attributed to the names of users.

Crowdsourcing Since we need to collect data from real OSN users to examine the current state of privacy issue, now we describe how we recruit users. Most prior work uses small-scale surveys of locally recruited users to study user attitudes towards privacy. This approach affords more control over the surveyed population, but also limits the scalability of the survey. In our work, we take a different approach, recruiting users from Amazon Mechanical Turk (AMT), which offers the potential of greater scalability and a lower cost of running experiments [104].

There have been multiple studies showing that the behavior of participants on AMT is comparable to the behavior of laboratory subjects in more traditional economic and psychological experiments [69, 123]. Considering that compensation may affect the quality of survey results, Mason and Watts [105] show that in online peer production systems like AMT, increased financial incentives increase the quantity, but not the quality of work performed by participants. These studies provide evidence that AMT offers a potentially attractive way of quickly recruiting significant numbers of survey users.

3.2.2 Privacy enhancing techniques

In this subsection, we look at related work on developing techniques to enhance privacy for end users.

Users on OSN sites have access to a variety of privacy controls. Common controls include limitation of profile access, content-level access control, as well as customized access such as blocking and hiding other site users.

There is also significant work that explores new approaches that can enhance the content sharing privacy on OSNs [17, 38, 40, 100, 131]. Banks et al. [17] utilized interactions between friends as the currency for data access. Fang et al. [38] built a privacy wizard which asks users to create privacy
CHAPTER 3. RELATED WORK

labels to a subset of friends, and constructs a classifier to infer privacy preferences for unlabeled friends. To reduce disclosure-related harms, Stutzman et al. [131] proposed a friends-only private Facebook profile, which limited contribution to and resources drawn from the encompassing network.

Researchers are proposing new privacy-preserving alternatives for OSN sites [19, 112, 152, 164]. Wilson et al. [152] implemented Polaris which keeps user data on user-managed devices, to protect certain information from the service providers. Nilizadeh et al. [112] proposed Cachet which is a decentralized architecture for privacy preserving OSN sites. They used a distributed pool of nodes to store user data and ensure availability, and leveraged cryptographic techniques to protect the confidentiality of data.

Community detection Community detection algorithms are widely applied in the privacy enhancing tools discussed above. Through grouping users’ friends into meaningful groups of users, they can highly reduce the entities that users need to handle in terms of privacy control complexity. For example, Fang et al. [38] found that using communities as features can significantly improve the prediction accuracy even with less user input.

Community detection in large networks is a well-studied problem with a rich literature and many proposed approaches [16, 23, 25, 39, 113, 114, 127]. At a high level, all these algorithms can be classified into two main categories: global and local. Global approaches [23, 39, 113, 114, 127] assume full knowledge of the network, and usually produce a complete partition of the nodes into communities. Local approaches [16, 25] only assume a limited knowledge of the network, and typically produce a few communities before halting.

Our study on Facebook users’ friend lists [89] has shown that the friend lists have a correspondence with the structure of the social network. In particular, many of the user-created friend lists correspond to communities in the user’s 1-hop social network, and hold the potential to be detected by existing community detection algorithms. In this thesis, we proposed Friendlist Manager app which applied two different community detection algorithms [23, 95] to generate meaningful friend lists for Facebook users, more details can be found in Section 4.4.

Client-side tools Researchers developed client-side techniques such as browser extensions or add-ons, including Adblock Plus [8], RequestPolicy [129], NoScript [115], and NoTrace [116]. They have been widely used by many users, and also provide a way for researchers to measure the PI transmitted in the traffic. All the analyzed tools rely on standard blacklist/whitelist mechanisms of
privacy protection. For example, AdBlock Plus is structured as a blacklist, blocking requests to a pre-defined list of ads and malware domains.

Some recent new tools: PrivacyGuard [135] and AntMonitor [81] utilized the Android VPNService to intercept traffic and perform traffic analysis from Android devices. Compared to our approach, these client-side tools offer users greater control over the process of blocking PI leaks, but face additional challenges obtaining large-scale deployment.

3.2.3 Detection of privacy leaks

We now discuss the related studies on measuring the personal information and privacy leakage.

Static data flow analysis  Static analysis aims to measure privacy leaks via the network between user applications and different web sites or mobile services. PiOS [35] uses program slicing to detect privacy leaks in iOS applications. Researchers can use the static analysis to explore broad execution paths including infeasible ones. Similarly, Unsafe [61] focused on the advertising libraries the mobile applications contain. FlowDroid [6] performed a highly precise taint flow static analysis within single components of Android applications, and Epicc [120] analyzed properties of messages sent between components of applications.

While static analysis has the potential to detect leaks before they occur and reduce the run-time overhead, it often requires access to the app source code and can only feasibly be run on a small set of applications or web sites. In contrast, our approach only requires access to the network traffic, which can often be accomplished via a router tap.

Dynamic data flow analysis  Different from the static analysis, dynamic analysis runs in real-time as a user executes applications. The major advantage of dynamic analysis is that, as users can provide relevance feedbacks to its false alerts, it can reduce false positives at runtime. TaintDroid [36] used lightweight dynamic taint analysis built into modified Android middleware; the system alerts the user to the presence and nature of the leaks in the whole applications. Similarly, Vision [55] directly instrumented the smartphone platform and tracked information flow at runtime. Leveraging unique Android execution model to reduce the search space, AppIntent [160] automatically presented a human analyst the UI manipulations that leads to the sensitive data transmission.

While these dynamic analyses can precisely pinpoint leaks from the devices they are installed, the intrinsic cost of their deployment (e.g., having to install browser add-ons or custom Android builds) makes them difficult to deploy to a large userbase. Instead, integrating these data flow monitoring
CHAPTER 3. RELATED WORK

techniques to our approach, which unobtrusively considers the network traffic at large, could lead to a more comprehensive solution to PI leak detection (i.e., using static/dynamic analysis to identify additional potential keys of interest).

Network flow analysis A number of projects [74, 76, 77, 97] examined privacy leaks by web sites, typically focusing on network traffic from OSNs directly to external entities. By creating fake accounts on OSNs and on mobile services and tracing all requests the browser makes, researchers were able to further examine how different information flows to third parties. In brief, these papers find that all OSNs exhibit some leak of private information to third parties, typically via Request URIs and HTTP headers (e.g., Host, User-Agent, Referrer, and Cookie). These approaches are largely complimentary to the work presented in this thesis, as we aim to develop a system that is able to detect the transmission of (unknown) user data in a live network.

Chaabane et al. [27] explored indirect privacy leak to external entities via third-party applications. Chaabane et al. [27] has looked into the third-party applications on Facebook and RenRen. They found that 22% of Facebook applications and 69% of RenRen applications provide users’ personal information to one or more fourth-party tracking entities, such as trackers and advertisers.

Ren et al. [128] developed Recon, a system designed to capture mobile device traffic using a VPN and the middlebox Meddle [126]. They used machine learning to identify PII in network traffic, enabling users to detect and manage potential privacy leaks, regardless of the device OS or network being used. Their system shares many of the same goals as ours, and the approach is largely complementary (the domain-keys that we identify as carrying PI can be leveraged by Recon as ones to flag, and their approach may offer visibility into HTTPS traffic if the user so chooses).

Another framework privacy capsules [67] aimed to prevent privacy leaks by executing mobile applications in two sequential phases: first the unsealed state, when applications have no access to user PI information and full access to untrusted resources, and second the sealed state, when applications have access to user PI but cannot communicate with untrusted devices. They implemented a prototype implementation in Android, which has low cost and is effective for some applications.

Mobile privacy While prior studies focused on OSN services and third-party applications, other studies examined the privacy leak in mobile network data. Shklovski et al. [133] identified a similar information asymmetry in the context of mobile privacy: users are unaware of the data collection performed by mobile applications.
CHAPTER 3. RELATED WORK

Researchers [7, 82, 83, 87] proposed fine-grained privacy recommendations for users to better manage the mobile app privacy settings. Agarwal et al. [7] provided ProtectMyPrivacy (PMP), which is a system that enables iOS devices to detect access to private data. It allowed users to grant or deny app access to user information, or provide fake information to an app. PMP also provides privacy setting recommendations for new applications to users via crowdsourcing.

Hornyack et al. [66] proposed AppFence, which could substitute fake information into API calls made by applications, such that applications could still function but with protection of users’ private information. Two studies [83, 87] applied profile-based method which can learn users’ general preferences and suggest the default settings for users.

Balebako et al. [20] proposed mobile privacy nudges based on applications’ access frequency to specific data and evaluated them in a lab study. Almuhimedi et al. [3] extended Balebako’s work by measuring the effect of privacy nudges in triggering users to review and adjust their app permissions. They focused on location access, phone contacts, calendar, and call logs, and showed that Android users can indeed benefit from an app permission manager such as App Ops, proposed in this study.
Chapter 4

Analyzing and Improving Facebook Privacy Settings

In this chapter, we begin to examine how the OSN systems are affecting the privacy of end users. The privacy we examine in this work is that how users understand and manage the privacy settings for their uploaded content to OSN sites. We first design and present a large-scale measurement study to quantify the privacy issues on Facebook. This user study aims to compare the ideal and actual settings for shared content, and quantify the extent to the privacy violations. Then, we show that user-created lists of friends often have a strong correspondence to the structure of the social network, implying that lists of friends may be automatically inferred by leveraging the social network structure. Using this insight, we deploy a Facebook application Friendlist Manager that leverages the users’ social network to propose meaningful lists, as a basis for privacy controls. We first presented these studies as a conference paper in IMC’11 [89] and a demo paper in WWW’12 [92].

4.1 Motivation

The new model of OSN services results in a fundamental shift in the patterns of context exchange over the Web. Users are allowed to upload and share their own content on the OSN sites, unfortunately, the fundamental changes in data sharing have caused significant debates in user privacy issues. Our overarching goal is to improve the set of privacy controls and defaults, but we are limited by the fact that there has been no in-depth study of users’ privacy settings on sites like Facebook.

Moreover, to reduce the user burden when managing their privacy, sites like Facebook allows the user to group friends into specific lists, known as the friend list mechanism [50], aggregating
CHAPTER 4. ANALYZING AND IMPROVING FACEBOOK PRIVACY SETTINGS

the interactions and content from all friends in each friend list. While this approach greatly reduces the burden on the user, it still forces the user to create and populate the friend lists themselves and, worse, makes the user responsible for maintaining the membership of their friend lists over time.

In this chapter, we take the first steps in quantifying the privacy issues from the perspective of end users. We address the problems by analyzing the current state of affairs on Facebook. In particular, we center our analysis around these questions:

- What is the distribution of user-selected privacy settings? What fraction of content items are shared with the default setting?
- What are the ideal privacy settings desired by users?
- How close are the ideal settings to the actual settings that users have?
- Is there potential to aid users in selecting the correct privacy settings for their content? Can we reduce the mental burden on users by automatically grouping others into meaningful groups for expressing privacy settings?

Since we wish to examine whether users’ desired privacy settings differ from their existing settings, we need to ask users detailed questions, i.e., survey them. Thus, we design a survey (implemented as a Facebook application) that examines users’ current privacy settings and queries users about their desired settings. In order to scale to a significant number of Facebook users, we recruited users to participate in the study using Amazon Mechanical Turk (AMT). We automatically crawled the existing privacy settings of each piece of uploaded content for 200 users, resulting in 116,553 observations of existing privacy settings. Additionally, each of the users answered survey questions about their desired privacy settings for up to 10 of their photos, resulting in a total of 1,675 measurements of desired settings.

Second, we take the first steps towards answering this situation by automating the creation and maintenance of Facebook friend lists. In the user study, we show that the friend lists have a correspondence with the structure of the social network. In particular, many of the user-created friend lists correspond to communities in the user’s 1-hop social network, and hold the potential to be detected by existing community detection algorithms (e.g., [23]). We present the Facebook

1 Subsequently, Facebook introduced smart lists [51], which are essentially automatically created friend lists based on profile attributes (such as high school attended). Unfortunately, many users do not provide the detailed attribute information necessary, so the smart lists created by Facebook typically only contain a subset of the “correct” users.

2 This study was conducted under Northeastern University Institutional Review Board protocol #10-10-04.
application Friendlist Manager, which analyzes a user’s 1-hop social network, detects communities, proposes the communities to the user as friend lists, allows the user to tweak the friend lists, and finally creates the friend lists for the user. Additionally, Friendlist Manager examines any previously existing friend lists to locate friends who may be tightly connected to friends in the friend list but not a member of the friendlist; if such friends are found, they are also presented to the user for consideration.

The remainder of this chapter is organized as follows: Section 4.2 describes our data collection methodology and data set statistics. Section 4.3 analyzes our survey data, focusing the relationship between the actual and desired privacy settings. Section 4.4 proposes Friendlist Manager as a new tool to aid user privacy management and Section 4.5 summarizes.

4.2 Methodology

We now describe our approach for collecting data from Facebook users concerning privacy settings. We then detail a few statistics of the collected data set, and examine the demographics of the users who participated in our survey.

4.2.1 Approach

Our survey was hosted on a web server located at Northeastern University, and is available at http://socialnetworks.ccs.neu.edu/yabing. We designed our survey as a Facebook application. By doing so, the application is able to query Facebook to select content to query the user about, as well as to collect the current privacy settings for the user’s uploaded content. It is important to note that all data collected is immediately hashed and anonymized; no non-anonymized data is ever written to disk.

When the user begins the survey, he is shown a consent form detailing the purpose and methodology of the experiment and asked to provide optional demographic information (age, gender, income, education level, and U.S. state). Then, the user is asked to answer questions about the ideal privacy settings of some of his uploaded content. Finally, the survey collects information from the user’s profile, including the privacy settings for all uploaded content (photos, videos, statuses, links, and notes), any user-created friend lists, and the structure of the user’s one-hop social network (i.e., the friendship connections between the user’s friends).
CHAPTER 4. ANALYZING AND IMPROVING FACEBOOK PRIVACY SETTINGS

The survey selects 10 photos to query the user about. In order to ask the user about both benign and potentially privacy-sensitive photos, the survey first randomly selects up to 5 photos that have non-default privacy settings (i.e., photos where the user has previously modified the privacy settings). Then, the survey chooses the remaining photos randomly from among all photos uploaded, regardless of privacy settings. For each photo, the survey asks the user who, ideally, should be able to view and comment on the photo. The user is presented with a number of options, which approximate the privacy settings currently allowed by Facebook (abbreviations are used in the remainder of the paper):

- **Only Me (Me)** Indicating that the photo should be private the user.

- **Some Friends (SF)** The user is asked which of his friends should be able to access the photo. The user can select friends individually from a list, or can specify users using any friend lists they have created.

- **All Friends (AF)** Indicating that all of the user’s friends should be able to access the photo.

- **Friends of Friends (FoF)** Indicating that all of the user’s friends, and all of their friends, should be able to access the photo.
CHAPTER 4. ANALYZING AND IMPROVING FACEBOOK PRIVACY SETTINGS

- **Everyone (All)** Indicating that all Facebook users should be able to access the photo.

A screenshot of our survey is shown in Figure 4.1.

It is important to note that Facebook’s user interface and features change frequently over time. Facebook has since simplified their privacy setting options, presenting “Public” \(^3\) and “Friends” settings by default. Facebook has also added all the users’ friend lists as privacy setting options. The other options are still available via “Custom” settings. Also since May 2014, for newly registered users, the default setting for sharing content has switched to “Friends”, instead of “Public” \([107]\). As a result, due to the feature changes on Facebook, studying the privacy settings of Facebook users in 2011, would probably change today.

One of the benefits of building the survey as a Facebook application is the ability to quickly attract a large number of diverse users. To do so, we recruited users using AMT. We posted a Human Intelligence Task (HIT) describing the application, and offered users $1 to add our application and complete our survey. On average, we found that our survey took users 6 minutes and 30 seconds to complete, implying that completing our survey represented an average hourly wage of $9.23.\(^4\)

There are a few limitations of our methodology that are worth addressing. We focus on photos, as these represent the most commonly uploaded content on Facebook and they have the most diverse privacy settings. Additionally, we focus on content that is uploaded by the surveyed user; content uploaded by other users (even if it concerns the surveyed user) is not considered. However, Facebook applications are able to access all content uploaded by the user (i.e., the user cannot have a set of more privacy-sensitive photos that are hidden from applications). Finally, we treat each photo equally (in terms of the impact of an incorrect privacy setting), even though certain photos are likely to be more privacy-sensitive than others.

