Exploiting Phone Numbers and Cross-Application Features in Targeted Mobile Attacks

Srishti Gupta
Indraprastha Institute of Information Technology
Delhi, India
srishtig@iiitd.ac.in

Payas Gupta
New York University, Abu Dhabi
Abu Dhabi, UAE
payasgupta@nyu.edu

Mustaque Ahamad
Georgia Institute of Technology
Atlanta, USA
mustaq@cc.gatech.edu

Ponnurangam Kumaraguru
Indraprastha Institute of Information Technology
Delhi, India
pk@iiitd.ac.in

ABSTRACT

Smartphones have fueled a shift in the way we communicate with each other via Instant Messaging. With the convergence of Internet and telephony, new Over-The-Top (OTT) messaging applications (e.g., WhatsApp, Viber, WeChat etc.) have emerged as an important means of communication for millions of users. These applications use phone numbers as the only means of authentication and are becoming an attractive medium for attackers to deliver spam and carry out more targeted attacks.

The universal reach of telephony along with its past trusted nature makes phone numbers attractive identifiers for reaching potential attack targets. In this paper, we explore the *feasibility*, *automation*, and *scalability* of a variety of targeted attacks that can be carried out by abusing phone numbers. These attacks can be carried out on different channels viz. OTT messaging applications, voice, e-mail, or SMS. We demonstrate a novel system that takes a phone number as an input, leverages information from applications like Truecaller and Facebook about the victim and his / her social network, checks the presence of phone number's owner (victim) on the attack channel (OTT messaging applications, voice, e-mail, or SMS), and finally targets the victim on the chosen attack channel.

As a proof of concept, we enumerated through a random pool of 1.16 million phone numbers and demonstrated that targeted attacks could be crafted against the owners of 255,873 phone numbers by exploiting cross-application features. Due to the significantly increased user engagement via new mediums of communication like OTT messaging applications and ease with which phone numbers allow collection of pertinent information, there is a clear need for better protection of applications that rely on phone numbers.

SPSM'16, October 24 2016, Vienna, Austria

DOI: http://dx.doi.org/10.1145/2994459.2994471

1. INTRODUCTION

The convergence of telephony and the Internet with technologies like Voice over IP (VoIP) is fueling the growth of Over-The-Top (OTT) messaging applications that allow smartphone users to communicate with each other in myriad ways. OTT messaging applications (like WhatsApp, Viber, WeChat), [1] and VoIP applications (like Skype, Google Hangouts) [2] are used by millions of users around the globe. In fact, the volume of messages via OTT messaging applications has overtaken traditional SMS [7] and e-mail [35]. As a result, OTT messaging has become an attractive attack vector for spammers and malicious actors who are now abusing it for illicit activities like delivering spam and phishing messages. For example, unsolicited messages like investment advertisements, adult conversation ads, and random contacts requests were seen to propagate on WhatsApp [4].

OTT messaging applications use phone numbers for user authentication and communication. Authentication is typically done when a user registers with the application by providing his / her phone number and the validity of the phone number is verified by delivering an SMS message to it. A phone number is a personally identifiable piece of information with which an individual can be associated uniquely, in most cases [37]. Although there exist burner phones [3] in some countries where a phone number may not be reliably associated with a person, in an overwhelming number of cases, phone numbers are linked to a wealth of information about their owners like name and place where he / she lives. Such phone numbers are a verified part of user identity because one needs to obtain a physical SIM card and complete the verification process of a service provider to obtain a phone connection. Service providers often require personal information to setup an account.

Fraudulent communication carried out with phone numbers has already resulted in loss of millions of dollars to individuals and organizations [31, 32, 34] despite the fact that phone numbers are considered private information and

[1] http://marketingland.com/four-top-six-social-networks-actually-chat-apps-115168
[2] http://beebom.com/2015/09/best-voip-apps
[3] https://www.puretalkusa.com/blog/what-is-a-burner-phone/

not easily exposed by many platforms. Attackers who want to target certain users may prefer phone numbers over other identities like e-mail addresses and online social identifiers due to multiple reasons - a) Phone numbers are ubiquitous due to smartphone penetration growing in all population segments of society, both rural and urban, and amongst all age groups too [26]. Therefore, attackers can expect more reachability, in terms of potential victims, while abusing phone numbers; b) In most countries, a verification process is required before someone can obtain a phone number. Because of this, attackers cannot obtain phone numbers as easily as e-mail addresses and online social identities which can be easily created and faked. As a result, there is a greater degree of trust associated with phone numbers as compared to other user identifiers; c) Phone numbers are personal and more persistent. People generally retain the same phone number for a long time due to the cost associated with it, where as, one can have multiple e-mail address and online identities. As victims would be using the same phone number, attackers can abuse it to increase the success rate of their attacks. On the other hand, due to multiplicity nature of e-mail address, the success rate for attackers to find and exploit currently being used e-mail addresses would be low; d) Phone calls and text messages are synchronous in nature and have faster response time than e-mail. This time sensitivity can be leveraged by the attacker to his / her benefit; e) We have mature and effective defenses against e-mail spam but this is not true for phone and messaging spam. Thus, attackers have an advantage when they exploit the telephony channel.

The proliferation of messaging, voice, and other related smartphone applications result in collection and access to a wealth of information about owners of smartphones. In this paper, we explore if malicious actors can exploit these applications to collect and aggregate information about intended victims to craft more targeted attacks. In contrast, most prior research has explored phone number abuse with either voice (vishing) / SMS spam, which aim to either collect personal information or direct users to fraud websites [21]. Although various e-mail tricks have been seen in the past, they are not well known on messaging apps.

In this paper, we explore how attackers can exploit phone numbers to launch targeted attacks over phone and other communication channels. We first demonstrate how a phone number can be used across multiple applications to collect private and personal information which can later be aggregated for targeted attacks. Reverse-lookup contact feature used by caller ID applications like Truecaller [4] can be exploited to find more details (e.g. name) about the owner of the phone number. Furthermore, by correlating this with *public* information present on online social networking platforms (e.g., Facebook), attackers can determine the social circle (friends) of the victim. Finally, address book syncing feature of OTT messaging applications allows attackers to determine what applications certain users are using on their smartphones. Based on this, attackers can identify the specific OTT messaging applications (e.g., WhatsApp) that can be used to reach the users.