### 4.2.2 Data statistics

We now provide a brief overview of the data set that was collected. We deployed our HIT to AMT on May 2nd, 2011, and 200 users completed the survey. These users had an average of 248 friends, and had uploaded an average of 363 photos, 185 status updates, 66 links, 3 notes, and 2 videos. Only 45 users had uploaded fewer than 10 photos (of which 7 users had uploaded none). 81 out of our 200 users had also created at least one friend list, with a total of 233 observed friend lists. Thus, the average user who had created at least one friend list had 3 friend lists.

\(^3\)Indicating that anyone on or off Facebook, should be able to access the content.

\(^4\)We chose the compensation rate to be in line with recommendations from existing literature, which recommends paying close to the U.S. minimum wage of $7.25 per hour \([104]\).
CHAPTER 4. ANALYZING AND IMPROVING FACEBOOK PRIVACY SETTINGS

4.2.3 User demographics

One potential concern with recruiting users from AMT is the issue of bias; these users are unlikely to be a random sample of the Facebook population. We first note that the issue of bias is fundamental in user studies (e.g., psychology studies often use college students, another biased population), and our survey is no different. Nevertheless, we use the self-reported demographics in order to help to understand the nature and distribution of our user population.

In total, 195 (98%) users answered all demographic questions. We restricted our AMT user population to users only in the United States, and we had users from 40 of the 50 U.S. states. The most popular states were California (11%), Florida (11%), and New York (8.2%). We observed a slight male bias, with 54% of our users self-reporting as male; this differs from the overall U.S. Facebook breakdown of 42% male [30]. The self-reported age ranged between 18 and 60, with the median age being 24; this distribution is in-line with the overall U.S. Facebook population [30]. Finally, the self-reported yearly income level ranged from $0 to more than $120,000, with the median being $10,000–$20,000. These results demonstrate a wide variety of users, and are consistent with prior studies of AMT users [105].

One additional concern with our recruitment methodology is that our AMT users might be a “close-knit” group of friends, and not a more general sample of the user population. To evaluate whether this is the case, we examine how closely related our users are by examining the number of users who are friends on Facebook, and the number of user pairs with at least one common friend. Out of the 19,900 pairs of users $\binom{200}{2}$, 11 (0.05%) were direct friends and 13 (0.07%) were not direct friends but had at least one friend in common. Thus, our user population is not biased towards one small region of the Facebook social network.

4.3 Analysis of privacy settings

In this section, we begin by investigating the distribution of user-selected privacy settings. We then use our user survey to compare the desired privacy settings and actual privacy settings, quantifying the frequency of discrepancies. Finally, in the following section, we examine the potential for aiding users in managing privacy by automatically grouping related users.
 CHAPTER 4. ANALYZING AND IMPROVING FACEBOOK PRIVACY SETTINGS

<table>
<thead>
<tr>
<th>Type</th>
<th>Count</th>
<th>Me</th>
<th>SF</th>
<th>AF</th>
<th>FoF</th>
<th>Net</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Photo</td>
<td>65,182</td>
<td>&lt;0.1%</td>
<td>17%</td>
<td>37%</td>
<td>18%</td>
<td>1.3%</td>
<td>26%</td>
</tr>
<tr>
<td>Video</td>
<td>428</td>
<td>0.5</td>
<td>5.6</td>
<td>32</td>
<td>11</td>
<td>3.5</td>
<td>48</td>
</tr>
<tr>
<td>Status</td>
<td>37,144</td>
<td>0.1</td>
<td>9.7</td>
<td>35</td>
<td>4.5</td>
<td>3.4</td>
<td>47</td>
</tr>
<tr>
<td>Link</td>
<td>13,197</td>
<td>&lt;0.1%</td>
<td>5.4</td>
<td>36</td>
<td>9.2</td>
<td>2.0</td>
<td>47</td>
</tr>
<tr>
<td>Note</td>
<td>602</td>
<td>0.5</td>
<td>6.3</td>
<td>28</td>
<td>5.8</td>
<td>9.8</td>
<td>50</td>
</tr>
<tr>
<td>Total</td>
<td>116,553</td>
<td>&lt;0.1%</td>
<td>13%</td>
<td>36%</td>
<td>13%</td>
<td>2.0%</td>
<td>36%</td>
</tr>
</tbody>
</table>

Table 4.1: Existing privacy settings for all content items. The different content types possess similar privacy setting distributions, and the default (All Facebook users) is selected for the plurality of the items.

4.3.1 Existing privacy settings

We begin our analysis by examining the distribution of existing privacy settings. For each user who completed our survey, we collected the current privacy settings for all of their uploaded content (photos$^5$, videos, statuses, links, and notes). Table 4.1 presents an overview of the aggregated privacy settings.$^6$

We make two observations. First, out of 116,553 content items, 41,437 (36%) are shared with default privacy settings, meaning they are each visible to over 750 million Facebook users. This fraction is significantly higher than users indicated they desire (20%, discussed below in Table 4.2), suggesting that the users have not bothered to change the privacy setting from the default. Second, while the various content types show similar distributions, we note that photos have the most privacy-conscious setting: photos have the highest fraction of Some Friends, All Friends, and Friends-of-Friends, and the lowest fraction of Everyone. This suggests that users are more aware of the privacy settings for photos, implying that our survey below (which focuses exclusively on photos) is likely to underestimate the frequency of privacy violations for other types of content (since we observed that other types of content are much more likely to have default settings).

Next, we take a closer look at the per-user settings distribution in order to determine the fraction of users who have changed none, some, or all of their privacy settings from the default. To do so, we calculate the fraction of each user’s content that remains at the default setting; Figure 4.2 presents the cumulative distribution of this fraction across our 200 users. We observe that the fraction of users who have changed either all or none of their privacy settings varies according to content

---

$^5$We do not consider the special album “Profile Pictures”, as the user’s profile picture is required to be publicly visible. We also disregard a total of 60 additional photo albums containing 7,540 photos for which the Facebook API returned uninterpretable privacy settings.

$^6$Note that we also include the legacy Networks (Net) setting, indicating that all users in the same network (e.g., university or workplace) should be able to access the photo. This setting can no longer be selected by all users.
Figure 4.2: Cumulative distribution of the fraction of each user’s content that remains at the default privacy setting, for the five different content types. The distribution differs across the content types; with many users only having changed the settings for a subset of their content.

type: for photos, this fraction is 43%, while for notes this fraction is 74% (implying that for photos, for example, 57% of people have some, but not all, of their photos shared with the default privacy setting).

4.3.2 Desired privacy settings

We now turn to examine the privacy settings that are desired by users, with a focus on comparing the desired settings with the current privacy settings. Recall that to measure the users’ desired settings, we survey users concerning up to 10 of their uploaded photos. Of our 200 surveyed users, 193 (97%) had at least one photo (and could therefore answer at least one survey question) and 155 (78%) had at least 10 photos (and could therefore answer all 10 survey questions). Additionally, Facebook also offers users the option of sharing photos with Networks [151]. We disregard this feature because many users are not members of networks and are unable to select this setting; this affects approximately 1.3% of photos. In total, our users answered questions concerning 1,675 photos (907 randomly selected photos and 768 random photos with non-default privacy settings).

It is important to note that while we selected photos independent of the albums to which they are assigned, Facebook’s privacy settings are per-photo album rather than per-photo. We now briefly examine how many albums our random photo selection strategy covered. In total, the randomly selected photos came from 578 distinct albums. Our users had a total of 752 albums, meaning that we covered over 76% of all possible albums. Similarly, the non-default-privacy-setting photos came from 449 distinct albums out of 586 total non-default-privacy-setting albums, for a similar coverage of over 76% of all possible albums. Thus, our strategy of randomly selecting photos did not bias our survey towards a minority of the albums.
CHAPTER 4. ANALYZING AND IMPROVING FACEBOOK PRIVACY SETTINGS

<table>
<thead>
<tr>
<th>Actual setting</th>
<th>Desired setting</th>
<th>Me</th>
<th>SF</th>
<th>AF</th>
<th>FoF</th>
<th>All</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Me</td>
<td></td>
<td>3</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>2</td>
<td>15</td>
</tr>
<tr>
<td>SF</td>
<td></td>
<td>3</td>
<td>12</td>
<td>28</td>
<td>3</td>
<td>0</td>
<td>46</td>
</tr>
<tr>
<td>AF</td>
<td></td>
<td>38</td>
<td>2</td>
<td>184</td>
<td>25</td>
<td>42</td>
<td>291</td>
</tr>
<tr>
<td>FoF</td>
<td></td>
<td>16</td>
<td>8</td>
<td>80</td>
<td>15</td>
<td>22</td>
<td>141</td>
</tr>
<tr>
<td>All</td>
<td></td>
<td>46</td>
<td>23</td>
<td>171</td>
<td>56</td>
<td>118</td>
<td>414</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>106</td>
<td>50</td>
<td>465</td>
<td>102</td>
<td>184</td>
<td>907</td>
</tr>
</tbody>
</table>

Table 4.2: Comparison of the actual privacy settings and desired privacy settings for randomly-selected photos. The shaded cells represent instances where two are the same; this only occurs in 332 (37%, ± 3.14%) photos. When the two are different, they are more often shared with more users than desired (443 photos) than fewer users than desired (132 photos).

We divide our analysis into two parts, first focusing on a random selection of photos, and then focusing on photos with non-default privacy settings.

4.3.2.1 Randomly-selected photos

Table 4.2 presents the results of our survey for the 907 randomly selected photos, counting the number of photos with each combination of desired setting (columns) and actual setting (rows). First, we observe that for only 332 (37%, ± 3.14%\(^7\)) of photos do the actual and desired settings match; indicating that 63% of the time, current privacy settings do not match users’ expectations. Second, we observe that if we focus on the 575 photos that have incorrect privacy settings, 443 (77%, ± 3.44%) of them are shared with more users than desired. Third, and most worrisome, 296 (51%, ± 4.09%) of the 575 photos with incorrect privacy settings are incorrectly shared with all 750 million Facebook users. Taken together, our observations indicate that the problem of privacy management is endemic on Facebook—nearly two out of three of photos have incorrect privacy settings, and over half of these are incorrectly shared will all other Facebook users.

4.3.2.2 Photos with non-default privacy settings

One cause of the observations in the previous section are poor defaults: since it is known that users do not always adjust default settings, many of the photos could have incorrect settings because users have not bothered to adjust them. In order to shed light on the frequency of default settings causing privacy violations, we turn to examine only those photos which have non-default privacy settings.

\(^7\)All reported confidence intervals represent 95% confidence intervals.
CHAPTER 4. ANALYZING AND IMPROVING FACEBOOK PRIVACY SETTINGS

<table>
<thead>
<tr>
<th>Actual setting</th>
<th>Me</th>
<th>SF</th>
<th>AF</th>
<th>FoF</th>
<th>All</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Me</td>
<td>2</td>
<td>6</td>
<td>4</td>
<td>0</td>
<td>4</td>
<td>16</td>
</tr>
<tr>
<td>SF</td>
<td>2</td>
<td>12</td>
<td>29</td>
<td>8</td>
<td>11</td>
<td>62</td>
</tr>
<tr>
<td>AF</td>
<td>40</td>
<td>8</td>
<td>237</td>
<td>40</td>
<td>69</td>
<td>394</td>
</tr>
<tr>
<td>FoF</td>
<td>39</td>
<td>17</td>
<td>148</td>
<td>45</td>
<td>47</td>
<td>296</td>
</tr>
<tr>
<td>All</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Total</td>
<td>83</td>
<td>43</td>
<td>418</td>
<td>93</td>
<td>131</td>
<td>768</td>
</tr>
</tbody>
</table>

Table 4.3: Comparison of the actual privacy settings and desired privacy settings for photos with non-default privacy settings. The shaded cells represent instances where two are the same; this only occurs in 296 (39%, ± 3.45%) photos. When the two are different, they are shared with more users than desired (254 photos) with approximately the same frequency as fewer users than desired (218 photos).

settings. Since these photos, by definition, have had their privacy settings adjusted, we can see if the adjusted privacy settings better match users’ expectations.

Table 4.3 presents the survey results for the 768 photos with non-default privacy settings. We observe that the fraction of correct privacy settings (296 photos or 39%, ± 3.45%) is approximately the same as the randomly selected photos. This indicates that even photos where the user has explicitly adjusted privacy settings still do not match the users’ expectations the majority of the time. However, we also observe that the fraction of incorrect photos that are shared with more users than expected (54%, ± 4.50%) is much more even, when compared to the same fraction for randomly selected photos (77%, ± 3.44%). This suggests that while poor privacy defaults cause photos to be shared with more users than expected, users who are cognizant enough to modify their settings still have significant difficulty ensuring their privacy settings match their expectations.

4.3.2.3 Summary

Our analysis reveals that while users are uploading significant amounts of content to Facebook, almost half of the content is shared with the default privacy settings, which expose the content to all Facebook users. Users in our survey reported that this was the desired setting only 20% of the time, suggesting that the default settings are poorly chosen. More worringly, even for photos for which the privacy settings have been modified by the user, the modified privacy settings match users expectations less than 40% of the time. This strongly suggests that users are having trouble correctly configuring their privacy settings and calls for new tools to manage privacy.
CHAPTER 4. ANALYZING AND IMPROVING FACEBOOK PRIVACY SETTINGS

4.3.3 Improving privacy tools

As our final point of analysis, we examine the potential for assisting users in managing their privacy. Specifically, we focus on friend lists, a mechanism for users to group their friends that is similar to the Circles feature of Google+. We explore whether the friend lists could be automatically populated using community detection algorithms [23, 39, 113, 114, 127] over the social network.

Figure 4.3: Cumulative distribution of the sizes of observed friend lists.

To do so, we examine the friend lists of our 200 surveyed users using the Facebook API. The cumulative distribution of the sizes of the 233 friend lists we examine is shown in Figure 4.3. More than 50% of friend lists have more than 10 members, while 20% of the lists have more than 30 members, which indicates the potential difficulties with manually generating and maintaining such large lists of friends.

Figure 4.4: The 8 automatically detected groups of friends from one of the authors’ Facebook social network. Nodes represent the author’s friends and links exist between pairs of friends who are also friends. Nodes with the same color are automatically grouped together by the community detection algorithm.

One potential solution to the challenge of privacy management lies in leveraging the social links
between the friends of a user to automatically group them into communities, where each community of friends can be used to create a friend list. We illustrate this in Figure 4.4, where we used a community detection algorithm [23] to automatically group the 144 Facebook friends of one of the authors into 8 friend lists.

For this approach to work effectively, users’ friend lists need to correspond to tightly-knit communities in the network graph. To verify the extent to which users in friend lists form closely connected communities, we examine the normalized conductance [95] of the existing friend lists, whose value ranges from -1 to 1, with strongly positive values indicating significant community structure. Prior studies of social network graphs have found that normalized conductance values greater than 0.2 correspond to strong communities, that could be detected fairly accurately by community detection algorithms [95]. We analyzed the conductance values for our 233 friend lists and we found a significant positive bias. Over 48% of the friends lists have values larger than 0.2, suggesting that a large fraction of friend lists could be automatically inferred from the social network.

4.3.4 Discussion

We now briefly discuss a few issues brought up in the preceding analysis.

Measuring privacy In general, privacy is difficult to measure and quantify, especially since it’s hard even for users themselves to quantify. For example, photos alone are likely to have wildly varying privacy requirements, depending on who is in the photo, where it was taken, etc. In our survey, we simply treated all privacy violations as being equal, even though this is certainly not true in practice. In future work, we will explore mechanisms for measuring the “importance” of the various privacy violations, potentially by asking the users or using machine learning approaches on content metadata.

Additionally, when measuring the users’ ideal privacy settings, we are treating the users’ answers as ground truth. This may not always be the absolute ground truth, as the users’ answers may vary with time (as social relationships change), or the users’ may have not fully thought through the implications of a given setting. However, other user studies [38] are subject to the same limitation.

Reasons for incorrect settings In this study, we refrain from exploring why the privacy settings were incorrect. However, we note that such a study is non-trivial: just a few of the reasons for privacy violations include poor human–computer interaction mechanisms, the static nature of privacy settings,
and the significant amount of work forced on the user to maintain the privacy of their content. We leave a full exploration of these to future work.

4.4 Friendlist Manager application

So far, we have explored the potential of leveraging the friend list mechanism to assist users in managing their privacy for content sharing. We found that the friend lists have a correspondence with the structure of the social network. In particular, many of the user-created friend lists correspond to communities in the user’s 1-hop social network, and hold the potential to be detected by existing community detection algorithms (e.g., [23]). Our intuition is also supported by previous studies [95] that demonstrate the existence of attribute-based communities on Facebook.

In this section, we leverage this observation to provide the users with more natural and effective ways to aggregate friends’ content and express privacy controls on their own content. We present the Facebook application Friendlist Manager, which automates the creation and maintenance of Facebook friend lists. In detail, Friendlist Manager analyzes a user’s 1-hop social network, detects communities, proposes the communities to the user as friend lists, allows the user to tweak the friend lists, and finally creates the friend lists for the user. Additionally, Friendlist Manager examines any previously existing friend lists to locate friends who may be tightly connected to friends in the friend list but not a member of the friend list; if such friends are found, they are also presented to the user for consideration.

4.4.1 Detecting communities

The first step in our approach is to figure out how to detect “strong” communities in user’s 1-hop social network. Below, we describe the community detection algorithms used in this study.

Community detection in large networks is a well-studied problem with a rich literature and many proposed approaches [23, 39, 113, 114, 127]. We leverage existing work on community detection to automatically infer friend lists. We require two types of community detection algorithms: one, which detects remaining members of a community if a partial set of community members are given, and second, those which globally partitions a given network into multiple communities. The use of above two classes of algorithms helps us to to expand existing friend lists (i.e those which the user has already created) and to create new friend lists for the user. It should be noted that new friend lists
CHAPTER 4. ANALYZING AND IMPROVING FACEBOOK PRIVACY SETTINGS

created can be overlapping friend lists, because a friend of a user could belong to multiple friend lists (e.g. a friend could be part of a “football team” and also a part of “hometown friends”).

Since Facebook only allows applications to have knowledge of each user’s 1-hop social network (i.e., only the user’s friends and the connections between them), we use the 1-hop network of a user to detect communities. To create new friend lists for a user, we use the BGLL algorithm proposed by Blondel et al. \cite{23}.\footnote{In brief, BGLL works in the following way: It initializes each node of the graph as an isolated community. Then, it searches for ways to merge two existing communities that would result in the highest increase in modularity \cite{113}. The algorithm continues until all of the nodes are in a single community, and then picks the stage in the algorithm with the highest modularity as the final division. BGLL has a very low computational overhead, allowing our application to give results quickly.}

Unfortunately, the BGLL algorithm only assigns each friend to a single community. To further detect overlapping friend lists for the user, we expand each of the BGLL-detected communities using the algorithm proposed by Mislove et. al. \cite{95} that detects other members of a community, given a partial set of community members by finding members (outside the existing community) which are tightly connected to the existing members.

Finally, to propose new members to existing friend lists, we again leverage the algorithm by Mislove et. al. to expand existing communities \cite{95}.

4.4.2 Simulating Friendlist Manager

In this section, we evaluate the potential for detecting communities in the 1-hop network of a user using community detection algorithms. We focus only on detecting non-overlapping communities in the user’s 1-hop network. Work in \cite{95} shows evidence of existence of overlapping communities in social networks and evaluates the possibility of expanding the set of potential community members given a seed set of existing members.

In order to rate the “quality” of a detected community, we use normalized conductance (a metric proposed in \cite{95}) to evaluate the community’s strength. Normalized conductance is defined as the fraction of the community’s links that lie within the community, relative to what would be expected from a random graph with the same degree distribution. Similar to modularity, the value of normalized conductance varies from -1 to 1, with a positive value indicating community quality.

\footnote{We experimented with other off-the-shelf community detection algorithms as well and observed similar results. More details are in Section 4.4.2.}

36
better than random, 0 representing no more community structure than random and a negative value indicates worse community quality than a random graph.

<table>
<thead>
<tr>
<th>Network</th>
<th>Nodes</th>
<th>Links</th>
<th>Avg. deg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Facebook City A [151]</td>
<td>3.0 M</td>
<td>46 M</td>
<td>15.2</td>
</tr>
<tr>
<td>Facebook City B [151]</td>
<td>2.9 M</td>
<td>40 M</td>
<td>14.2</td>
</tr>
<tr>
<td>Facebook New Orleans [148]</td>
<td>63 K</td>
<td>1.6 M</td>
<td>25.6</td>
</tr>
</tbody>
</table>

Table 4.4: Number of nodes, links, and average user degree in the samples of the Facebook networks used to evaluate community detection algorithms.

We use 3 datasets collected from Facebook to evaluate the quality of communities detected. The first two are Facebook City A and Facebook City B, two (anonymized) Facebook regional networks collected by Wilson et al. [151]. We also use the Facebook New Orleans network [148]. The high-level statistics of these networks are shown in Table 4.4. We selected 1,000 users at random from each of these networks for this work.

![CDF](image)

Figure 4.5: Cumulative distribution of the maximum normalized conductance values per user in Facebook New Orleans [148]. In 40% of cases BGLL creates communities with conductance greater than 0.2, and all methods create communities with almost equally good conductance values (the results are similar for the other two Facebook networks).

We detect communities for each of them using three different algorithms: BGLL, CNM [26], and Walktrap [124]. We present the cumulative distribution of the maximal normalized conductance per user for Facebook New Orleans in Figure 4.5 (results for the other networks are similar). Figure 4.5 shows that almost 40% of the communities that the algorithms detect have normalized conductance value more than 0.2, indicating that the algorithms are able to find reasonably good quality communities for many users. Also, we notice that all three of these algorithms find communities of similar quality. So, we do not compromise on the quality of the communities by choosing the BGLL algorithm. When looking at the community quality of detected communities at a per-user level, we observe that a majority of users (56%) have at least 10% of their total number of communities
CHAPTER 4. ANALYZING AND IMPROVING FACEBOOK PRIVACY SETTINGS

with normalized conductance greater than 0.2. This suggests that for most users, there are a few communities that are tightly knit in their local neighborhood.

Figure 4.6: Cumulative distribution of the size of the communities per user in Facebook New Orleans. In more than 90% of cases, BGLL creates communities with fewer than 20 members.

Figure 4.6 presents a cumulative distribution of the community sizes that are found; this further indicates that BGLL is able to find small communities. At a per-user level, we find that 94% of users have a median community size less than 20.

Since we observe that automatically detected communities have good conductance values, they are likely to represent small groups of closely knit friends. This observation is compatible with our intuition that these communities can be directly converted into meaningful and useful friend lists.

We now present the user interface of Friendlist Manager and explain the functionalities we provide for Facebook users. We then describe the results from the deployment of the application.

4.4.3 User interface

The Facebook application Friendlist Manager is available at http://apps.facebook.com/friendlist_manager. The application first requests permissions to access the user's data, and then downloads and analyzes the user's local social network. The user is then taken through two steps. In the first step, if the user has existing friend lists on Facebook, Friendlist Manager proposes new additions to each existing friend list. The user can update existing friend lists with any of the proposed additions, or can skip to the next step directly. In the second step, the user is presented with a set of proposed new friend lists. A screenshot of our application in step two is shown in Figure 4.7. The user is allowed to remove users from friend lists or drag-and-drop users between friend lists in both steps.
CHAPTER 4. ANALYZING AND IMPROVING FACEBOOK PRIVACY SETTINGS

Figure 4.7: Screenshot of user interface of the application with proposed new friend lists.

Figure 4.8(a) shows the interface for proposed additions to existing friend lists, with the following functionalities:

- **Collapsible tab groups** We present two tabs within each friend list box – one showing the existing members of the friend list and the second showing the proposed new additions to that friend list. The newly proposed members are shown in red.

- **Delete members** The user can delete any members shown in the friend list box (both newly proposed additions and existing members) by clicking the minus button next to each member.

- **Drag-and-drop friend** allows the user to drag one friend out of the friend list and drop into another friend list, if the user feels that this friend should belong to another friend list.

- **Update friend list** Once the user has finished editing the friend list, she can save the friend list to Facebook by clicking “Update List” button at the bottom of each friend list.

Figure 4.8(b) shows the interface for newly proposed friend lists, with the following functionalities:

- **Merge friend lists** The user can merge two or more of the proposed friend lists by selecting the merge button on one friend list and selecting others from a drop-down menu.
CHAPTER 4. ANALYZING AND IMPROVING FACEBOOK PRIVACY SETTINGS

Figure 4.8: Screenshot of (a) an existing friend list with proposed additions and (b) a newly proposed friend list with merge friend list functionality.

- **Delete friend lists** The user can delete any friend list which is not meaningful.
- **Delete members** This is the same feature as provided in step one.
- **Drag-and-drop friend** This is the same feature as provided in step one.
- **Create friend list** Create the friend list on Facebook.

Here, we give more explanation about the Friendlist Manager’s functionality: First, in each proposed friend list, the order of the friends is based on the number of common friends with the user, meaning that the first friend in each friend list has the highest number of common friends with the user. This results in an ordering of members in each friend list from most-strongly-connected to least-strongly-connected. Second, the per-friend delete button and the drag-and-drop functionality allow the user to edit the proposed friend lists in a precise way, correcting all errors in assignment that are made by the algorithm. Third, in each friend list, the two buttons “View all names” and “Hide some names” are used to make the interface more clear for users (especially when users have many friends resulting in large proposed friend lists). Finally, the button “Create List” automatically
CHAPTER 4. ANALYZING AND IMPROVING FACEBOOK PRIVACY SETTINGS

Table 4.5: Topological characteristics of authors’ 1-hop Facebook subgraph along with number of friend lists detected by Friendlist Manager, number of friend lists finally users create on Facebook and time taken by users to use Friendlist Manager for modifying and creating these friend lists on Facebook.

<table>
<thead>
<tr>
<th>User</th>
<th>Friends</th>
<th>Links between friends</th>
<th>friend lists detected by Friendlist Manager</th>
<th>friend lists created by user</th>
<th>Total time in Friendlist Manager</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>51</td>
<td>126</td>
<td>6</td>
<td>5</td>
<td>2.4 min</td>
</tr>
<tr>
<td>2</td>
<td>121</td>
<td>608</td>
<td>12</td>
<td>9</td>
<td>5.1 min</td>
</tr>
<tr>
<td>3</td>
<td>613</td>
<td>18,820</td>
<td>20</td>
<td>9</td>
<td>4.3 min</td>
</tr>
<tr>
<td>4</td>
<td>94</td>
<td>622</td>
<td>6</td>
<td>5</td>
<td>8.8 min</td>
</tr>
<tr>
<td>5</td>
<td>149</td>
<td>1,219</td>
<td>5</td>
<td>3</td>
<td>1.1 min</td>
</tr>
</tbody>
</table>

creates the proposed friend list on Facebook; after it is created, the friend list disappears from the screen.

More information about using Friendlist Manager is available at:

http://friendlist-manager.mpi-sws.org/

4.4.4 Experimental evaluation

Now we briefly describe the results of the Friendlist Manager application when we first deployed to the five authors of this study. In this evaluation, we focused on the new friend lists created by each user. We presented some basic information on the authors’ social networks in Table 4.5. From Table 4.5, we observed that totally all 5 authors created 31 friend lists on their Facebook accounts and on average each user created 69% of the friend lists proposed by Friendlist Manager. Moreover, each user spent an average of 4 minutes to create all their friend lists.

Later we performed another evaluation for all Friendlist Manager users up to April, 2015, when our application failed to fully function due to the Facebook API upgrade. Up through Aug. 30, 2014, a total of 5,644 users installed the application, at a rate of 5 users per day on average. We observe that users found Friendlist Manager to be helpful; 2,878 (51%) of the users allowed Friendlist Manager to create at least one new friend list, and 766 (14%) of the users updated at least one existing friend list using our app. Additionally, we got 127 total page likes as-of 2016 for the official Friendlist Manager app page.
4.5 Summary

Over the past decade, OSN services have gained increasing popularity all over the world. Privacy has become an extremely important concern when users use the free services provided by OSN services, and share large amounts of personal contents. Unfortunately, there has been no in-depth study of users’ privacy settings on sites like Facebook.

In this chapter, we first performed a large-scale measurement study to quantify the magnitude of the privacy problem that users experience. We deployed a survey, implemented as a Facebook application, that queries Facebook to select content to ask the user about, as well as collects the current privacy settings for the users uploaded content. In brief, we found that the Facebook privacy settings match users’ expectations only 37% of the time, indicating that current settings are incorrect the majority of the time.

To explore the potential to assist users in selecting appropriate privacy settings, we performed an evaluation of friend lists, which suggests that the social network can be automatically leveraged to aid users in selecting groups of friends to share content with. Thus, we developed a Facebook application Friendlist Manager that uses the social network to help users generate friend lists conveniently. This can help users automate the creation and maintenance of meaningful friend lists, as a basis for content sharing. We deployed Friendlist Manager to over 5,600 Facebook users as-of April, 2015.

Our tool Friendlist Manager is complementary to recent work on privacy “wizards” [38], which uses machine learning algorithms to infer communities. One potential advantage of leveraging the structure of the social network is the potential to easily update the friend lists as the user forms or breaks friendships. Many subsequent studies used a methodology similar to ours and reported results that corroborate our findings. For example, Madejski et al. [102] extended this study to consider all textual content, rather than limit the evaluation to photos.

Additionally, our user dataset has been used in a follow-up study by Mondal et al. [100]. They performed a large-scale analysis on the social access control lists (SACLs) that were specified and used by over 1000 Facebook users who used the Friendlist Manager application. They found that the SACLs are often reused, suggesting that simply making recent SACLs available to users is likely to significantly reduce the burden of privacy management on users. These studies provide valuable insights for site operators in designing new user interface to enhance user privacy.

So far, we have examined the OSN privacy problem from user activities in content sharing, quantifying the privacy violations and mismatched user expectations. In the next chapter, we will turn to further understand how OSN site operators utilize user data in the targeted ad services.
Chapter 5

Analyzing Ad Auctions and Evaluating User Value in OSNs

In the previous chapter, we quantified the extent to the privacy violations of user content sharing activities on Facebook. We found that privacy settings match users expectations only 37% of the time, and when incorrect, almost always expose content to more users than expected. To help users better manage their privacy, we implemented a new tool Friendlist Manager, which can automatically create meaningful friend lists. However, another underlying privacy issue is that users still do not full control over their data, because OSN site operators have access to all user data, and they are using such data for their targeted ad services.

In this chapter, we turn to examine the privacy implication about how OSN sites are monetizing user data to provide their ad services for third parties, such as advertisers, using user’s demographic information, interests, and so on. We bring visibility to OSN ad markets, focusing on Facebook. We demonstrate that the undocumented feature that suggests bids to advertisers is most likely calculated via sampling recent winning bids. We then explore how this feature can be used to explore the relative value of different user demographics and the overall stability of the advertising market. We first presented this study as a conference paper in COSN’14 [91].

5.1 Motivation

Advertising is now the economic underpinning of much of the Web; large ad networks (e.g., Google’s ad network [63]) serve advertisements for millions of web sites. Many of these ad services are implemented as auctions, with individual advertisers bidding on specific keywords, pages, or
search terms. These auctions are extremely popular with advertisers (Google alone earned over $67 billion in ad revenue in 2015 [62]) and are well-studied in the research literature [52, 139, 150, 159].

A new type of ad network has emerged [70]: closed-site ad services run by OSNs such as Facebook. Unlike prior systems, where the ad network was forced to infer user information from cookies, browsing history, and search terms, OSN-based ad services are provided demographic information directly by the users themselves. As a result, advertisers are able to target users directly (via profile attributes), rather than targeting keywords or search terms. Although OSN-based ad services are nascent, they already carry a significant number of ads: Facebook alone had over $17 billion in ad revenue in 2015 [49].

Unfortunately, there has been little academic study of these ad networks, and OSNs have released very little data about their ad markets; the most in-depth numbers are from U.S. Securities and Exchange Commission (SEC) filings by the OSNs, which are typically at per-continent-per-fiscal-quarter granularity. Thus, researchers have little visibility into the dynamics of these markets, and it remains unclear which user demographics are the most valuable to advertisers (and therefore to the OSNs) and how stable these values are over time.