In addition, we show a novel targeted vishing attack that can be carried out by compromising the integrity of caller ID applications. Although the phone number itself is verified, several caller ID applications do not check validity of other information provided to these applications during registration, which is what the user actually rely on. For example, a malicious user can associate the name of a legitimate bank to impersonate bank officials and trick people into giving out their personal information like bank account number, credit card number etc. The success rate of vishing attacks can be increased by making them more personalized and targeted using information collected from OTT messaging applications. Finally, we provide early evidence of crafting whaling attacks [9] against the owners of vanity numbers [8], phone numbers generally owned by people with high influence or high-net-worth individuals.

By developing an automated and scalable system that uses phone numbers to facilitate targeted attacks to be crafted at scale, we make following contributions:

***Feasibility*:** This is the *first* attempt to systematically understand the threat posed by the ease of correlating user information across caller ID lookup application (Truecaller) and social networking application (Facebook) using phone numbers as unique identifiers. We show the attack is feasible with easily available computational resources, and poses a significant security and privacy threat.

***Automation*:** To carry out attacks on a large scale, the entire attack cycle from determining the attack channel to the launch of an attack should be automated. We design and implement an automated system that takes a phone number as an input and targets the victim on the attack channel.

***Scalability*:** The attack strategy should be scalable, and our system is scalable to a large user population. This is based on the level of information that is available about the users. For 1,162,696 random pool of Indian phone numbers that we enumerated, it is possible to launch social and spear phishing attacks against 51,409 and 180,000 users respectively. Vishing attacks exploiting caller ID applications can be launched against 722,696 users. We also found 91,487 highly influential victims who can be attacked by crafting whaling attacks. This emphasizes the magnitude and significance of the attack.

To the best of our knowledge, this is the first exploration of large-scale targeted attacks abusing phone numbers that rely on smartphone apps. We demonstrate how phone numbers enable collection of information about users by exploiting OTT messaging applications and caller ID applications. Such information can facilitate highly targeted attacks for which we currently lack effective defenses. Given that the telephony medium is not as well defended as e-mail, we believe that these contributions offer a promising new direction and demonstrate the urgent need for better security for such applications.

2. FEASIBILITY AND AUTOMATION

In this section, we demonstrate the *feasibility* and the *ease* with which different targeted attacks can be crafted by abusing phone numbers. To *automate* the whole process, we build a system that exploits cross-application features to collect information about a user and determines the attack channel (OTT messaging applications, voice, e-mail, or SMS) and targeted or non-targeted attack vectors (see Figure 1). Specifically, the system has three main steps: a) Based on a numbering plan, phone numbers are randomly generated and inserted into an address book of a smartphone. This address book is on a device that is under the

[4]http://truecaller.com/

control of the attacker; b) The system fetches data from Truecaller and Facebook applications to determine any additional information about these phone numbers; c) After the information is aggregated, the system determines the attack channel viz. OTT messaging applications, voice, e-mail, or SMS to launch targeted attacks. The attacker can craft multiple attack vectors, depending on the information gathered about the victim and the presence of victim on a particular attack channel, to launch attacks against victims (see Section 3).

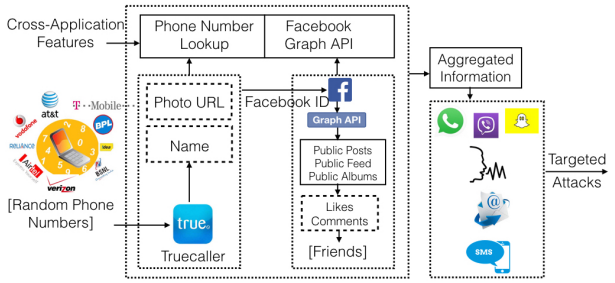


Figure 1: System for Cross-Application Information Gathering and Attack Architecture.

2.1 Step 1: Generating Phone Numbers

This section elaborates phone number generation and setting up the device under attacker's control. The system generates a large pool of phone numbers which could be used as a seed by an attacker to launch targeted attacks. There are several methods to obtain a pool of phone numbers - consolidating white-pages directory entries or any other public online directories, or scraping the Internet using regex patterns. We chose the easiest method for an attacker, i.e., taking random phone numbers as initial seeds, incrementing the digits by one to obtain a sufficient pool. Unlike e-mail addresses, the phone number set is finite, therefore, an entire range can be enumerated and inserted into the address book. This may result in some phone numbers that are currently not allocated to any user. Once phone numbers are generated, the attacker initializes the address book of the device under his control with these numbers. The phone numbers added to the address book are now his potential victims for carrying out various kinds of targeted attacks as demonstrated in this paper.

2.2 Step 2: Using Phone Numbers to Collect Information for an Attack Vector

In this step, the system aggregates all the available information to launch an attack against the victim. To obtain information about the victim, we used Truecaller, an application that enables searching contact information using a phone number. Its legitimate use is to identify incoming callers and block unwanted calls. It is a global collaborative phone directory that keeps data of more than 1 billion people around the globe. We used Truecaller as an example, but any such application (e.g., Facebook Hello, WhitePages, Contactive etc.) can be used to determine this information. Truecaller also maintains data from social networking sites (Facebook, Twitter, and LinkedIn) and correlates this information to create a large dataset for people who register on it. Also, due to its address book syncing feature, it retrieves information about contacts (friends) of the "owner of the phone number" who installed it too. The 'search' endpoint of Truecaller application provides details of an individual like:

> *name, address, phone number, country, Twitter ID, e-mail, Facebook ID, Twitter photo URL*

However, the private information obtained is according to the privacy settings of users.

We automated the whole process of fetching information about phone numbers from Truecaller by using the search end-point (used to search information about a random phone number) to obtain the registration ID corresponding to a particular phone number. [5] This was necessary to make authenticated requests and retrieve the information from their servers. We extracted the registration ID from the network packet sent while searching a random phone number on Truecaller application installed on our iPhone. Once the registration ID was obtained, we programmatically fetched information for phone numbers in our dataset. Multiple instances of the process were initiated, on a 2.5 GHz Intel i5 processor, 4GB RAM at the rate of 3000 requests / min. We worked with only one registration ID such that we do not abuse the Truecaller servers and effect its services, however, it is easy for an attacker to scale the process by collecting multiple registration IDs to bypass rate limits imposed by Truecaller.