In this chapter, we develop techniques that will allow researchers to measure and understand how OSNs leverage user data in their ad markets. We focus on Facebook (currently the largest OSN ad market) and make three contributions: First, we explore how the suggested bid—a common feature of ad services that suggests prices to bid for a given target demographic—can provide insights on the revenue attainable from different users. On Facebook, the suggested bid is an undocumented feature, and the internal algorithm that Facebook uses is not public. However, we demonstrate that this feature is likely based on a sample of the recent winning bids on users in the target demographic, and we provide strong supporting evidence for this hypothesis by conducting an experiment where we actively participate in the ad market.

Second, we demonstrate how researchers can use the suggested bid data. The raw data returned from the queries are noisy due to the sampling methodology, but we demonstrate that repeated sampling of the ad market can provide consistent results with distinctive trends. We verify that the derived relative revenue per user correlates well with ground-truth figures from Facebook’s SEC filings [46]. While this methodology focuses on Facebook, it likely can be applied to other OSNs that provide suggested bids for placing advertisements.

Third, we use the suggested bid mechanism to explore two questions about how different users contribute to Facebook’s revenue: How do the advertising prices compare across different demographics, and how stable are the prices for different target demographics over time? We
explore different attributes, including location, age, and user interests, and provide a summary of the
distribution of prices paid to advertise to different user demographics.

The rest of this chapter is organized as follows: Section 5.2 describes the data collection
methodology, examines the properties of the suggested bid data, and explores how it can be interpreted
and used. Section 5.3 presents an analysis of the current Facebook ad market, exploring the prices of
different demographics and the stability over time, and Section 5.4 summarizes.

5.2 Suggested bids

We now describe our approach for measuring the Facebook Advertising Platform using suggested
bids. We analyze the properties of the suggested bid data, with a goal of determining what the
suggested bids represent and how they are calculated by Facebook. Finally, we describe how we
interpret and use the data for our analysis in the following section.

5.2.1 Collecting suggested bids

Figure 5.1: Screenshot of Facebook’s ad creation webpage, showing the suggested bid (bottom right)
for the selected targeting parameters. We programmatically collect the suggested bid information.

Facebook provides an API\(^1\) that allows advertisers to create and manage ads efficiently by making
different API queries. Unfortunately, the Ads API is restricted to high-volume advertisers and we
were unable to obtain access. Instead, we obtain suggested bid data from Facebook’s Ad Creation
Web page,\(^2\) a screenshot of which is shown in Figure 5.1.

\(^1\)https://developers.facebook.com/docs/ads-api/
\(^2\)https://www.facebook.com/advertising/
CHAPTER 5. ANALYZING AD AUCTIONS AND EVALUATING USER VALUE IN OSNS

Query:

https://graph.facebook.com/reachestimate?targeting_spec=
{"countries":["US"],"age_min":21,"age_max":30,genders=[1]}
&currency=USD&accountId=XXX&access_token=XXXX

Response:

{"data": {"users":62984500,"bid_estimations":
[{"location":3,"cpc_min":54,"cpc_median":82,"cpc_max":144,
"cpm_min":3,"cpm_median":14,"cpm_max":83}]}}

Figure 5.2: Example query targeting U.S. males between 21 and 30. Also shown is the suggested bid response (in JSON format).

We programmatically send HTTP GET requests to the Facebook URL that serves the suggested bid data presented on the page. This URL accepts arguments representing the desired targeting parameters and returns a suggested bid to the user. A user must be logged in to make the suggested bid request, so we create a pool of three accounts to place the queries. These accounts have no prior advertising history or uploaded content.

The suggested bid response from Facebook actually contains 7 different values: the estimated audience size (i.e., the number of Facebook users in the target demographic), the CPC min/med/max, and the CPM min/med/max. An example of query and response data is shown in Figure 5.2. It is important to note that all suggested bid data is public for all advertisers, and contains no personally identifiable information for either the other advertisers or other Facebook users.

5.2.2 Suggested bid observations

Given the suggested bid data made available by Facebook, we now provide five observations about the bid data. To help illustrate these observations, we collect a data set consisting of 1,000 suggested bids in quick succession (i.e., issuing 1,000 queries back-to-back within 30 seconds), separately targeting each of the 204 countries that Facebook supports. Thus, for each country, we collect a set of 1,000 suggested bids obtained in quick succession, where each bid uses the default targeting parameters with the exception of specifying the respective country (e.g., the default gender with different accounts and observed no significant differences in characteristics of the returned suggested bid data.

The audience size is returned with a granularity of 20 users, presumably to provide user privacy. The location parameter that is provided in the result is undocumented, but likely to the different locations on Facebook in which the ad will be shown.
parameter targets both male and female users).

**Skewed distribution** Our first observation is that the suggested bid data is highly skewed, with the median of the suggested CPM almost always being significantly closer to the minimum than the maximum. For example, if we target users in the United States, the minimum of the suggested CPM is typically between $0.03 and $0.07, the median is typically between $0.08 and $0.18, and the maximum is typically between $0.80 and $2.00. This property holds regardless of the targeting parameters that we choose.

![Graph](image.png)

Figure 5.3: 1,000 suggested bids for three different sets of targeting parameters with massively different audience sizes (note the log-scale on the y-axis). Significant variance is observed, as well as a skewed distribution, across all three countries.

<table>
<thead>
<tr>
<th>Country</th>
<th>CPM min</th>
<th>CPM med</th>
<th>CPM max</th>
</tr>
</thead>
<tbody>
<tr>
<td>US</td>
<td>0.176</td>
<td>0.143</td>
<td>0.260</td>
</tr>
<tr>
<td>NZ</td>
<td>0.143</td>
<td>0.110</td>
<td>0.273</td>
</tr>
<tr>
<td>AG</td>
<td>0.0</td>
<td>0.308</td>
<td>0.358</td>
</tr>
</tbody>
</table>

Table 5.1: Comparison of the coefficient of variation of the CPM values for the three different countries. Significant variance is observed, especially for the CPM maximum values.

**Significant variance** Our next observation is that multiple suggested bids with the same targeting parameters show significant variance over short time periods. For example, consider the graphs presented in Figure 5.3, which shows the 1,000 suggested CPM bids for three different countries with very different populations (United States, 159M users; New Zealand, 2.2M users; and Antigua and Barbuda, 29K users). In all three cases, the minimum, median, and maximum values show significant variance, even from query to query (queries were roughly spaced 35 milliseconds apart). To quantify the variance, we calculate the coefficient of variation (the standard deviation divided by

---

6The minimum and median values for Antigua and Barbuda take on fewer values, but this is likely an artifact of the $0.01-granularity of the data returned.
the mean) of the distribution, and present the results in Table 5.1. In almost all cases, significant variance is observed, with the CPM maximum always showing the highest coefficient of variation.

![Plot](image.png)

**Figure 5.4:** Audience size vs. coefficient of variation (standard deviation divided by mean) of suggested CPM bids for all 204 countries. No correlation is observed for CPM minimum, median, or maximum.

**Variance independent of audience size**  Our third observation is that the variance observed is independent of the audience size. We compare the audience size versus the coefficient of variation of the CPM minimum, median, and maximum values for each country in Figure 5.4. Across suggested bids from the 204 countries, we observe no correlation between the audience size and the coefficient of variation of any of the CPM values: the correlation coefficients are -0.02 (CPM minimum), -0.08 (CPM median), and -0.03 (CPM maximum). We observe similar results with audiences derived from over 100 different sets of targeting parameters including US states and zip codes, user interests, and relationship status.

**Variance across accounts**  Our fourth observation is that the suggested bids queried at the same time from different accounts show no correlation. To explore this, we used our multiple Facebook accounts described above and queried for suggested bids for the same target demographic at the same time from multiple accounts. Despite using a range of targeting parameters (resulting in a variety of audience sizes), we did not find any correlation between the suggested bids received by the different accounts (despite the fact that the queries were issued at the same time).

An example is shown in Figure 5.5, containing the CPM maximum value received by two accounts targeting US users. Visually, we can observe little correlation between the values returned by the two accounts; for example, the spike around 85 seconds observed by account 2 is never
CHAPTER 5. ANALYZING AD AUCTIONS AND EVALUATING USER VALUE IN OSNS

Figure 5.5: CPM maximum values for 100 successive suggested bids targeting US users, queried from 2 different accounts at the same time. The prices received by the two different accounts show no correlation.

reflected in account 1’s results. Moreover, the Pearson’s correlation coefficient between the values received by the two accounts is -0.01, further indicating the lack of a correlation.

Non-persistence of min or max   Our final observation is that the CPM minimum and maximum values do not “persist” from query to query. For example, consider the graph in Figure 5.3 (c), which shows the CPM data for Antigua and Barbuda. While there are only 29,580 Facebook users in the country (as measured by the audience size), the CPM maximum value varies repeatedly between $0.04 and $0.30 in less than 100 milliseconds. This indicates that the minimum and maximum are very likely not calculated from the same pool (or a rolling pool) each time; instead, as we describe in Section 5.2.4, we believe they are calculated by sampling from the pool of recent winning bids.

5.2.3 Reverse-engineering suggested bids

The suggested bid feature is not documented by Facebook; the most relevant documentation describes the purpose of the feature as helping advertisers select a bid that is likely to cause their ads to be shown to users [44]. We requested additional information from Facebook’s Advertising Support Team on how the suggested bids are calculated and received the following information (emphasis ours):

The suggested bid range you see when creating your ads is based on the bids that are currently winning the ad auction for the users you’ve chosen to target.

Thus, it is clear that the suggested bids are derived from the recent winning bids on the target users, but it is not quite clear exactly how they are derived.
Ultimately, the suggested bid algorithm is a black box; we are unlikely to be able to definitively reverse-engineer how they are calculated. Instead, we look for the most reasonable explanation for how suggested bids are derived given our observations. We present three hypotheses below and rule out two as unlikely.

**Hypothesis 1: Winning bids change rapidly** The first hypothesis is that the suggested bids are derived from the most-recent-k winning bids for the target users (for some value of k). If this were the case, the observed variance would be due to the set of recently-won bids changing rapidly. However, this hypothesis does not explain the significant variance observed on very short timescales for countries with very small audience sizes (e.g., the Antigua and Barbuda CPM from Figure 5.3 (c)); with such small audiences, it is unlikely that ads are served to these users quickly enough to account for the rapidly changing minimum and maximum values.

![Example probability distribution function of CPM maximum values](image)

Figure 5.6: Example probability distribution function of CPM maximum values for 20,000 successive suggested bids (US 25-year-old females interested in computer programming; 179,760 users). This distribution fails statistical tests for multiple common distributions, suggesting the absence of random noise.

**Hypothesis 2: Adding random noise** The second hypothesis is that Facebook is adding random noise to the returned suggested bid data (possibly to obfuscate the true value). To explore this hypothesis, we collected 20,000 suggested bids in quick succession for a small target population (25-year-old U.S. females interested in computer programming; 179K users); we then ran a number of statistical tests to see if the data matched a number of common statistical distributions (as would be expected were Facebook adding random noise): Uniform random, Gaussian, Cauchy, Log-Normal, or Logistic distributions. We found a poor fit for all distributions, with a p-value of less than $10^{-16}$. An example probability distribution function of one of these suggested bid sets is shown in Figure 5.6.
Hypothesis 3: Sampling winning bids The third hypothesis is that Facebook is sampling from the recent-\(k\) winning bids, and is reporting the minimum, median, and maximum of the sample; a diagram of this process is presented in Figure 5.7. This hypothesis is consistent with the format of the returned data (i.e., the fact that Facebook returns a minimum, median, and maximum) and is also consistent with our observations: we would expect to see significant variance from query to query, as different samples of the recent winning bids are used to generate each response. Moreover, this hypothesis is compatible with the variance being independent of audience size if the same \(k\) is used.

While verifying the correctness of Hypothesis 3 is likely only possible if one has access to Facebook’s internal systems, we rely on the fact that it is both a logical mechanism for calculating suggested bids and is consistent with all of our observations on the properties of suggested bids.

5.2.4 Validating suggested bids

We now validate that suggested bids can be used to measure the overall Facebook ad market by showing that suggested bids reflect changes to the marketplace and that they correlate with Facebook’s revenue.

Changes to the market To test our hypothesis that suggested bids are generated using recent winning bid data, we ran an experiment where we actively participate in the advertising market to see how quickly changes propagate into the suggested bids. To do so, we chose a small country (Seychelles, 26K Facebook users) with a low suggested CPM. We then created three accounts\(^7\) to advertise with, and submitted one ad campaign from each targeting Seychelles users. To visibly

---

\(^7\)We choose to create three accounts, rather than one, as Facebook places multiple advertisements from different advertisers on each page.
affect the market, each of our three advertising accounts bid a significantly higher CPM ($1.00) than the suggested CPM maximum ($0.16). We ran the advertising campaigns concurrently for 8 hours, receiving an average of 19,903 impressions to 3,543 users.

To measure the effect the campaigns had on suggested bids for Seychelles users, we also collected data on the suggested bids for targeting Seychelles users every 5 minutes using a separate account. In order to observe the changes, we started collecting this data 8 hours before all three of the advertisement campaigns became active; we also collected suggested bid data for the 8 hours during the campaign and for 44 hours after the campaigns ended.

Figure 5.8 presents the results of this experiment, showing the CPM medians and maximums before, during, and after our campaigns (the shaded region represents the time of the ad campaigns). We observed that our advertising campaigns were quickly reflected in the suggested bids.

Figure 5.8: (a) CPM median values and (b) CPM maximum values before, during, and after our three advertising campaigns targeting Seychelles users (the shaded region represents the time of the ad campaigns. We observed that our advertising campaigns were quickly reflected in the suggested bids.

Examining Figure 5.8 (b), we find that the suggested bids maximum rose dramatically from $0.16 up to $7.64; after we paused our campaigns, it fell back to a low price $1.47. Overall, this experiment shows that changes to the ad market are reflected in the suggested bids, and provides evidence for our hypothesis that the suggested bids data comes from a sample of the recent winning bids.

Comparison with Facebook’s revenue Finally, we compare the data we observe via suggested bids to the only ground-truth that we know of: Facebook’s SEC filings. In Facebook’s March 2013 10-Q filing, Facebook reports the Average Revenue Per User (ARPU) at the granularity of region (US+Canada, Europe, Asia, and the Rest of the World). To compare the suggested bid data to Facebook’s ARPU, we aggregate our CPM median data into the same regions and take the average

---

8The fractional values result from a change by Facebook on May 1, 2013 to provide more precise suggested bids.
CHAPTER 5. ANALYZING AD AUCTIONS AND EVALUATING USER VALUE IN OSNS

<table>
<thead>
<tr>
<th>Region</th>
<th>Facebook ARPU</th>
<th>Suggested Bid</th>
</tr>
</thead>
<tbody>
<tr>
<td>US, Canada</td>
<td>$3.50</td>
<td>1.00</td>
</tr>
<tr>
<td>Europe</td>
<td>$1.60</td>
<td>0.45</td>
</tr>
<tr>
<td>Asia</td>
<td>$0.64</td>
<td>0.18</td>
</tr>
<tr>
<td>Rest of World</td>
<td>$0.50</td>
<td>0.14</td>
</tr>
</tbody>
</table>

Table 5.2: Comparison of Facebook’s ARPU and CPM median suggested bids. We scale all values relative to the US+Canada region. We observe the same ranking of regions.

across all countries in the region (weighted by audience size). We then scale both Facebook’s ARPU and our aggregated suggested bid data relative to the US+Canada region. Of course, aggregating suggested bids in this way ignores many aspects of how revenue is generated (e.g., the activity level of different demographics), but can provide rough guidance on the relative revenue for different regions.

The results of this experiment are presented in Table 5.2. We observe similar trends between the two measures: Both Facebook’s ARPU and the suggested bids rank the regions in the same order, with the Europe and Rest of the World regions at approximately the same ratios. While our results are far from being conclusive, this result indicates that the suggested bid data that we obtain from Facebook’s advertising pages at least correlates with the distribution of Facebook’s revenue.

5.2.5 Using suggested bids

From the previous section, we conclude that the suggested bid data is most likely calculated by sampling from the recent winning bids for the target users. Since the properties of the suggested bids are somewhat unique, we now explore how researchers can use suggested bids to measure the Facebook ad market.

Multiple samples  Given that each suggested bid is most likely generated from a sample of the recent winning bids, it is clear that a single suggested bid may misrepresent the overall bid distribution. Instead, we collate multiple samples together, extracting the overall minimum, median, and maximum from the collated samples (i.e., for the remainder of the paper, all reported minima are the minimum across multiple suggested bid minima; the same holds for median and maximum).

Convergence  The next step is to choose how many samples to collate together. To do so, we examine how quickly different numbers of collated suggested bids converge towards the overall “true” minimum, median, and maximum. To explore this question, we use the 204-countries data...
from the previous section. Since we do not know the true distribution of the recent winning bids, we instead use the overall minimum, median, and maximum of each country’s 1,000 samples in its place.

Figure 5.9: Convergence of different numbers of collated suggested bids towards overall CPM minimum, median, and maximum values for each of the 204 countries (each country is represented by a line in each graph). The average value across all countries is shown as the dark dashed line. Very quick convergence is observed for minimum and median, as expected.

We observe that the minimum and median converge quite quickly: after only 25 suggested bids, both the minimum and median are within 15% of their eventual value (on average). Second, the maximum value converges more slowly (after 25 suggested bids, the maximum is within 30% of its eventual value, on average), which is expected due to the high variance and skewed distribution.

Choosing the number of suggested bids to collate together represents a tradeoff between accuracy and the load we place on Facebook. For the remainder of the paper, all of our reported data is the result of 25 collated suggested bids.
CHAPTER 5. ANALYZING AD AUCTIONS AND EVALUATING USER VALUE IN OSNS

5.2.6 Limitations

Before examining the results for suggested bids for different user demographics, we first discuss a few of the limitations of our methodology.

User accounts As mentioned before, the advertising history of accounts could be a factor in determining the suggested prices (i.e., Facebook may show different suggested bids to different advertisers). Unfortunately, it is difficult (and expensive) to build an account with successful advertising history, as a result, to control any impact of this factor, all the user accounts used in our experiment are new created with no prior advertising history or uploaded content. Thus, the results in this section are all comparable to each other, and we leave an exploration of the effect of advertising history on suggested bids for future work.

Facebook changes Our observations and methodology in this section are based on the current advertising model used by Facebook; if Facebook makes internal changes in their advertising model, the suggested bid model, or the way they utilize the user data, our results may no longer be valid. However, we believe that our work provides researchers with a new approach for determining how suggested bids are calculated by Facebook, as well as ground-truth data to compare against.

Correlation versus causation In our analysis in the next section, we examine how the suggested bids are correlated with different user demographics. Of course, correlation does not imply causation, and it is possible that other, unknown factors are responsible for our observed correlations. Regardless, our analysis presents the first measurements of the relative value of different user demographics in OSNs.

5.3 Analysis of bid data

We now explore the properties of Facebook’s ad auctions, using the suggested bid data.

5.3.1 Location

We examine how the location of the target demographic influences the ad auction winning bids, using the data set on 204 countries. We examine how the ad market CPM prices (represented by CPM median) correlate with the relative wealth of countries. To quantify the latter, we use GDP per capita [31], which is widely used in economics literature. The results are presented in Figure 5.10.
Figure 5.10: Scatterplot of CPM median versus GDP per capita for the 204 countries Facebook supports. Labeled are some of the countries with high CPM median values. All but a few countries have ad markets with very low CPMs.

As expected, we observe a correlation of 0.37 (statistically significant at the 0.001 level) between the GDP per capita and CPM prices. One notable outlier is Nigeria, which shows a CPM maximum on par with the U.S. while having a GDP per capita over an order of magnitude lower.

Figure 5.11: CPM maximum prices for all U.S. cities with population over 100,000. The color of each city corresponds to CPM maximum price (yellow to red represents increasing prices, and the size of the circle is in proportion to the number of Facebook users in each city.

We now dig deeper into a single country and explore the differences in CPM prices between multiple cities in the same country. We choose to focus on the U.S., as it is the most mature Facebook ad market with the largest number of users. We query for the CPM prices of all U.S. cities with a population over 100,000 (285 cities). The results are presented in graphical form in Figure 5.11, plotting both CPM maximum (more red color representing higher values) and population (circle size) for each city. We observe that certain cities such as Las Vegas, NV and Hartford, CT show CPM
maxima significantly higher than other cities like Cambridge, MA and Ann Arbor, MI, suggesting that certain cities have much more desirable users for advertisers to target.

Overall, we see dramatic differences in ad auction prices across different locations, with the most dramatic differences coming between users in different countries. As Facebook’s ad markets continue to evolve, we can use our suggested bid methodology to measure their relatively maturity.

Figures 5.12 (a) CPM median versus Age (b) Audience Size versus Age for three countries. CPM prices increase significantly with age for the United States and New Zealand, and massively different user populations are observed (note the log scale on the y-axis of the lower graph).

5.3.2 Age

We next explore how CPM median price is correlated with user age. We select the same three countries as before (the U.S., New Zealand, and Antigua and Barbuda), and retrieve suggested bids for users with different ages in each country. We note that Facebook’s age policies come into effect here: the smallest age that an advertiser can target is 13, and targeting age 65 (the largest age one can target) encompasses all users 65 and over. The results of this experiment are presented in Figure 5.12 (top), showing the CPM median for different ages. We observe that in both the U.S. and New Zealand, as the age of target users increases, the CPM median price increases as well. The trend is less clear for Antigua and Barbuda, which we suspect is due to the smaller user population and less-well-developed ad market.

From Figure 5.12 (bottom), we observe that in all three countries, there is a rapid rise in the audience size (the number of Facebook users) between 13 to 18, followed by a slow decline over
CHAPTER 5. ANALYZING AD AUCTIONS AND EVALUATING USER VALUE IN OSNS

the remaining ages. This observation corresponds strongly with previous studies about the age distribution of users in OSNs [147, 154].

5.3.3 Interests

For our final user attribute, we examine user interests in order to determine whether there are segments of the user population—based on particular interests—that are highly valued by advertisers. We are inspired by the wealth of studies on Web search auctions [79, 80, 137], which show there are certain keywords (e.g., “mesothelioma”) that command prices thousands of times more than the average. For this experiment, we focus on broad interests, which are interests derived by Facebook based on user activity.

![Figure 5.13: (a) Cumulative distribution of CPM median values for 129 interest categories for three countries. (b) Cumulative distribution of audience size for three countries (United States, New Zealand, and Antigua and Barbuda).](image)

We retrieve suggested bids on all 129 Facebook-provided broad interest categories in each of the three countries that we have considered so far, and present the cumulative distribution across categories in Figure 5.13 (a). Surprisingly, the distribution of CPM median prices are rather broad: the most-expensive category is more than 20 times more expensive than the least-expensive category in all three countries. Interestingly, the most expensive categories in all three countries correspond to users who are traveling, recently engaged, or who like Apple products.

In Figure 5.13 (b), we observe that the distribution of audience size across different broad categories are widespread for all the three countries (note the log-scale on the x-axis). There are a significant number of broad categories are very popular within each country, for example, the largest categories in the United States are Mobile Users (All), Travelers, and Music (All).

---

9The “jump” in audience size at age 65 is because 65 represents “65 and older.”
5.3.4 Price stability

In addition to studying the CPM prices of different demographics, we also study the stability of different market prices over time. To do so, we select four different sets of targeting parameters, designed to cover a variety of targeting parameter types and audience sizes: $G_1$: U.S. users (167M users), $G_2$: 21-50 year-old Canadian users (11M users), $G_3$: 25-40 year-old college graduated Brazilian users (4.7M users), and $G_4$: 13-15 year-old British users (1.2M users). We track each of these sets of targeting parameters by retrieving 25 suggested bids each hour for a period of three weeks (April 3rd, 2013 through April 23rd, 2013).

![Figure 5.14: Long-term tracking of CPM median prices over three weeks for four different sets of targeting parameters that show different properties with distinct trends over time.](image)

Figure 5.14 plots the CPM median values for these groups during our period of observation. We observe that while there is significant short-term variance in many of the groups, there are also a number of longer-term trends present. For example, group $G_1$ shows a periodic increases spaced out approximately one week apart, and groups $G_2$ and $G_3$ shows a multi-day increase starting approximately on 04/16. Moreover, certain groups—such as groups $G_2$ and $G_3$—show significant fluctuations in price (up to six-fold), while others—such as $G_4$—do not vary much over the study period. Overall, our results suggest that there are significant short-term and long-term dynamics present in Facebook’s ad auctions, which may be explained by the relative immaturity of different sectors of the market.

5.4 Summary

Advertising is now ubiquitous on the Web; most of the popular OSNs are funded via advertising on their site. While OSNs themselves have all information of their advertising models as well as how they utilize user data, there is little information that is shared with external researchers, advertisers,
even the users themselves. Thus, OSN ad markets remain quite difficult for researchers to study, even as these markets are growing in prominence.

In this chapter, we explored how site operators leverage user data in their online ad services. We examined Facebook ad auctions through the suggested bid feature, showing how this feature can be used by researchers to make inferences on properties of the ad market. We found significant differences in ad prices across different locations and user interests, and fewer differences by age. We also found prices to be variable over the long-term, but with distinct trends; this is consistent with OSN-based ad markets being in a nascent phase. While we only presented a subset of the experiments here, we explored suggested bids for many other targeting parameters that Facebook provides, including gender, precise interests, relationship status, education, and workplaces. In all cases, we observed fewer differences than for the targeting parameters presented.

While we are far from covering the entire space—the set of possible targeting parameters is prohibitively large—the results suggest that advertiser interest is focused on location, user interest, and age parameters (consistent with ads in other media). Though the results are not necessarily surprising, we present the first mechanism for quantifying the relative value of different user demographics towards the OSNs. For advertisers, our work offers some guidance especially when they start to advertise on Facebook users; for end users, we provide them a basic idea about how OSN services utilize their data, and how valuable they are towards advertisers and Facebook, based on their own demographic information and activities which fall under different interests.

We make all of our suggested bid collection code and collected data available to the research community at

http://osn-ads.ccs.neu.edu/

So far, we have examined how OSN site operators utilize user data for their ad services. In the next chapter, we will turn to examine how the role and behavior of third parties has changed with online services.
Chapter 6

Identifying Personal Information in Internet Traffic

In the previous chapter, we examined how OSN site operators are monetizing user-uploaded data and explored how advertising prices differ across user demographics and interests. More specifically, we explored the Facebook ad market, focusing on the suggested bid feature. We reverse-engineered the undocumented feature suggested bid, and demonstrated how researchers can use the suggested bid data to explore the relative value for users with different demographics. Unfortunately, it remains difficult for end users to understand the scope of privacy implications caused by third parties, which has changed with online content-sharing services and the rise of mobile applications.

In this chapter, the privacy implication we examine is about how third parties in the network gain access to user data, such as their name, gender, age, email, phone number, city, geo locations, and so on. We aim to bring visibility into user PI collected when accessing online services. The detection of user PI is a major challenge because user PI is transferred in a proprietary way by each service. We develop a novel method that can automatically discover various types of user PI carried within protocol fields of network traffic; the method includes techniques to filter out potential “containers” that do not actually carry user PI and extend the set of containers initially found with additional ones. We evaluate the false positive/negative rates of our proposed method and show examples of interesting findings, including what kinds of web sites or applications are more likely to transmit user PI and which types of user PI are most commonly collected. We first presented this study as a conference paper in COSN’15 [93].
CHAPTER 6. IDENTIFYING PERSONAL INFORMATION IN INTERNET TRAFFIC

6.1 Motivation

People heavily and constantly rely on services accessible through the Internet for professional, personal, and entertainment needs. These services are accessed not only via web sites, but also via dedicated applications on mobile devices, thus being accessible through a range of devices (PCs, tablets, and smartphones) and a variety of wired, wireless, and mobile networks. Even though users are often provided with privacy controls, these generally only affect flow of information to other users or third-party applications; users have no option of making their data private from service providers. Even worse, the limited visibility into app behavior coupled with the significant amount of data stored on smartphones makes it even harder for users to understand the extent to which these services are automatically collecting personal data.

In this chapter, we address this situation by developing techniques that automatically detect personal information (PI) traveling through the network as it is collected by services accessed via web browsers or mobile applications. This phenomenon is hereafter referred to as PI leaks. In contrast with related approaches that rely on “rooting” a user’s device [36, 55] or instrumenting applications or browsers [8, 116], we instead aim at a solution requiring access only to the network itself, because such an approach significantly lowers barriers to deployment. Moreover, it has the potential to achieve higher coverage since the system can leverage visibility on traffic from multiple users in learning how PI is transmitted (i.e., how it is encapsulated in proprietary ways). Although the approach is general, we choose to focus on HTTP, as this is the protocol on which both traditional web services and a large fraction of mobile applications base their communications.

In designing our approach, we make two high-level contributions. First, we underscore the difficulty of the problem of locating PI in network traffic by demonstrating that only a very small fraction of protocol fields actually convey PI, making our endeavor akin to “finding a needle in the haystack” (in § 6.4.1). We also show that an approach based on simple statistical analysis (e.g., selecting fields that are unique to a user, or common across different services) is not practical as it results in unacceptably high levels of false positives/negatives (in § 6.4.2).

Second, we develop a novel method based on (i) grouping data according to the domain name of the servers it is sent to and the key associated to it for the transmission, called a domain-key, and (ii) concluding that all of the domain-key combinations in a group are PI “containers” if a threshold subset of them are found to contain PI (§ 6.5.1). This subset of PI containers are identified through a list of seed rules manually crafted to locate PI of different types, including, but not limited to, users’ names, genders, email addresses, ages, geo-locations, cities, postal codes, and phone numbers.
CHAPTER 6. IDENTIFYING PERSONAL INFORMATION IN INTERNET TRAFFIC

(§ 6.5.2 and § 6.5.3). Then the coverage is extended by inferring additional containers by analogy with the seeded ones (§ 6.5.4).

The remainder of this chapter is organized as follows: Section 6.2 formally defines PI and discusses our threat model. Section 6.3 describes our data sets, and we explore detecting PI with statistical approaches in Section 6.4. Section 6.5 presents how we parse data traffic and proposes our seeded algorithm. Section 6.6 quantitatively evaluates seeded algorithm and presents interesting findings from PI discovered from a large commercial network. Finally, we summarize in Section 6.8.

6.2 Background

In this section, we provide a discussion on the characteristics of personal information (PI) and describe our assumptions and intended operating environment.

6.2.1 Personal Information (PI)

As explained in Section 1.1, the definition of personal information (PI) is “information about user’s demographics or other identifiable information, including personally identifiable information (PII), but not necessarily lead to distinguish or trace an individual’s identify”.

There are many different kinds of PI, like a user’s name and social security number, their current location when performing a purchase over the Internet, or even rich media information in the form of photos and videos. For simplicity, we focus on a text-based personal information (e.g., names, user identifiers, and locations) collected by web sites and smartphone applications. Further, to discern different characteristics of PI, we classify them along the following three dimensions.

Static vs. Dynamic  Static PI does not typically change over time (at least, over short- to medium-length time intervals). Examples include the user’s name, gender, phone number, and email address. In contrast, dynamic PI may change over such intervals; examples include the user’s geo-location, a user’s session ID, or the user’s set of personal interests.

Unique vs. Non-unique  Unique PI distinctly identifies a single (human) user from others. For example, a user’s email address or phone number uniquely distinguishes a user from the rest. On the other hand, non-unique PI may be shared by multiple users; examples include a user’s name, gender, or date of birth.
CHAPTER 6. IDENTIFYING PERSONAL INFORMATION IN INTERNET TRAFFIC

<table>
<thead>
<tr>
<th>Personal information</th>
<th>Static?</th>
<th>Unique?</th>
<th>Shared?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>✗</td>
<td>✗</td>
<td>✓</td>
</tr>
<tr>
<td>Email address</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Date of Birth</td>
<td>✓</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>Geo-location</td>
<td>✗</td>
<td>✗</td>
<td>✗</td>
</tr>
<tr>
<td>Username</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Tracking cookie</td>
<td>✗</td>
<td>✓</td>
<td>✗</td>
</tr>
</tbody>
</table>

Table 6.1: Examples of different types of textual PI, and the breakdown of PI along different dimensions.

**Shared vs. Distinct** The third dimension we consider is PI that, for a given user, is likely to be shared across services or distinct. An example of shared PI is mailing address of a user (presuming that the user provides factual information to each site). In contrast, distinct PI is potentially different for each web site (or domain name); examples include the login name of a user or the session identifier in a tracking cookie.