To obtain the social network of the victim, we used Facebook, the largest social network of family and friends. We assume that friends obtained will be related to the person in some way or the other which can increase the probability of success of a social phishing attack. Truecaller aggregates data from various social networking websites and sometimes provides a link to the public profile picture of the victim on Facebook. We extracted Facebook ID from these links to retrieve friends of the victim on Facebook. Extracting friends from victim's profile is a non-trivial task, since everyone does not have their friendlist set as public. Therefore, we decided to use victim's public sources like public feed, photo albums, and public posts on Facebook to obtain friends information [6], assuming users liking / commenting on any of these public sources are friends of the victim. However, since these sources are public (not visible to only friends), there might be a possibility that the person commenting / liking that post might not be a friend on Facebook. To validate the above hypothesis, we performed a small experiment to determine if friends obtained from public sources on Facebook are a subset of public friendlist. Even though normal access token from Facebook does not provide these details, we were able to fetch the information using a never-expiring mobile OAuth token obtained from iPhone's Facebook application. [7] We monitored the data packet sent while launching Facebook application on our iPhone device and extracted the authentication token to make further requests.

We collected a random sample of 122,696 Facebook IDs

[5]We used this phone number only for research purposes and nothing else.
[6]https://developers.facebook.com/docs/graph-api
[7]We are not sure if this is an additional feature provided by Facebook or a bug in their system. At the time of writing this paper, we did not find any official Facebook documentation about it.

and obtained 95,756 friends from public sources and 80,979 friends from public friendlist (see Figure 2). There were only 62,574 users for whom we were able to find friends from both public sources and public friendlist. Out of which, we found that 42,552 (68%) user-IDs liking and commenting on public sources were part of victim's friendlist with more than 95% matching rate. As observed in Figure 2, in some cases friends from public sources were not a complete subset of friends from public friendlist. We obtained 5,881 friends with 90 - 95% matching, 3,754 friends with 85 - 90% matching, and 10,387 friends with less than 85% matching. This could be because some users might have disabled all platform applications from accessing their data. In this case, they might not appear anywhere in any Facebook API [3]. To launch attacks using friends information, friends can be picked from public friendlist, if available, else, the attacker can rely on public sources to extract friends. Therefore, we extracted the Facebook ID from the photo URL (using regular expression) obtained from Truecaller response, and obtained public sources using Facebook Graph API to find friends on Facebook to craft social phishing attack vector. For example, the following data object was obtained for one of the phone numbers in the dataset –

```
{
"NAME": "XXXXX",
"NUMBER": "+91XX0000000X",
"COUNTRY": "India",
"PHOTO_URL": "http://graph.facebook.com/
    XXXXXX/picture?width=320&height=320",
"e-mail": " "
}
```

The Facebook ID was parsed from `PHOTO_URL` and used to make further requests. E-mail addresses for some users were also available which can be used to target them.

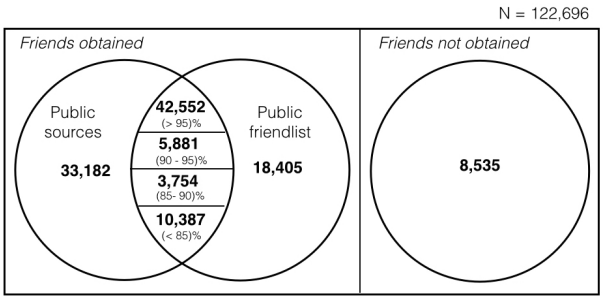


Figure 2: Relation between friends obtained from public sources and public friendlist. Friends from public sources are found to be a subset of friends from public friendlist in 68% cases (with more than 95% matching).

To increase the success rate of attacks, an attacker can choose a Facebook friend X who is closer (greater affinity) to the victim than Facebook friend Y (lesser affinity). To do this, we propose an algorithm (see Algorithm 1) which an attacker can use to find a friend having greater affinity to the victim. The algorithm returns Facebook friends that can be obtained from public sources (from Facebook) in the form of 4 clusters; according to their level of closeness with the victim. Input to the algorithm is friendlist i.e., a list which contains all the friends obtained from victim's Facebook account, and output is clusters that rank friends based on their affinity / closeness. Friends are ranked based on their frequency (count) of comments / likes on the victim's public sources (feed, posts, and albums) on Facebook. *We assume that a friend commenting on the post in closer than the one just liking the post.* We believe that success rate for choosing a close friend which can deceive the victim might not be 100%, however, failure rate would be small which can be ignored.

An attacker can utilize the information, as obtained in this step and craft attack vectors described in Section 3. In case no information about the victim is obtained, the attacker can craft non-targeted attacks. Apart from applications like Truecaller and Facebook that we explored in this paper, attackers can exploit CNAM (Caller ID Name) database [8], a database that is linked to names of calling number. This service which is operational in US, provides information associated with a landline number. Attackers can use this to obtain basic information about their targets which is out of scope of current work.

Algorithm 1 Friend Affinity Clusters.

```
1: procedure FRIEND_AFFINITY_CLUSTERS(friendlist, v))
2:     friendlist ← φ
3:     clusters ← φ
4:     for each v ∈ friendlist do
5:         if v ∈ friendlist then continue
6:         cluster ← AFFINITY_SCORE(v)
7:         clusters ← clusters ⋃ cluster
8:     return clusters

9: procedure AFFINITY_SCORE(v)
10:    if v ∈ likes_array & v ∈ comments_arr then
11:        cluster_rank_1 ←v
12:        return cluster_rank_1
13:    else if v ∉ likes_array & v ∈ comments_arr then
14:        cluster_rank_2 ←v
15:        return cluster_rank_2
16:    else if v ∈ likes_array & v ∉ comments_arr then
17:        cluster_rank_3 ←v
18:        return cluster_rank_3
19:    else
20:        cluster_rank_4 ←v
21:        return cluster_rank_4
```

2.3 Step 3: Identifying Attack Channel

Once the data is collected about a phone number, the system determines the channel (OTT messaging applications, voice, e-mail, or SMS) to phish the victim. This entirely depends on whether the victim is present on that particular channel.

OTT messaging applications: If the attacker decides to choose OTT messaging applications like WhatsApp, Viber, or Snapchat, he needs to ensure if the victim is using one of these applications. This is achieved by exploiting the address book syncing feature in OTT messaging applications. Once a user registers himself on these applications, his contacts in the address book are uploaded (automatically, for

[8]http://www.voip-info.org/wiki/view/CNAM

some applications) to the OTT messaging applications' service provider and are matched against the users of the application to find already existing contacts. Only the information about the owners of the phone numbers present in the address book is retrieved.

Voice and SMS: In addition, an attacker can choose voice or SMS as the attack channel to phish their victims. Similar to OTT messaging applications, to target victims on this channel, an attacker needs to gather relevant information without checking the presence of the victim beforehand.

E-mail: Since so many people around the world depend on e-mail, it is the most lucrative channel for phishing attacks. Attackers lure people in giving away their information or entice them to take some action. Truecaller's search response (as shown in Section 2.2) gives an "e-mail" field which can be used to phish users. Attackers can craft these e-mails to look convincing, sending them out to literally millions of people around the world [5].

3. ATTACK VECTORS

After the attack channel (OTT messaging applications, voice, SMS, or e-mail) is determined, attacker can craft appropriate vector to target the victim. We describe the attack vector generation details for each of the attacks below.

3.1 Targeted Phishing Attacks

Based on the data that we are able to collect in this paper, spear phishing attack vectors can be crafted by using victim's name, as obtained from Truecaller. Phishing attacks can be better targeted by making them appear to be coming from a friend within victim's own network, also known as social phishing. Friends' information can be conveniently chosen to gain trust, therefore, the attacker uses victim's name and one of his friend's information (i.e., friend's name) to craft the attack vector. This information is obtained from Facebook, as discussed earlier in Section 2.2. For example, we found person X to be present on Truecaller and his friend (Y) on Facebook. Now, a phishing message against X can be launched on WhatsApp pretending it to be coming from Y. Since X knows Y, there is a high chance that X will fall for the phishing attack.

Non-targeted attacks are aimed to target as many users as possible. The goal is to reach out to a large audience and not to target a particular individual. Since it only requires the knowledge whether the victim is present on the channel, this can be achieved by crafting a non-targeted phishing attack, even if no information about the victim is available.

3.2 Targeted Vishing Attacks

Targeted Vishing attacks can be carried out by manipulating the information provided by crowd sourced caller ID applications (for instance, Truecaller). Cloud-based caller identification applications are emerging to help in getting additional information about the caller. Millions of people are using such applications, namely Truecaller, Facebook's Hello, [9] and Whitepages Caller ID and Block. [10] In general, these applications allow an individual to register using his / her phone number and help in identifying the caller by showing the information (like name) from their respective databases. Caller ID applications also gather information from social networking sites to collect more information about the caller. Scammers can undermine such an application to vish their victims. This information which users enter during registration can be exploited to conduct targeted social engineering / vishing attacks. Specifically, a) scammers can register a phone number (controlled by him / her) as a trusted bank / company / organization in which a user is interested in or is dealing with; b) spoof one of the already registered phone numbers with the caller ID applications and call victims such that the call appears to come from a real entity.

[9]http://www.engadget.com/2015/04/22/facebook-hello/
[10]http://www.whitepages.com/caller-id

Fake Registration.

Here, we show that an attacker can add spurious information in caller ID applications fairly easily, thus compromising the integrity of the information provided by them to launch vishing attacks. Associating an identity with a phone number increases the trust of an individual and likelihood to pick a call. Since caller ID applications do not have a mechanism for verification of the users' details, and rely on the information provided by the user while registering, it is easy for an attacker to abuse this trust. For example, an attacker can register as multiple fake banks on multiple caller ID applications (see Figure 3). For registration, he needs a smartphone device with working phone connection. It is a manual process where a short SMS code is sent from caller ID applications to verify the phone number. Since the number of banks are limited, it is not difficult for the attacker to do this manually.

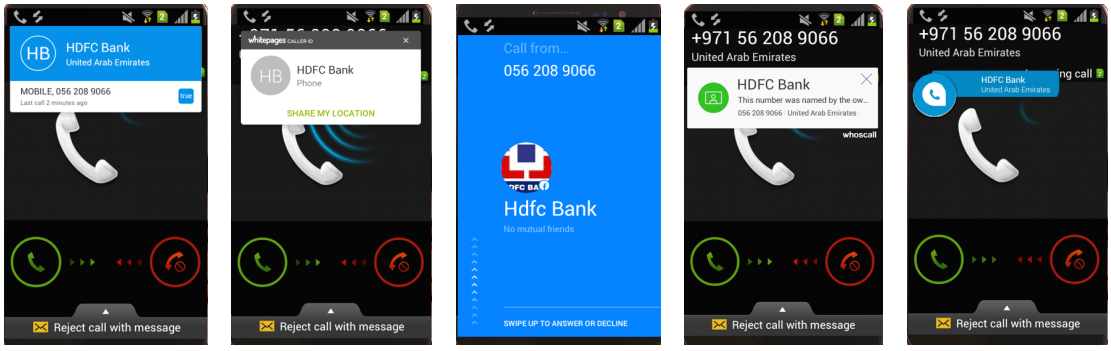
From the data we collected in this paper, attacker gets to know the victims who are using Truecaller. This is required to ensure success of the attack, as for users not using Truecaller, the call appears to be coming from a random phone number. Further, fetching details about the victim (person to be called) from Facebook can increase the success of such attacks by making it personalized. Since vishing attacks are already known to be successful [2, 6], we believe success rate of targeted attacks would be higher or at-least equivalent.

Caller ID spoofing.

Another form of attack uses caller ID spoofing which can be carried out by imitating already registered phone numbers or other phone numbers whose details were uploaded by caller ID applications exploiting address book syncing feature. As a user must have entered some details about him / her while registering, it makes him / her a more likely target, than an unknown phone number (not present in the address book) flashing on the screen. For example, based on the information we are able to collect in this paper, we know that Truecaller has information about X and Y. Now, the attacker can spoof X's number and call Y making him believe that X is calling. Note that this attack is similar to the above one with the difference that the attacker is in control of the phone number, so any callback from the victim on the phone number will be picked up by the attacker in the former as compared to the later. Further, the information about X and Y as obtained from Truecaller and Facebook can be used to increase the success of vishing attack.

3.3 Whaling attacks

Whaling attacks [9] that are directed specifically at senior executives or other high-profile individuals within a business, government, or other organization, can be crafted on OTT



(a) Truecaller (b) Whitepages Caller ID & Block (c) Facebook's Hello (d) Whoscall-Caller ID and Block (e) Contactive

Figure 3: Incoming call showing fake HDFC bank (example in our case) on various caller ID applications.

messaging applications or voice channel. It uses the same technique as above mentioned targeted phishing / vishing attacks but the intended victims are people with high influence or high-net-worth individuals. There is a particular set of phone numbers reserved by mobile operators for politicians, bureaucrats, and people willing to invest large amount of money to get a phone number. They are called Vanity / VIP / Fancy numbers and follow a specific pattern. [11] For example, in India, it could be one digit repeated several times, `99999-xxxxx` or `xx-8888-xxxx`; two digits, `xx-85-85-85-xx`; or in different orders, `xx-123-123-xx` or `xx-11-112233`. Vanity numbers are not restricted to India, different countries follow different patterns. [12] The main advantage of vanity phone numbers over standard phone numbers is increased memorability. Since these are bought at higher price, owners of these phone numbers can be assumed as people with high-net-worth. [13] For very special numbers, network providers host auctions online where people can purchase these numbers [1]. Using only vanity numbers in the address book, attackers can launch whaling attacks that only targets HNIs (High-net-worth individuals) by sending them targeted or non-targeted phishing messages or initiating vishing calls.