We provide breakdowns of how different examples of PI are classified along these dimensions in Table 6.1.

### 6.2.2 Assumptions, environment, threat model

In contrast to approaches that assume access to user devices (e.g., via browser plugins, “rooting”, or operating system modifications), we instead assume that network administrators wish to understand when their users’ personal information is being transmitted over the network. Thus, we have access to traces of network activity from a large group of users, as would be the case at a large corporation or university.

While web sites are constrained by the browser to only using HTTP to communicate with remote servers, smartphone applications are free to use any UDP/TCP protocol. However, as much as 40% of application traffic actually is HTTP [37, 158], presumably to re-use many of the same APIs as web-based services and to avoid certain firewalls. Hence, we only consider PI leaks that occur over HTTP, but our approach could easily be extended to other protocols if given appropriate parsers.

We therefore develop techniques to look for instances of textual PI in certain HTTP fields of the observed network traffic. We assume that applications and web sites are not actively obfuscating transmitted information by hiding PI or obscuring data by using steganography-like techniques. Handling PI transmitted by actively adversarial applications (e.g., malware) introduces significant additional challenges, and we leave it to future work.
Finally, because we do not assume any privileged access to devices, we are unable to gain visibility into HTTPS traffic. While HTTPS and, more generally, any TLS/SSL encrypted traffic, represent an ever increasing fraction of Internet data, we find that a significant fraction of traffic remains in plain HTTP. For example, in our Lab traffic dataset described in Section 6.3.2, we find that 62% of the flows are HTTP, and that 44% of the ground-truth user PI is located in these HTTP flows. As a result, without the entire HTTPS traffic, we still have captured a large fraction (44%) of PI transmitted in the traffic.

6.3 Dataset description

In this section, we introduce the datasets used for characterization of PI transmission over the network and for evaluating extraction of PI from traffic. While it is ideal to collect traffic traces with ground truth PI on every user, it is unrealistic collecting such data at large scale. Instead, we use two complementary datasets: (i) small scale traffic traces with reliable ground truth collected in a controlled lab environment (Lab traffic) and (ii) large-scale traces collected from an ADSL point-of-presence (ISP traffic). The Lab traffic dataset helps us to obtain preliminary understanding on the mechanisms of PI transmission on the Internet; the ISP traffic dataset is then used to build models for PI extraction and test them at scale.

We begin this section by overviewing how we parse out PI from raw traffic, followed by detailed explanation of the Lab traffic and ISP traffic data sets, respectively.

6.3.1 HTTP parsing

From Layer-7 flows of user traffic, we extract PI of users from HTTP requests assuming that it is transferred in the form of key-value pairs. In order to properly handle these keys, we use the concept of domain-key in which we combine a key name with the name of the domain associated to the request. The intuition behind this is that key names will likely be used coherently (i.e., for carrying the same type of information) within the same domain (e.g., google.com may use the keyword “ggender” to collect user’s gender regardless the specific Google service). On the other hand, a same key might be used in the context of different domains with a different meanings (e.g., the key...
CHAPTER 6. IDENTIFYING PERSONAL INFORMATION IN INTERNET TRAFFIC

may be used different by Google and Facebook). Specifically, we extract the domain from Host HTTP header, and derive keys (and values) from three locations: (a) the query string of HTTP GET requests, (b) the query string in the Referer HTTP header, and (c) the Cookie HTTP header. For each location of potential keys, we divide the contents into key/value pairs using standard formatting rules (e.g., for GET query string parameters, we use the & character; for cookies, we use commas, semicolons, and ampersands).

The use of domain-keys (as opposed to just keys alone) allows us to capture how different domains use keys with the same name. Consider two HTTP GET messages http://loginradius.com/login?name=alice and http://ymail.com/getservice?name=send_mail. While the query string name is the same for both domains, the former domain uses it as a login ID, while the latter uses it the name of the service that the user is requesting. Thus, keeping these separate allows us to identify the former as potentially carrying PI, while the latter is unlikely to.

6.3.2 Controlled lab environment dataset

In the Lab traffic dataset, we collect traffic of a single user who intentionally transmitted a variety of known PI (i.e., ground truth) to a number of popular online services. To study contents of TLS/SSL encrypted traffic as well, we tap in to the connections with a middle-box that passes traffic through VPN/Man-In-The-Middle (MITM) transparent setup [108].

<table>
<thead>
<tr>
<th>Dataset</th>
<th>HTTP flows</th>
<th>Tuples</th>
<th>Domain-keys</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lab traffic</td>
<td>9,227</td>
<td>20,810</td>
<td>8,372</td>
</tr>
<tr>
<td>ISP traffic</td>
<td>40,775,119</td>
<td>51,368,712</td>
<td>3,113,696</td>
</tr>
</tbody>
</table>

Table 6.2: High-level statistics of our two datasets.

Using an Apple iPhone 4 with iOS 7, we ran the top 35 free iTunes applications for 15 minutes each, conducting a variety of activities including service registration, login/logout, message posting, chat message transmission, browsing etc. Detailed information about the size of this dataset is provided in Table 6.2. Given that this dataset is highly biased toward one specific user and one device, we use this dataset as an exploratory sandbox, in which we manually inspect how the known PI is transmitted in HTTP/S traffic.

6.3.3 Real ISP dataset

In order to generalize our PI extraction models with larger scale data, we conducted a large-scale traffic collection from a point-of-presence of an ISP providing ADSL service in a European city. We
CHAPTER 6. IDENTIFYING PERSONAL INFORMATION IN INTERNET TRAFFIC

use a Layer-7 traffic trace, *ISP traffic*, of 13,000 users for 24-hour in August 2011.\(^2\)

We identify over 3 million unique domain-keys and 51 million unique domain-key/value tuples (see Table 6.2 for more details). While this real-user dataset lacks ground truth information on the user PI, we evaluate correctness of our proposed method by having humans manually inspect sampled domain-keys in Section 6.6.1.3.

### 6.4 Statistical approach

We begin by measuring the overall level of PI present in traffic, and then explore whether simple statistical techniques (inspired by Section 6.2.1) might be able to identify PI leaked in large ISP traffic.

#### 6.4.1 Small scale study on controlled lab traffic

Using the *Lab traffic* dataset, we locate all of the domain-keys present in the traffic; this results in 8,372 domain-keys in total. Because we know the ground truth entered by the user, we search through the values of the domain-keys, looking for information that was provided by the user including the email address, name, city, postal code, gender, age, and geo-coordinates. We find that the fraction of domain-keys with different PI varies between 0.01% (for phone number) to 0.31% (for postal code), but overall, only 1.25% of all domain-keys ever contained any PI. Thus, we observe that locating PI in raw traffic is akin to finding needle in a haystack.

#### 6.4.2 Statistical metrics in discovering PI

Next, using *ISP traffic* dataset, we explore whether we may be able to identify PI in traffic by looking for simple statistical properties of the domain-keys. For example, perhaps values that users upload to different domains may be more likely to be PI than other values. Below, we explore each of the three properties of the domain-keys in the *ISP traffic* data set based on the taxonomy presented in Section 6.2.1. Then, we analyze the effectiveness of combining the statistical metrics in discovering the PI leaks.

**Static vs. Dynamic** Overall, we find that there are 341,179 (11.0%) static domain-keys (i.e., every user has only one value for the domain-key) and 111,664 (3.6%) dynamic domain-keys (i.e.,

\(^2\)Per privacy policy of the ISP, we anonymized true identities of the users prior to this work.
every user has two or more values); the remaining are mixture of both (i.e., some users have single value, some have multiples). This is unsurprising as the vast majority of domain-keys have very few values. Manually inspecting the static and dynamic domain-keys reveals a few candidates for PI, but the majority of domain-keys have no obvious semantic meaning for PI. Thus, while this approach does identify domain-keys that are static or dynamic for users, it is still not precise enough to be useful for pinpointing PI.

![Figure 6.1: The cumulative distribution of the uniqueness ratio of all static domain-keys in ISP traffic dataset.](image)

**Unique vs. Non-unique** The second feature we explore is whether each user is mapped to a unique value. We focus on the static domain-keys discovered above in order to use domain-keys that are likely to map a single value to each user (i.e., examining uniqueness of dynamic values requires careful consideration of the number of online users, etc). We define a new metric *uniqueness ratio* for a domain-key, which is the number of unique values divided by the number of users for the domain-key. We show the cumulative distribution of the *uniqueness ratio* in Figure 6.1. Among all the static domain-keys, 96,375 (28.24%) of them have the ratio 1.

While the static domain-keys with *uniqueness ratio* of 1 are more likely to contain the static type of PI, such as user’s username, and email address, the majority of them are comprised of session identifiers, GUIDs, and the like. Thus, relying on uniqueness alone is also likely to produce too many false positives to be useful.

**Shared vs. Distinct** Data a user is sharing across domains suggests that the value may correspond to the user’s PI (e.g., the same email address used as login account for different web sites). Out of 26,453,858 unique user-values, we find 5,923,084 (22.4%) of them have been sent to multiple domains; we show the cumulative distribution of the number of domains for each user-value in Figure 6.2. There, we find over 7% of user-values have been sent to at least 10 domains.
Among the values sent to multiple domains (i.e., distinct data), we find some meaningful PI (e.g., we find tracking user identifiers that are used across domains). However, the majority of them are common values with no implications on PI such as 0, 1, true, false, etc (e.g., a user sent the value of 1 to more than 100 different domains). Thus, as before, looking only at the values that are shared across domains is simply not precise enough to effectively locate PI.

**Combining features** We have observed so far that looking for individual features of domain-keys is not precise enough to locate PI. We now briefly explore combining statistical features together, looking for domain-keys that are static, unique to a user, and shared across domains. Our selection of these features is to capture PI such as email addresses, which may be used as login information for multiple web sites. This combined criteria leaves with a small set of 262 pairs of domain-keys that are unique, static, and have at least 20 users that share the same value in both domain-keys.

To evaluate whether these domain-keys contain real user PI, several human raters\(^3\) manually inspected each of the domain-keys. Overall, we find that only 33 (12.6%) of the domain-keys were labeled by humans as PI; examples include email addresses, ids, user interest, locations, etc. Taking a closer look at the false positives, we find that they contain values that are related to the users’ activities, however, not users’ sensitive PI, including dates, timestamps, click tags, referrers, etc.

**Summary** Applying individual statistical tests in the real ISP traffic results in too many false positives; applying a combination of the tests results in too few true positives. In the following section, we propose a more sophisticated technique based on the properties of values of different domain-keys we learned so far.

\(^3\)To further protect privacy of users from de-anonimization, the only human participants allowed to review the data was limited to the six authors of the research paper [93].
CHAPTER 6. IDENTIFYING PERSONAL INFORMATION IN INTERNET TRAFFIC

6.5 Seeded approach

This section describes the method we propose to identify PI in network traffic traces, parsing traffic data and distinguishing user PI in an automated fashion. We begin by extracting fields from the various HTTP headers in the manner described in Section 6.3.1.

Figure 6.3: Overview of our proposed approach, exemplifying the four steps in identifying domain-keys that may contain PI. In the Pre-processing step, domain-keys with too few values are filtered. In the Seed rules step, different rules are applied to the values in each domain-key; domain-keys with too few matching values are filtered out in the Filtering stage. Finally, in the Expansion step, the newly discovered values are shown in **bold**, and these may be used to help refine the seed rules.

As observed in the previous section, in the vast majority of cases, reliance on the statistics of domain-keys fails to reveal values with PI. Hence we shift our focus to the domain-key values and propose a novel semantically-based method which we refer to as the **seeded method**. We briefly describe our high-level approach here, and provide more details on each step. First, we have an initial pre-processing step, where we examine all domain-keys of a dataset and filter out those that do not have enough values to produce statistically meaningful results. Second, we apply a number of **seed rules** crafted to find clues of PI directly from the values contained in each domain-key. Using these rules, we select candidate domain-keys to be those that have a sufficient level of matches. Fourth, we extend the set of possible values to include those in the candidate domain-keys by adding the missing values into our value pool. A diagram of these four steps is presented in Figure 6.3.

These four steps are described in more detail in the following subsections. Many of the steps require choices of constants and parameters; when describing each of the steps, we describe our process for userpi-r selecting the parameters based on observations from the **Lab traffic** and **ISP traffic** data sets.
6.5.1 Pre-processing

Our approach relies on the format of the values of different domain-keys to select domain-keys that are likely to be carrying PI. Thus, we need a large enough sample of values to be able to produce statistically significant results. To do so, we simply select a threshold \( n \), and only consider domain-keys for which we have observed \( n \) tuples (user/value pairs). For example, \( n = 5 \) can either mean one unique value from each of 5 different users, or 5 different values from a single user.

![Figure 6.4: The cumulative distribution of the number of unique tuples, users, and distinct values for each domain-key discovered in ISP traffic dataset.](image)

When applying this pre-processing step, we naturally face a tradeoff between the potential false positives and the coverage of domain-keys where we have few data points. Thus, we briefly explore the coverage of domain-keys that different choices of \( n \) provide. Using the ISP traffic data set, we plot the cumulative distribution of the number of distinct values each domain-key has (we also plot the number of users and total number of tuples for comparison) in Figure 6.4. We observe that out of the 3.1M total domain-keys, only the top 270,756 (8.7%) “heavy hitter” domain-keys have at least 10 distinct tuples. However, these heavy hitter domain-keys in aggregate cover 90.8% of all observed tuples; thus, when applying pre-processing, we filter out a significant fraction of the domain-keys, but still retain the vast majority of the observed tuples in the trace.

6.5.2 Seed rules

We develop a list of constraints, or seed rules, based on the format of expected PI. For many of the different types of PI, seed rules can be expressed as simple regular expressions, and are sufficient to express the possible data formats. For example, in Table 6.3, regular expressions are sufficient to capture email addresses, genders, age ranges, geo-coordinates, postal codes, and phone numbers.\(^4\)

\(^4\)We note that our examples of phone numbers and postal codes use the local formats of the country where our data set is from.
**CHAPTER 6. IDENTIFYING PERSONAL INFORMATION IN INTERNET TRAFFIC**

<table>
<thead>
<tr>
<th>PI type</th>
<th>Regular expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age Range</td>
<td>/^[0-9]{1,3}-^[0-9]{1,3}$/ (where the second number is larger than the first)</td>
</tr>
<tr>
<td>Email</td>
<td>/^[w-</td>
</tr>
<tr>
<td>Geo</td>
<td>/^[0-9]+^[0-9]+^[0-9]+$/ (where the value is within the range of the country)</td>
</tr>
<tr>
<td>Gender</td>
<td>/^[mf]$ or /^[fe]?male$/ (or the corresponding words for male/female in local language)</td>
</tr>
<tr>
<td>Phone</td>
<td>/^[+^[089]{8,9}</td>
</tr>
<tr>
<td>Postal code</td>
<td>/^[d{5}]/</td>
</tr>
</tbody>
</table>

Table 6.3: Examples of regular expressions used for a subset of the seed rules. Some of the regular expressions require minor post-processing, such as the “age range” PI category, where the second element of the range must be greater than the first.

For some of these, some simple post-filtering is required (e.g., to express that an age range is from a lower number to a higher one).

However, other types of PI may not be as easily expressible as a regular expression. Examples of such PI include user’s names, cities, and regions. To capture these, we also allow seed rules to be expressed as dictionaries containing lists of possible values. For example, for first names of users, we create a comprehensive list of names by downloading a set of corresponding web pages with boys or girls’ names from the given country. Similarly, we create a dictionary of different cities and regions in the country of interest in order to create a seed rule for the user’s location.

Our dictionary-based rules do not need to be exhaustive to be effective. As we show in the next section, as long as our seed rules are sufficient to cover a significant fraction of the actual values (in practice, we have found good performance with as low as 20% coverage), our Expansion step is able to discover the additional values as potential PI.

Lastly, we note that we limit our seed rules to the above eight PI instances simply for brevity, not because the expressiveness of the rules is limited to just these types of PI. We believe these exemplary rules are sufficient for demonstrating both the utility of our proposed method and the applicability to various other types of PI.

Of course, the seed rules that we have selected are unlikely to cover all the cases, formats, and languages; they can easily be improved and expanded, based on the input and results. Though the seed rule solution is not universally applicable, for example in the Table 3 “where the value is within the range of the country”, we need to apply different things in the seed rules based on the input of dataset. However, once the seed rules are generated, they can help us in discovering domain-keys with different types of user PI in an efficient and automatic way.