4. SCALABILITY

To define *scalability*, we assume that an attacker starts with no information about its potential targets. The attack method's scalability can be characterized by the fraction of people who can be reached over an attack channel, and targeted attacks can be launched against them. To demonstrate the scalability of our attacks, we enumerated through a list of 1,162,696 random Indian phone numbers, out of which 255,873 phone numbers can be abused using the attacks proposed in this paper. Since these numbers are chosen randomly, no additional information is available about their owners at the beginning. We demonstrate the scale at which each of the proposed attacks can be carried out with the techniques described earlier. We chose a set of random Indian phone numbers, however, numbers from any country can be used for attacks.

4.1 Targeted Attacks on OTT Messaging Apps

We forged the address book of an Android device by inserting all these numbers in multiple phases. The next step was to collect attributes associated with the owner of the phone number (victim). Truecaller (TC) was used to collect more information about the victims. Detailed information for 722,696 (62%) users was collected using Truecaller; name was obtained for all the users as shown in Figure 4. For rest of the users whose information cannot be obtained from Truecaller, non-targeted phishing attacks can be launched against them. To craft more targeted and personalized attacks, i.e., social phishing attacks, friends information was leveraged from Facebook (FB). Social circle information was obtained for 114,161 (93%) out of 122,696 users; 80,979 from public friendlist and 33,182 friends from public sources. To check the presence of these numbers on an attack channel, they were synced with WhatsApp application (WA) using address book syncing feature. About 51,409 users were present on WA, who can be social phished. Spear phishing attacks can be launched against other 180,000 users whose social circle was not obtained, but were present on WA. Numbers which were not found on WhatsApp either may not be allocated to any user or may not be registered on it.

Spear phishing attacks can be launched against 600,000 users on voice or SMS as their information was available on TC, but not FB. In addition, 122,696 users can be social phished on voice or SMS. E-mail address for 81,389 users were obtained from Truecaller; 13,754 can be social phished and 67,635 can be spear phished on voice or SMS.

4.2 Targeted Attacks on Caller ID Apps

Personal information for 722,696 users was found on TC against 1,162,696 phone numbers searched. Vishing attacks can be crafted against the owners of these phone numbers. The information can be used to either deceive victim by spoofing an already registered number, or register and fake bank and use personal information against the victim.

Automated User Profiling.

In addition to launch targeted attacks on OTT messag-

[11] http://www.openthemagazine.com/article/real-india/calling-9999999999

[12] http://store.vanitynumbersource.com

[13] http://articles.economictimes.indiatimes.com/2007-10-13/news/27675454_1_digit-numbers-mukul-khanna-minimum-price

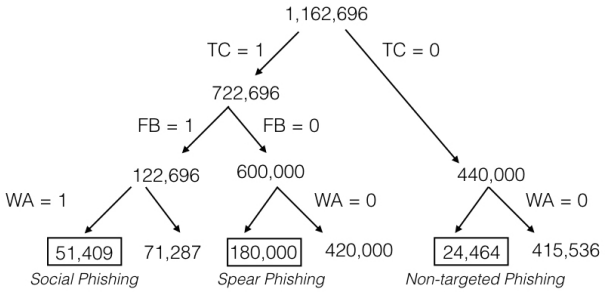


Figure 4: Data collection to demonstrate scalability of phishing attacks of the system choosing OTT messaging applications as the attack channel, WA–WhatsApp, TC–Truecaller, FB–Facebook.

ing applications, voice, e-mail, or SMS, it is easy to profile a victim in an automated manner. The detailed information about the victim can be used against him. The goal is to investigate victim's profile deeply and gather as much information as possible. Automated user profiling is possible by fetching details from Facebook. We could extract Facebook IDs for 122,696 users as shown in Figure 6. Using Facebook Graph API, we obtained following details for these users: gender (112,880), relationship status (57,755), work details (92,352), school information (110,426), employer details (106,746), birthday (9,728), and hometown (80,979). The collated information can be used to increase the success rate of targeted vishing attacks.

4.3 Whaling Attacks

As the owners of vanity numbers might belong to elite members of the society, they can be of particular interest to attackers. We looped through the "patterns" available from an e-auction website to enumerate vanity numbers pool [1]. We initialized our smartphone's address book with 171,323 vanity numbers. We found 91,487 vanity numbers on Truecaller and 11,286 on Facebook. They were synced with WhatsApp and 5,756 (51%) were found on it. Whaling attacks with social information was obtained for 11,286 users which can be attacked on voice or SMS. However, only name was obtained for 80,201 users who can be made targets on voice or SMS. E-mail address for 11,013 users was obtained; social information was obtained for 1,354 users.

Automated User Profiling.

Out of 11,286 vanity numbers that were found on Truecaller and Facebook; we obtained personal information (using Facebook) about owners as follows: gender (10,246), relationship status (3,733), birthday (726), work details (6,729), school details (10,994), employer details (9,801), and hometown (6,952). We manually analyzed Facebook profiles of 100 random vanity number owners to find their occupation details and found director / CEO / chairman (10), student (10), engineer (12), consultants (2), business (5), accountant / officer (8), lecturer (5), manager (8), bank officials (12) for 70 user profiles.

4.4 Ethical and Legal Considerations

The main motivation of our data collection was to demonstrate the feasibility and scalability of targeted attacks by abusing a phone number. The aim was to highlight how naive, minute information can be exploited to carry out large scale targeted attacks, and there exists no mechanism to prevent it. In order to find security weaknesses in several OTT messaging applications and caller ID applications, we found several possible attack vectors that can be used to abuse or exploit phone numbers. The goal of this work was not to collect personal information about individuals, but to explore how multiple applications can be abused to collect personal information. As a proof of concept, we collected information about owners of limited set of random phone numbers ensuring that the collected information is not made available to any other organization or individual. We collected only the public information available on Facebook using it's Graph API. Overall, our aim is to create awareness of the risks involved in collating information exploiting cross-application features and prevent future harm.

4.5 Attack Limitations

There are certain factors that can raise the difficulty bar for an attacker, and prevent large scale attacks. Applications like Truecaller and WhatsApp have incorporated certain measures to prevent large scale attacks. For instance, Truecaller limits the number of queries that can be made to search a random phone numbers at a particular time. Recently, we noticed that WhatsApp blocked the account if repeated large number of contacts are uploaded in the address book. Although an attacker can devise strategies to mitigate these issues, like sending requests in batches, however it increases the computational load for him / her. WhatsApp has recently added spam reporting feature which allows a user to report a phone number as spam [14]. These nudges can help a user take an informed decision about an incoming phone number and reduce the success rate of attacks proposed in this paper. Since phone number is a sensitive information, we could not conduct attacks in real-world.

5. DISCUSSION

In this section, we present a discussion on countermeasures to mitigate or limit the extent of attacks presented in this paper. We have spoken to a few popular caller ID applications providers who acknowledged the threat and are working towards fixing the problem.

Due to inherent trust on phone numbers and the fact that spammers are moving towards abusing this unique identifier, there is a dire need to protect its abuse. Content-based filters have been created to filter legitimate and spam content on e-mails. However, due to the lack of sufficient data on OTT messaging applications, for instance, end-to-end encryption on WhatsApp, it is difficult to implement similar approach. Moreover, content-based approach which previously worked with e-mails might not work well with spam related to phone numbers due to short text received in the messages as compared to e-mails.

To fill this gap, there is a need to use existing infrastructure to develop solutions and filter bad phone numbers. One of the solutions we plan to implement, as part of our future work, is to create a phone reputation system that can model bad phone numbers. Although IP and domain reputation systems have been in existence, something similar related

[14]http://www.ibtimes.co.uk/whatsapp-rolls-out-new-spam-blocker-feature-1497715

to phone numbers does not exist [10, 24]. Several industry based solutions have been proposed in this direction like Whitepages Phone Reputation API [15] and Pindrop's PRS [16] . These services provide information about phone numbers that appear in e-mails, online complaint sites, or directly from applications. However, there are large number of other sources, for instance, online social networks, that have traces of campaigns related to phone numbers, which go unnoticed by these services. Services like Truecaller also label a phone number as spam phone number (or low reputation), however, it is fairly easy to add noise in these crowdsourced platforms, as discussed in Section 3.

Specifically, our reputation system will help users in making a conscious decision (real-time) before responding to text or message received from that particular phone number on OTT messaging applications or online social networks. The reputation system would include a learning model which will learn features from past fraudulent numbers, and detect new phone numbers. To accomplish this task, we plan to develop self-adaptive algorithms capable of filtering bad phone numbers, information about which can be leveraged from online social networks. The phone intelligence derived from these reputation systems can be used by OTT messaging applications and caller ID applications. For instance, OTT messaging applications can do an initial filtering while registering a bad phone number. Similarly, caller ID applications can associate a spam score to the bad phone number, and display alerts to the user. In addition to reputation system which can be used by OTT messaging applications and caller ID applications, we propose some recommendations on how to alleviate (if not eliminate) the security risks created by these exploits.

Recommendations to OTT Messaging Applications.

Given the plausibility of targeted attacks on OTT messaging applications, as depicted in this paper, this medium needs to be defended. Humans are the weakest link in security, many are not security-aware and there is too much implicit trust. However, even advanced users can be deceived on mobile platforms due to lack of sufficient authentication mechanisms. Therefore, it is imperative that platforms implement some solutions to combat the problem. We propose following recommendations.

Restrict Address Book Syncing: In this paper, we described a common weakness across OTT messaging applications, which is inherent feature to sync address book contacts. As a countermeasure, OTT messaging applications can restrict address book sync feature, such that people can be added only based on requests (like Facebook), or unique secret like BBM pin. In addition, we suggest that OTT messaging applications should not provide any personal information (profile picture, online status etc.) about new friends (contacts) added after automated address book syncing. Only after a sanity check, where people verify knowing each other, more information about them should be updated. Perhaps these recommendations increase user load in handling such services, trade-off between security and usability always remains a challenge.

Rate Limit Queries: Another viable option to limit our attack is the huge number of queries that we are able to perform. As a solution to this, OTT messaging applications should limit the number of contacts that can be uploaded in the address book.

Crowdsourced Phishing or Spam Score: In order to effectively defend against phishing messages, one solution could be assigning crowd-sourced phishing / spam score to the incoming phone number. OTT messaging applications can filter messages coming from a phone number with high phishing score.

Recommendations to Caller ID Applications.

There is also a necessity to ensure the integrity of the information provided by caller ID applications, as people rely heavily on them to know about the incoming call and trust the information provided by these services. As discussed in the paper, an attacker can add spurious or wrong information while registering on caller ID applications. To combat against such attacks, caller ID applications can adopt following countermeasures.

Verification: One of the biggest challenge that caller ID applications have to face is to implement verification of the information provided by users. Currently, at the time of registration, only phone numbers are verified, and neither the entity behind these phone numbers nor the details of owners of these phone numbers are verified. Caller ID applications can check the integrity of specific business organizations with appropriate authority, listing them as verified users, and routinely scan for any malicious activity in these accounts. However, this puts an additional cost on caller ID applications, but increases the trust of its customers in their services.

Expanding User Information During Incoming Call to Help Users: Additional information can be provided about the caller / the owner of the phone number, during the incoming call. For instance, applications can record the timestamp when the account was registered and call frequency patterns. These details can be provided to the user so that he / she can make an informed decision about the caller. In addition, social information about the caller can be displayed, like number of mutual friends, presence on social networks etc. Caller ID applications can design several metrics like social rank based on the information aggregated across social networks. If the same name appears across multiple networks and the user is found to be active, he / she can be assigned a higher score than a passive user. Sharing this kind of information poses a privacy threat to legitimate users, therefore, caller ID applications can restrict information expansion for only potentially bad phone numbers.