[^5]: http://www.babynamespedia.com/search/m/countryname
6.5.3 Filtering domain-keys

Not all the domain-keys matching seed rules represent PI. To confirm that the domain-keys is indeed likely used as a container for transmitting PI, for each domain-key, we look at all its values, and we compute a metric \textit{ratio matched values} in relation to each seed rule. This metric is simply number of values that match the seed rule, divided by the total number of values. When the \textit{ratio matched values} is above a given threshold as described below, we consider that this domain-key is likely to carry the type of PI with a confidence represented by the matching value ratio.

![CDF of Ratio](image)

Figure 6.5: The cumulative distribution of \textit{ratio matched values} for domain-keys in ISP traffic dataset for each of the eight different seed rules. For each rule, only those domain-keys that have at least one matching value are considered. Different rules show different properties, but most show a “knee” in the curve at some threshold.

Again, choosing an appropriate \textit{ratio matched values} is a tradeoff, where a small threshold \textit{ratio matched values} increases the coverage, but results in higher false positive rates. To illustrate how we choose the threshold in practice, Figure 6.5 plots the distribution of \textit{ratio matched values} across domain-keys in the ISP traffic dataset for each of the eight seed rules. For each rule, only those domain-keys that have \textit{at least one} matching value are considered.

We make a number of interesting observations. First, we observe that while the distribution is different for each of the seed rules, all of the rules show a “knee” at some point in the curve. In the case of email addresses, for example, 21% of domain-keys have all their values matching the rule (i.e., the ratio is 1) and over half (58%) of the domain-keys have at least 20% of their values matching the seed rule. In contrast, the name domain-key shows that the vast majority (over 90%) of domain-keys that have at least one matching value have less than 10% of all their values match. This difference in performance is due to the nature of the seed rules; for email, the regular expression is unlikely to select values that are not actually email addresses, whereas the name seed rule is only a subset of the possible user names.
CHAPTER 6. IDENTIFYING PERSONAL INFORMATION IN INTERNET TRAFFIC

To choose a good tradeoff between false positives and coverage, for each rule, we choose a *ratio matched values* threshold to the knee points of the corresponding distribution. For example, we select a threshold to be 1 for postal codes, 0.9 for geo locations, and 0.2 for the rest.

6.5.4 Expanding candidate domain-keys

We understand our seed rules are not exhaustive, and it is challenging to develop perfect seed rules that can match all the possible formats of PI. To address this limitation, we expand the candidate values associated with selected domain-keys into our value pool, to compliment our findings. For example, if we pick 0.2 for the email case shown in Figure 6.4, we get a list of domain-keys with email, and each of them have at least 20% of their values matched the seed rule. When we consider the set of PI, we add the values that do not match the rule into our value pool as well. Though they do not match the exact regular expression, they may also contain user PI (e.g., the username without a domain, an email address with extra whitespace, etc).

<table>
<thead>
<tr>
<th>Domain</th>
<th>Key</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>google-analytics.com</td>
<td>email</td>
<td><a href="mailto:johnDoe@gmail.com">johnDoe@gmail.com</a></td>
</tr>
<tr>
<td>google-analytics.com</td>
<td>email</td>
<td><a href="mailto:janeDoe@hotmail.com">janeDoe@hotmail.com</a></td>
</tr>
<tr>
<td>google-analytics.com</td>
<td>email</td>
<td>johnDoe</td>
</tr>
<tr>
<td>google-analytics.com</td>
<td>email</td>
<td>janeDoe</td>
</tr>
<tr>
<td>facebook.com</td>
<td>gender</td>
<td>female</td>
</tr>
<tr>
<td>facebook.com</td>
<td>gender</td>
<td>m</td>
</tr>
<tr>
<td>facebook.com</td>
<td>gender</td>
<td>f</td>
</tr>
<tr>
<td>facebook.com</td>
<td>gender</td>
<td>l</td>
</tr>
<tr>
<td>facebook.com</td>
<td>gender</td>
<td>f-f</td>
</tr>
<tr>
<td>facebook.com</td>
<td>gender</td>
<td>f-m</td>
</tr>
</tbody>
</table>

Table 6.4: Examples of values that both match and do not match the seed rules. We observe that values that do not match the seed rules may still identify potential leaks, and may help to refine the seed rules.

A few examples of matching and non-matching values are presented in Table 6.4. The first four rows show values associated with domain-key `<google-analytics.com, email>`. While the first two examples match the regular expression, next two rows with values “johnDoe” and “janeDoe” are truncated form of the first two email addresses used as usernames. The last six rows show values associated with the domain-key `<facebook.com, gender>`. We observe that the first three match the seed rule, but the final three do not (although they likely contain some form of PI being conveyed by Facebook or a third-party application). In both cases, considering the values that do not match the seed rules can help to refine the seed rules, as well as present additional potential leaks to the administrator.
We note that there are a few reasons why a service would use different formats for the same key. First, the service may have different formats of values within different parts of HTTP header, such as url, referrer, and cookie. Second, some services provide APIs for the external developers, who may use formats that differ from the main service. For example, we notice that Facebook advertising API allows us to specify the key “gender” with value of 0 to target male users and value of 1 for female users. Instead, the main Facebook service uses a list of other values, such as male, female, m, and f to specify gender. Third, the values may be based on end user input, which is not always well-formatted.

6.6 Evaluation

In this section, we apply our proposed approach to ISP traffic and evaluate its performance in comparison to a baseline approach. We then present interesting findings on user PI through an in-depth analysis on the discovered domain-keys.

6.6.1 Evaluation of seeded method

Having applied the seeded method to our datasets, we now analyze the performance of the method in terms of coverage and accuracy. To evaluate our proposed method in controlled environment with ground truth, in Section 6.6.1.1, we begin by analyzing results from small-scale Lab traffic data set (Sec. 6.3.2). Then in Sections 6.6.1.2 through 6.6.1.4, we evaluate the method on full-scale ISP traffic data set (Sec. 6.3.3). Here, we first quantify coverage of our seeded method by comparing it to a baseline naïve method. We then further analyze the accuracy of our results by manually inspecting correctness on a sample of the discovered values.

<table>
<thead>
<tr>
<th>PI Type</th>
<th>Dom-keys selected</th>
<th>Ground truth</th>
<th>Coverage</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age Range</td>
<td>0</td>
<td>0</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>City</td>
<td>19</td>
<td>19</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>Email</td>
<td>22</td>
<td>22</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>Geo</td>
<td>34</td>
<td>7</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>Gender</td>
<td>38</td>
<td>14</td>
<td>100%</td>
<td>36.8%</td>
</tr>
<tr>
<td>Name</td>
<td>16</td>
<td>16</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>Phone</td>
<td>1</td>
<td>1</td>
<td>100%</td>
<td>100%</td>
</tr>
<tr>
<td>Post code</td>
<td>144</td>
<td>26</td>
<td>100%</td>
<td>18.1%</td>
</tr>
<tr>
<td>Total</td>
<td>274</td>
<td>105</td>
<td>100%</td>
<td>38.3%</td>
</tr>
</tbody>
</table>

Table 6.5: Comparison of domain-keys selected by the seeded method and ground truth in Lab traffic dataset.
CHAPTER 6. IDENTIFYING PERSONAL INFORMATION IN INTERNET TRAFFIC

6.6.1.1 Verification on Lab traffic dataset

Using the ground truth on PI we have in Lab traffic dataset, we measure the validity of domain-keys and their values obtained using the seeded method. Out of 20,810 tuples from 8,372 domain-keys available in the HTTP/S data, seeded rules extract 274 domain-keys as containing PI. A breakdown of domain-keys discovered by the rules, and those containing the ground truth PI leaks, is detailed in Table 6.5.

We make a number of interesting observations from our inspection of the results: overall, only 3.27% of all domain-keys are discovered as containing PI. Given the “needle in a haystack” observation we made in Section 6.4.1, it is reasonable that the selected domain-keys are only a small fraction of all domain-keys.

In a few particular cases, we find a single rule matching multiple instances (different values) of PI. For example, a domain-key <cm.g.doubleclick.net, ct> matching the City rule has a list of different values: Boston, Beijing, Seoul, Shanghai. While we consider Boston as the most “correct” answer (as it is the current residential city of the user), the rule found other cities that she visited in the past. At the same time, the existence of multiple PI instances supports the utility of the candidate value expansion in Section 6.5.4 as it accommodates broader range of the candidate values into user PI pool.

Occasionally, we found the expansion stage of the approach to introduce a few false negative values. For example, we find an extracted domain-key <graph.facebook.com, name> containing two application names, “Pinterest” and “iHeartRadio”, along with the target user name “Yabing Liu” (one of the authors). The application names were found in the domain-key because the user was logging on to the applications through her Facebook account.

Overall, our evaluation on the Lab traffic data set suggests that our approach is able to identify domain-keys that carry PI with high coverage. Moreover, we observed a significant fraction of these (44%) were observed in unencrypted HTTP traffic, including the user’s name, gender, city, and postal code. We now explore running our approach on the much larger ISP traffic data set.

6.6.1.2 Improvement over a baseline approach

From this section on, we evaluate our method on a larger, realistic dataset of ISP traffic. To understand the improvement of our seeded method over baseline results, we begin this section by running a comparative study of our seeded method against a naïve, key-semantic based approach that analyzes key names (as opposed to values as in our seeded approach). Then we provide our reasoning
CHAPTER 6. IDENTIFYING PERSONAL INFORMATION IN INTERNET TRAFFIC

<table>
<thead>
<tr>
<th>Type</th>
<th>Key semantics</th>
<th>True positive ex.</th>
<th>False positive ex.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age Range</td>
<td>substring of age</td>
<td>age, age_range,</td>
<td>message, language, pagesize</td>
</tr>
<tr>
<td></td>
<td></td>
<td>city, location</td>
<td>client_state, locale</td>
</tr>
<tr>
<td>City</td>
<td>substr. of city, citta, area, state, provincia, loc, region, where</td>
<td>email, login_email</td>
<td>user_segment, login_password</td>
</tr>
<tr>
<td>Email</td>
<td>substr. of email, user, account, login, logon, or equal to “e”</td>
<td>gender, user, sex</td>
<td>pagename, useragent</td>
</tr>
<tr>
<td>Gender</td>
<td>substr. of gen, gnd, gdr, ycg, sex, or equal to “g”</td>
<td>latitude, logitude</td>
<td>platform, relation</td>
</tr>
<tr>
<td>Geo(Lat/Lon)</td>
<td>substr. of lat, lon, lng, geo</td>
<td>name, person</td>
<td>app_name, slotname, listname</td>
</tr>
<tr>
<td>Name</td>
<td>substr. of name, nome, pers, author</td>
<td>phone, pid</td>
<td>appid, sdkapid</td>
</tr>
<tr>
<td>Phone</td>
<td>substr. of phone, pid, or equal to “p”</td>
<td>zipcode, geo</td>
<td>gzip, gzipbyteencoding</td>
</tr>
<tr>
<td>Postal Code</td>
<td>substr. of zip, geo</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6.6: Examples of both true and false positives when using the key semantic method.

on why seeded method constantly outperforms the baseline method without even considering the semantics of the keys.

Baseline key-semantics approach We create a strawman approach based on key semantics in which we leverage common intuition that keys that are suggestive of PI (e.g., keys named “email”, etc) would carry the PI as their values. In other words, if the majority of keys containing PI have dictionary words such as “email”, “gender”, or “name”, the baseline approach should be able to collect all such domain-keys with PI. We later compare the results of our seeded approach against that of the simple baseline approach, and quantify their gap in terms of coverage.

To create the baseline approach, we select a list of lexicons of each PI category. Table 6.6 presents the selected key terms for each of the eight PI categories we consider, along with some examples of true and false positives. Overall, we find 20,565 of our domain-keys match at least one of the rules (a breakdown is shown in Table 6.7, column six).

Performance comparison. Table 6.7 presents a detailed comparison of the coverage of our seeded method to the coverage of the key-semantic method. In particular, comparing columns four and six shows the total number of domain-keys selected by the two methods in each category, respectively. We immediately observe that the key-semantic method finds many more potential domain-keys containing PI (in some cases, up to three orders-of-magnitude more). However, this result may be somewhat misleading, as these are only potential domain-keys that may or may not carry PI (e.g., the domain-key <facebook.com, function-name> would be selected by the key-semantic method, as it has name in the key name). Thus, to fairly compare the two methods, we need to estimate their false positive rate.
CHAPTER 6. IDENTIFYING PERSONAL INFORMATION IN INTERNET TRAFFIC

Table 6.7: Comparison between our proposed seeded method and baseline key-semantic method. The seeded method has a dramatically lower false positive rate (13.6%, when disregarding postal code) than the baseline method.

<table>
<thead>
<tr>
<th>PI Type</th>
<th>Seeded # DKs above threshold</th>
<th>Key-semantic # DKs positives</th>
<th>Comparison Common Unique to Seeded Key-semantic Unique to Key-semantic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age Range</td>
<td>199</td>
<td>0.0%</td>
<td>3,729</td>
</tr>
<tr>
<td>City</td>
<td>1,402</td>
<td>8.8%</td>
<td>3,191</td>
</tr>
<tr>
<td>Email</td>
<td>382</td>
<td>3.9%</td>
<td>3,253</td>
</tr>
<tr>
<td>Gender</td>
<td>2,041</td>
<td>0.0%</td>
<td>1,358</td>
</tr>
<tr>
<td>Geo (lat/lon)</td>
<td>341</td>
<td>10.0%</td>
<td>1,986</td>
</tr>
<tr>
<td>Name</td>
<td>1,549</td>
<td>52.5%</td>
<td>2,142</td>
</tr>
<tr>
<td>Phone</td>
<td>993</td>
<td>90.9%</td>
<td>3,864</td>
</tr>
<tr>
<td>Postal Code</td>
<td>13,449</td>
<td>100%</td>
<td>1,044</td>
</tr>
<tr>
<td>Total</td>
<td>20,356</td>
<td>2,789 13.6% (25.3%)</td>
<td>20,565</td>
</tr>
</tbody>
</table>

6.6.1.3 Accuracy analysis on samples of results

As we do not have the ground truth PI of the ISP traffic dataset users (i.e., customers of the ISP), we instead rely on multiple human raters to identify potential PI. Due to the size of the data set, we use sampling to make the evaluation tractable.

We first describe how we evaluate the accuracy of our seeded method. To evaluate the false positives, we began by choosing up to 170 random domain-keys from each of the eight PI category from the final output of the seeded method, hence choosing 873 flagged domain-keys in total (31.3% sampling rate out of the final 2,789 domain-keys). Similarly, to measure false negatives, we randomly sampled 1,000 domain-keys that the seeded method did not chose (0.032% sampling rate out of 3,110,907 non-flagged domain-keys). To measure the accuracy of the key-semantic method, we take a similar approach. We select up to 25 random domain-keys from the domain-keys identified by the key-semantic method in each category of PI.

For each of the domain-keys tested, the six human raters either labeled positive (i.e., the type of the PI), negative, and neutral (i.e., don’t know) for the question of whether the domain-keys contain PI or not. In all tests, for each domain-key, we then chose 10 values randomly to present to the human rater, alongside the domain and key name. Each domain-key was reviewed by three raters, allowing us to run majority voting when labels disagreed.

Seeded method Overall, we find that 221 out of the 873 domain-keys flagged by the seeded method to be false positives (i.e., the human raters indicated not containing PI), resulting in a false positive
rate of 25.3% (with the corresponding confidence interval from 22.4% to 28.2%). However, we notice that the false positive rate for postal code is as high as 91.6%, which means our seeded rule does not work well in identifying it. As detailed in Section 6.5.2, a seed rule generalizes particular patterns embedded in PI. In the case of postal codes, the seed rule of `/\d{5}$/` is not specific enough to separate the PI from random five digit numbers. For this reason, we filter out postal code from our ruleset, and obtain a resulting false positive rate of 13.6% (with the corresponding confidence interval from 11.3% to 15.9%).

We also find that 27 out of the 1,000 non-flagged domain-keys to be identified by the human raters as containing PI, thereby representing a false negative rate of 2.7% (with the corresponding confidence interval from 1.7% to 3.7%).

**Key-semantic method**  For the key-semantic method, we observe that the human raters found 179 of the 200 domain-keys flagged by the key-semantic to be false positives, resulting in an extremely high false positive rate of 89.5%.

Overall, the survey finds the false positive rate to be high for the seeded method, and unacceptably high for the key-semantic method. However, we believe that the 13.6% false positive rate of our seeded method is acceptable due to three reasons: First, PI is rare, and it is difficult to find the correct PI from a huge dataset without any ground truth. Second, the false positive rate is tunable by selecting a different threshold; we opted for increased coverage in these experiments, and could easily lower our false positive rate at a cost of increased false negatives (currently 2.7%). Third, we observe that it is difficult even for humans to agree what is PI and what is not. For example, among the 873 labeled domain-keys from seeded method, only on 81% of them did the human raters agree: on 18% one rater disagreed, and 1%, all disagreed. The upshot is that our method is able to focus quickly on the small subset of domain-keys that potentially leak PI.

### 6.6.1.4 Exploring higher accuracy of the seeded method.

While the seeded method only focuses on the *syntax of values* (via regular expressions and dictionaries), it captures many more DKs with PI than the baseline approach focusing on the *semantics of keys*. To better understand the reason for the large difference, we take an in-depth look at the key semantics of the domain-keys found by the seeded method.

We first analyze domain-keys the key-semantic method selected but our seeded method did not. Column 8 of the Table 6.7 shows the number of domain-keys overlapping between the two methods.
CHAPTER 6. IDENTIFYING PERSONAL INFORMATION IN INTERNET TRAFFIC

Compared to column 10 (i.e., total number of domain-keys selected by the key-semantic method), we observe that only a small fraction (3.1%) of the domain-keys selected by the key-semantic method are indeed included in the final results of the seeded method, suggesting that the vast majority of services do not name their keys semantically accurately. For instance, a key term “name” does not always draw terms relevant to user names we target. Instead, as exemplified in Table 6.6, it erroneously includes mobile app names, names of data slot, and a binary representation of existence of a name.

We then analyze domain-keys the seeded method selected but the key-semantic method did not. As the small difference between column 4 and column 8 of Table 6.7 suggests, the majority of DKs seeded method selects are included in the selection of key-semantic method as well.

Figure 6.6: CDF of the ratio of values matched for email domain-keys discovered via seed rule.

From the total of 20,356 domain-keys that match seed rules, the “ratio” thresholds we impose in Section 6.5.3 selects only 2,789 of them (13%) as the rest do not contain enough number of valid values. Figure 6.6 further explains this using the email category as an example. The curve shows the cumulative distribution of email domain-keys ordered by the fraction (ratio) of values matching our seed rule. Out of the total of 382 domain-keys that match seed rules, the seeded method selects 154 of them after imposing the preset threshold ratio of 0.2. Upon our manual inspection on the 228 domain-keys that were left out, many of them contained values irrelevant to emails. For example, in domain-key `<static.ak.connect.facebook.com.email>`, binary tags of 0 and 1 are used for its value, possibly encoding the existence of emails. In domain-key `<adserver.adtech.de.city>`, indexes of cities are used, which we are unable to decipher without knowing its indexing mechanism.

In summary, we observed important shortcomings of the baseline key-semantic method to be applicable for automatic discovery of PI: sensitivity to selection of input domain-key categories.
and key terms, inability to discern domain-keys containing irrelevant values, and overly high false positives due to limited expressiveness of key terms. Our proposed seeded method, on the other hand, is deemed to be much more robust to the above issues.

6.6.2 Analysis on services leaking PI

Using the results of our seeded method, we now analyze the 2,789 domain-keys that contain user PI. In particular, we aim to answer the following questions: (i) are there any specific types of user PI collected by particular kind of services and (ii) verify the existence of abusive domains that collect a broad range of user PI.

To this end, we focus on six types of PI: age range, city, email, gender, geo location, and name. For each root domain labeled as positive by our seeded method, we assign one of the following eleven service categories by manual inspection: Advertisement (e.g., ads.bluelithium.com), Adware\(^6\) (citibank.0009.ws), CDN (img-cdn.mediaplex.com), User-tracking (pixel.quantserve.com), along with more familiar service categories of Game, OSN, Search-engines, Web-portals, and Adult. As an example of a domain with multiple services, we add Google in the category.

Figure 6.7 shows a heat map based on the probability density distribution of PI types by category. Comparing popularity of PI used across services, we notice that residential city turns out to be the most prevalently leaked PI (30.7%) followed by fine-grained geo-coordinates (15.2%) and gender information (10%).

\(^6\)Different from advertisement services, we categorize adware as domains known to distribute undesired ads by means of phishing or malware.
PI leak per service category. With respect to question (i), we analyze the service categories that leak PI. CDNs highly benefit from spatial locality of cached data to users. Therefore, knowing user location is one of their primary interests. As shown in Figure 6.7, there is a high correlation between CDN and city. Similarly, for search engines, portals, and ad services, to which providing local information to users is also important, we observe high correlation to city and geo-location.

In contrary to the majority of the Internet service categories that exhibit high correlation to location information, OSNs show very low correlation to city and geo-location; they have comparatively high correlation with emails, gender, and age range. We speculate that this is due to the online-nature of the OSNs which weighs more on the knowledge of age and gender groups rather than physical locations.

In the case of adult services, knowledge on the age range and gender of users is deemed to be important as they may provide age-restricted, gender focused contents. From tracking category, we notice that some user-tracking web bugs, which are supposedly used for aggregated web analytics, can track and identify users by their email addresses.

PI leak per service domain. With respect to question (ii), we analyze the types of user PI leaked by different domains and identify domains more prone to leak user PI. Out of 588 different domains, we find 489 (83.1%) of them only have one type of PI leaked. 79 (13.4%) of them have two types of PI leaked, and overall, 20 (3.4%) of the domains have more than two types of PI. Interestingly, one domain in user-tracking category collects five types of PI. A cursory investigation reveals that this domain is identified by users as an invasive service and sometimes associated to spyware. From our traffic trace, we confirm similar suspicious behavior, as we observe the domain collecting email, name, gender, city, and location-related information, such as geographical coordinates. Among other examples of domains collecting above average amount of PI, we find large Internet service companies with ad services such as Google and Yahoo!: google.com contains an average of 5 PI types per user, google-analytics.com with 4, doubleclick.net with 6, and yahoo.com with 5.

6.7 Impact and limitations

Selecting thresholds The intervention of an analyst performing manual inspection on the PI identified by the methodology might be required in order to optimally set the thresholds at the basis of the operations of the proposed method. One of our future work directions will focus on automating threshold setting. For example, a “state space exploration” of the potential settings could
be conducted to then pick the ones that perform best on a number of tests. This is not trivial as it might require a large amount of computation resources and carries the risk of overfitting. In the meantime, the need for having a human in the loop does not undermine the high value of the approach since it brings to the analyst attention only a small fraction of the large quantity of information flowing through the Internet. Without the support of this method, manually inspecting the full traffic has widely proven unfeasible.

**Encrypted traffic** Since the methodology here presented relies on inspection of data exchanged by Internet services its applicability can be limited when such data is encrypted or obfuscated. While the latter is a complexity that most service providers currently do not want to incur, in the last years the fraction of web traffic being encrypted (i.e., using HTTPS) has increased significantly. This does not undermine the relevance of the solution as it has several application areas of significant impact where the effectiveness of the methodology is not affected by the deployment of HTTPS.

**PI leakage protection** A service provider (being it an Internet, cloud, or cellular service provider) could offer to its customers a service to audit their traffic for PI that is potentially collected by third parties because it is being sent through the network in clear text form. This application does not require visibility into PI sent over encrypted connections as it does not represent a leakage in this context.

**Enterprise protection against information leakage** The proposed approach can be deployed by a company wanting to detect intentional and unintentional leakage of information critical to their business, which includes employees’ PI. In this scenario the company will enforce all HTTP(S) traffic to go through a corporate (man-in-the-middle) proxy that terminates SSL sessions, thus acquiring visibility into encrypted traffic (in fact, there exist companies that use such proxies today). Such an approach requires applications (e.g., web browsers) to accept as legitimate the certificates that the proxy generates, signs with its own certificate, and presents in the initial SSL negotiation phase. This can be achieved by pre-loading the proxy certificate into corporate PCs as the certificate of a trusted certification authority. Employees that want to use their own devices to access the Internet through the corporate network are required to install the proxy certificate or manually accept the certificates offered when SSL sessions to new servers are negotiated.

**PI disclosure assessment and control** A provider could offer a service that identifies PI being embedded in the traffic of a user, both protected (i.e., through HTTPS) and unprotected. As
CHAPTER 6. IDENTIFYING PERSONAL INFORMATION IN INTERNET TRAFFIC

it is common in many other contexts, such as online social networks, free e-mail services, etc., customers interested in the service are willing to grant the provider with visibility into their encrypted traffic. This can be achieved by the user either loading the certificate of a man-in-the-middle proxy operated by the service provider in the trusted certification authority repository, or installing a module (e.g., a browser plugin) that analyses the traffic before being encrypted [98]. Our technique can then be applied within the proxy or the plugin.

6.8 Summary

In this chapter, we proposed a method for automatically identifying user PI embedded in network traffic to online services, including OSNs and non-OSNs. The proposed method first identifies a limited number of “containers” of user PI (called domain-keys) by detecting limited instances of eight types of seed PI. Then, coverage is extended by inferring additional containers by analogy with the seeded ones.

We evaluated our approach on a network dataset collected from a point-of-presence of a European ISP, covering 13,000 real users. As we do not know the ground truth PI for these users, we establish ground truth by relying on multiple human raters to label domain-keys that contain PI. In § 6.6.1, we find that our approach is able to identify these rare domain-keys automatically, with a low false negative rate (2.7%) and an acceptable false positive rate (13.6%).

We then applied our approach to the entire data set in § 6.6.2, exploring the frequency with which web sites and applications transmit PI in practice. There, we discover that different types of Internet service focus on different PI (e.g., , CDNs tend to leak physical locations of users while adult services leak age and gender information). We also find that an invasive user-tracking service leaks higher amount of PI than others.

As our final chapter of analysis in privacy implications of third parties, which has changed with online services, we examined what kinds of PI were collected and transmitted over the Internet network. This work can help develop approaches to prevent privacy leaks, and compliment related studies which have applied different methodologies. In the next chapter, we will conclude the thesis and discuss the future work.
Chapter 7

Conclusion

In recent years, online systems such as OSNs have exploded in popularity, with millions and even billions of users to connect, socialize and share content. The growing amount of shared content on OSN sites arises emerging privacy implications for end users. The popularity of online systems and the significant privacy concerns inspired the work in this thesis. We designed, deployed new methodologies to examine privacy implications by focusing separately on each of the three entities that are involved in online services: end users, site operators, and third parties. In the following sections, we describe the high-level contributions of this thesis and discuss potential future research directions.

7.1 Summary

In this thesis, we made important contributions where we developed novel methodologies to measure and analyze the data sharing activities and the privacy implications of such behavior in today’s online systems from the roles of three entities: end users, site operators, and third parties in online systems, as well as to explore new tools and systems to improve privacy for end users.

First, as end users need to manage the privacy settings for each piece of uploaded content, the content access control is becoming a significant mental burden for many users. While previous studies surveyed the users’ awareness, attitudes, and privacy concerns, we conducted the first large-scale measurement study which can quantify the magnitude of the problem of managing privacy on Facebook. We designed a user survey, implemented as a Facebook application, to select content to ask the users about, and compare the existing privacy settings with users’ expectations. We found that the privacy settings match users’ expectations only 37% of the time, and users have troubles in
CHAPTER 7. CONCLUSION

managing their content. Using insights from this measurement study, we proposed using community
detection techniques to assist users in selecting appropriate privacy settings. In detail, we presented
the design, implementation, and deployment of Friendlist Manager, a Facebook application that
leverages communities to provide meaningful friend lists for users, allowing users to share content
with particular subsets of friends, especially for privacy sensitive contents.

Second, the new model of OSN systems opens up new opportunities for site operators to monetize
user data by providing site-specific targeted ad services to third parties (e.g., advertisers). We
demonstrated how to use the suggested bid, a common mechanism on OSNs to explore the user value
with different demographics and the stability of the ad market. Through the exploration of suggested
bid data for different demographics, we find dramatic differences in prices paid across different
user interests and locations. And the Facebook ad market shows long-term variability, suggesting
that Facebook ad services have yet to mature. This work provides a first mechanism to quantify
the relative value of users with different demographics on OSNs. Since suggested bid is a common
feature used by OSN sites, as more OSNs develop ad markets, our approach can be used to measure
other markets as well.

Finally, users often share all kinds of personal information on the online systems, but with
little regard for which will be collected and transmitted to online systems, including OSNs and
non-OSNs, such as some mobile applications. Most previous studies proposed methodologies under
the assumption that the ground truth about user data is given. To deal with the cases when the ground
truth is not available, we presented a semantically-based seeded method which can identify a limited
number of “containers” of PI (called domain-keys). Our evaluation on a large-scale traffic trace
collected on the network of a residential service provider shows that our proposed approach can locate
the rare domain-keys that serve as containers for PI with low false-negatives (2.7%) and acceptable
false positives (13.6%). Our seeded approach is novel in that it can detect user PI embedded in
massive amounts of network flows, without any ground truth information, and can be performed at
scale.

7.2 Future work

So far, we have developed novel methodologies to explore privacy implications from different
perspectives of three entities in the online systems: end users, site operators, and third parties. The
continuing popularity of OSN sites and newly developed applications, which are designed not just for
desktop but also mobile devices, underscore a trend that provides a number of interesting research
CHAPTER 7. CONCLUSION

challenges. In the following paragraphs, we briefly outline a few of new challenges to the work presented in this thesis.

Our first exploration on user privacy is to enhance the privacy controls on uploaded content for end users. Our measurement study has inspired many following studies in quantifying the privacy problem for different kinds of content on different OSN sites. Due to the upgrade of Facebook API, our tool Friendlist Manger can not fully function as before, since we can not automatically create the friend lists for users. However, it still provides the insights for users to determine the appropriate set of friend lists to create. One potential next step is to explore mechanisms for automatically suggesting names for the friend lists, further removing the burden on end users. Also, we plan to improve the user interface to make it more intuitive and easy to understand.

Second, to examine how OSNs utilize the user data for the ad services, we focused exclusively on the largest OSN ad market Facebook, and provide a new methodology to evaluate user value using suggested bid feature. As the Facebook ad market continues to mature, we plan to repeat our analysis to study the evolution of the market. Moreover, as more OSNs develop advertising markets, our approach can be used to measure these markets as well; it is typical for ad markets to provide suggested bids. In other words, we can apply the proposed methodology to analyze the underlying models, and compare user value. For example, Twitter has developed and deployed advertising markets in 2013 [11, 143]; we hope to be able to apply our methodology to those sites.

Last but not least, we proposed a novel methodology for detecting user PI embedded in large amounts of network flows, which can help us fully understand the scope of the privacy implications and develop approaches to prevent privacy leaks. In the future, we plan to extend the work to differentiate between PI the user has intentionally shared and other that was not. One possible way to do this is by occasionally surveying users about observed leaks, learn of a few (in)voluntary ones, and extend the knowledge across users and web services. Eventually, we aim to build a system capable of informing users when personal information is being leaked without an explicit act on their side, so that they can decide whether the leak should be allowed or blocked (e.g., by substituting information with placeholders). Overall, the general problem of privacy implications and privacy leaks in the online systems remains an open challenge.
Bibliography


BIBLIOGRAPHY


BIBLIOGRAPHY


[40] P. W. L. Fong, M. Anwar, and Z. Zhao. A privacy preservation model for Facebook-style social network systems. ESORICS, Berlin, Germany, 2009.


BIBLIOGRAPHY


BIBLIOGRAPHY


[66] P. Hornyack, S. Han, J. Jung, S. Schechter, and D. Wetherall. These Aren’t the Droids You’re Looking for: Retrofitting Android to Protect Data from Imperious Applications. CCS, Chicago, IL, USA, Oct. 2011.


BIBLIOGRAPHY


BIBLIOGRAPHY


[90] Y. Liu, C. Kliman-Silver, and A. Mislove. The tweets they are a-changin’: Evolution of Twitter users and behavior. *ICWSM*, Ann Arbor, MI, USA, June 2014.


BIBLIOGRAPHY


BIBLIOGRAPHY


BIBLIOGRAPHY


BIBLIOGRAPHY