Apart from this, an effective defense technique is to educate users about privacy implications of using such platforms. To combat the abuse, there have been services in place, for instance, CNAM lookup databases [17] for landlines numbers and Secure Telephone Identity Revisited (STIR) working group [18] that aim to authorize the calling party to use a particular phone number. Recently, some services have initiated defense in this direction. WhatsApp incorporated spam blocker feature as a first step in this direction, though their effectiveness need to be studied. [19]

[15]http://pro.whitepages.com/developer/documentation/phone-reputation-api/
[16]https://www.pindrop.com/phone-reputation-service/
[17]http://www.voip-info.org/wiki/view/CNAM
[18]https://datatracker.ietf.org/wg/stir/charter/
[19]http://www.ibtimes.co.uk/whatsapp-rolls-out-new-spam-

6. RELATED WORK

Abusing Address Book Syncing in OTT Messaging Applications: Recent work shows that collection of user profiles can be automated and yields a lot of personal information like phone numbers, display names, and profile pictures [17, 28]. Schrittwieser et al. analyzed popular OTT messaging applications like WhatsApp, Viber, Tango etc. and evaluated their security models with a focus on authentication mechanisms [33]. Authors also highlighted the enumeration and privacy-related attacks that are possible due to address book syncing feature of these applications. Antonatos et al. proposed HoneyBuddy, an active honeypot infrastructure designed to detect malicious activities in Instant Messaging applications like MSN [11]. It automatically finds people using a particular messaging service and adds them to its contacts. The findings confirmed the ineffectiveness of existing security measures in Instant Messaging services.

Targeted Attacks and User Profiling: Bilge et al. launched automated identity theft attacks via profiling users on SNS (Social Networking Services) by employing friend relationship with the victims [15]. The authors showed that people tend to accept friend requests from strangers on social networks. In [14], the authors presented "social phishing" experiments. They crawled social networking sites to obtain publicly available information about users and manually crafted phishing e-mails containing certain information about them. This study showed that victims are more likely to fall for phishing attempts if some information about their friends is included in the phishing e-mail. Jagatic et al. showed that Internet users might be over *four times* more likely to become victims if the sender is an acquaintance [27]. Huber et al. presented friend-in-the-middle-attack on Facebook which could leverage social information about users in an automated fashion [25]. They further pointed out the possibility of context-aware spam and social phishing attacks, where attacks were found to be cheap in terms of cost and hardware. Boshmaf et al. highlighted vulnerabilities that can be exploited by social bots to infiltrate OSNs [16]. They showed that social bots can mimic real users and exploit friendship network leading to strong privacy implications. Kurowski showed a manual attack on WhatsApp to retrieve personal information about victims and proposed the feasibility of social phishing attacks against victims [29].

Vishing Attacks and Spam Over Internet Telephony: Due to low cost and scalability of VoIP based calling systems, scammers are using the telephony channel to make millions of call and expand the vishing ecosystem. Prior work has explored the detection and ways to combat scam on VoIP. Griffin et al. demonstrated that vishing attacks can be carried out using VoIP [21]. They illustrated how several vishing attacks can be crafted in order to increase information security awareness. Chiappetta et al. analyzed VoIP CDRs (Call Detail Records) to build features that can classify normal or malicious users during voice communication [18]. The features were built using mutual interactions and communication patterns between the users.

Past literature demonstrates detection of spam over Internet telephony using several techniques like semi-supervised clustering [36], constructing multi-stage spam filter based on trust and reputation of callers [20], building a system using features like call duration, social networks, and global reputation [12], and placing telephone honeypots to collect intelligence about telephony and mobile attacks [13, 22, 23]. Costin et al. showed that in case of spam activities [19], phone numbers are often more stable than e-mail addresses. Hence, phone numbers can be used as a better detection feature to identify online threats. However, collecting data about phone related attacks is more difficult than regular e-mail phishing [30]. The authors also studied real vishing calls to study the modus operandi of spammers who exploit phone channel to gather sensitive information.

While past literature has looked into identity theft attacks against e-mail or online social network users, feasibility of large-scale attacks abusing phone numbers has been unexplored. In this work, we present *first* evidence of feasibility, automation, and scalability of targeted attacks by exploiting cross-application features like address book sync feature of OTT messaging applications like WhatsApp, and exploiting the integrity of information provided by caller ID applications e.g., Truecaller. Since the space of phone numbers, unlike e-mail addresses, is finite and enumerable, the proposed attacks have significant impact.

7. CONCLUSION

OTT messaging and Voice over IP applications are gaining popularity worldwide. These applications have millions of registered users. As much as these applications attract users, spammers find them attractive as well. In this paper, we demonstrated the *feasibility, automation, and scalability* of targeted attacks that can be carried out by abusing phone numbers. We investigated how easy it would be for a potential attacker to launch automated targeted and non-targeted attacks on different channels viz., OTT messaging applications, voice, e-mail, or SMS. We highlighted several possible attack vectors that can abuse phone numbers to launch large scale targeted attacks. We presented a novel, scalable system which takes a phone number as an input, leverages information from Truecaller (to obtain victim's details) and Facebook (to obtain social circle), checks for the presence of phone number's owner on the attack channel (OTT messaging applications, voice, e-mail, or SMS), and finally targets the victim. We collected information for 1,162,696 Indian phone numbers and show how non-targeted and targeted phishing and whaling attacks can be crafted against the owners of 255,873 phone numbers by exploiting cross-application features. In addition, we depicted automated user profiling to collect personal attributes associated with a victim like work details, birthday, relationship status etc. Our experiments demonstrated that we were able to automatically extract information about users that they may actually wish to hide. In addition to utilizing this information to make the attacks more targeted and personalized, it has a significant privacy impact.

8. REFERENCES

[1] BSNL auction for Vanity Numbers. http://eauction.bsnl.co.in/auction1/index.aspx?id=74.

[2] Ex-policeman under suspicion of voice phishing. http://koreajoongangdaily.joins.com/news/article/Article.aspx?aid=2997528.

[3] Fetching friends from Graph API. http://stackoverflow.com/questions/11135053/fetching-list-of-friends-in-graph-api-or-fql-appears-to-be-missing-some-friend.

blocker-feature-1497715

[4] HeadsUp for WhatsApp. http://www.adaptivemobile.com/blog/headsup-for-whatsapp.

[5] How to send 5 million spam emails without even noticing. https://nakedsecurity.sophos.com/2014/08/05/how-to-send-5-million-spam-emails/.

[6] I.R.S Tech Support Scams. http://www.forbes.com/sites/michaelzakkour/2015/04/14/i-r-s-tax-phone-scam-claims-more-victims-than-ever-as-2015-tax-day-arrives/.

[7] Over-The-Top Messaging Apps Overtake SMS Messaging. http://mobilemarketingmagazine.com/over-the-top-messaging-overtakes-sms.

[8] Price for Vanity Numbers. http://articles.economictimes.indiatimes.com/2007-10-13/news/27675454_1_digit-numbers-mukul-khanna-minimum-price.

[9] Whaling? These Scammers Target Big Phish. http://www.scambusters.org/whaling.html.

[10] M. Antonakakis, R. Perdisci, D. Dagon, W. Lee, and N. Feamster. Building a dynamic reputation system for dns. In *USENIX security symposium*, pages 273–290, 2010.

[11] S. Antonatos, I. Polakis, T. Petsas, and E. P. Markatos. A systematic characterization of IM threats using honeypots. In *Proceedings of the 17th Annual Network and Distributed System Security Symposium (NDSS), San Diego, CA*, 2010.

[12] V. Balasubramaniyan, M. Ahamad, and H. Park. CallRank: Combating SPIT Using Call Duration, Social Networks and Global Reputation. In *Conference on Email and Anti-Spam, CEAS*, 2007.

[13] M. Balduzzi, P. Gupta, L. Gu, D. Gao, and M. Ahamad. Mobipot: Understanding mobile telephony threats with honeycards. In *Proceedings of the 11th ACM SIGSAC Symposium on Information, Computer and Communications Security*, ASIA CCS '16, New York, NY, USA, 2016. ACM.

[14] M. Balduzzi, C. Platzer, T. Holz, E. Kirda, D. Balzarotti, and C. Kruegel. Abusing Social Networks for Automated User Profiling. In *Recent Advances in Intrusion Detection*, pages 422–441. Springer, 2010.

[15] L. Bilge, T. Strufe, D. Balzarotti, and E. Kirda. All Your Contacts Are Belong to Us: Automated Identity Theft Attacks on Social Networks. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, pages 551–560, 2009.

[16] Y. Boshmaf, I. Muslukhov, K. Beznosov, and M. Ripeanu. Design and analysis of a social botnet. *Computer Networks*, 57(2):556–578, 2013.

[17] Y. Cheng, L. Ying, S. Jiao, P. Su, and D. Feng. Bind Your Phone Number with Caution: Automated User Profiling Through Address Book Matching on Smartphone. In *Proceedings of the 8th ACM SIGSAC symposium on Information, computer and communications security*, pages 335–340. ACM, 2013.

[18] S. Chiappetta, C. Mazzariello, R. Presta, and S. P. Romano. An anomaly-based approach to the analysis of the social behavior of VoIP users. *Computer Networks*, pages 1545–1559, 2013.

[19] A. Costin, J. Isacenkova, M. Balduzzi, A. Francillon, and D. Balzarotti. The role of phone numbers in understanding cyber-crime schemes. In *Privacy, Security and Trust (PST), Eleventh Annual International Conference on*, 2013.

[20] R. Dantu and P. Kolan. Detecting spam in voip networks. In *Steps to Reducing Unwanted Traffic on the Internet Workshop, SRUTI'05*. USENIX Association, 2005.

[21] S. E. Griffin and C. C. Rackley. Vishing. In *Proceedings of the 5th annual conference on Information security curriculum development*, pages 33–35. ACM, 2008.

[22] P. Gupta, M. Ahamad, J. Curtis, V. Balasubramaniyan, and A. Bobotek. M3AAWG Telephony Honeypots: Benefits and Deployment Options. Technical report, 2014.

[23] P. Gupta, B. Srinivasan, V. Balasubramaniyan, and M. Ahamad. Phoneypot: Data-driven Understanding of Telephony Threats. In *22nd Annual Network and Distributed System Security Symposium, NDSS*, 2015.

[24] S. Hao, N. A. Syed, N. Feamster, A. G. Gray, and S. Krasser. Detecting spammers with snare: Spatio-temporal network-level automatic reputation engine. In *USENIX Security Symposium*, 2009.

[25] M. Huber, M. Mulazzani, E. Weippl, G. Kitzler, and S. Goluch. Friend-in-the-middle attacks: Exploiting social networking sites for spam. *Internet Computing, IEEE*, 15(3):28–34, 2011.

[26] S. Insights. Mobile marketing statistics. http://www.smartinsights.com/mobile-marketing/mobile-marketing-analytics/mobile-marketing-statistics/, July 2015.

[27] T. N. Jagatic, N. A. Johnson, M. Jakobsson, and F. Menczer. Social Phishing. *Communications of the ACM*, 50(10):94–100, 2007.

[28] E. Kim, K. Park, H. Kim, and J. Song. I've Got Your Number: Harvesting users' personal data via contacts sync for the KakaoTalk messenger. In *Information Security Applications: 15th International Workshop, WISA 2014, Jeju Island, Korea, August 25-27, 2014*.

[29] S. Kurowski. Using a whatsapp vulnerability for profiling individuals. *Open Identity Summit,GI-Edition - Lecture Notes in Informatics (LNI) - Proceedings 237*, pages 140–146, 2014.

[30] F. Maggi. Are the con artists back? a preliminary analysis of modern phone frauds. In *Computer and Information Technology (CIT), IEEE 10th International Conference on*, 2010.

[31] S. News. Losses from telephone banking fraud rise 95 percent. http://news.sky.com/story/1562860/losses-from-telephone-banking-fraud-rise-95-percent, October 2015.

[32] E. Pais. The premium-rate text-messaging scam worth 5 million euros. http://elpais.com/elpais/2015/04/20/inenglish/1429529298_001329.html, April 2015.

[33] S. Schrittwieser, P. Frühwirt, P. Kieseberg, M. Leithner, M. Mulazzani, M. Huber, and E. R. Weippl. Guess Who's Texting You? Evaluating the Security of Smartphone Messaging Applications. In *19th Annual Network and Distributed System Security Symposium, NDSS 2012*.

[34] W. Staff. Biggest phone scam in irs history continues to grow as tax season approaches. http://wiat.com/2016/01/20/biggest-phone-scam-in-irs-history-continues-to-grow-as-tax-season-approaches/, January 2016.

[35] v3.co.uk. Instant messaging to overtake email as biggest digital communication platform. http://www.v3.co.uk/v3-uk/news/2416558/instant-messaging-to-overtake-email-as-biggest-digital-communication-platform, July 2015.

[36] Y.-S. Wu, S. Bagchi, N. Singh, and R. Wita. Spam detection in voice-over-ip calls through semi-supervised clustering. In *Dependable Systems & Networks, 2009. DSN'09.*, pages 307–316. IEEE, 2009.

[37] E. Zheleva and L. Getoor. Privacy in social networks: A survey. In *Social network data analytics*, pages 277–306. Springer, 2011.