Towards Measuring the Effectiveness of Telephony Blacklists

Sharbani Pandit
Georgia Institute of Technology
pandit@gatech.edu

Roberto Perdisci
University of Georgia
Georgia Institute of Technology
perdisci@cs.uga.edu

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

Payas Gupta
Pindrop
pgupta@pindrop.com

***Abstract*—The convergence of telephony with the Internet has led to numerous new attacks that make use of phone calls to defraud victims. In response to the increasing number of unwanted or fraudulent phone calls, a number of call blocking applications have appeared on smartphone app stores, including a recent update to the default Android phone app that alerts users of suspected spam calls. However, little is known about the methods used by these apps to identify malicious numbers, and how effective these methods are in practice.**

In this paper, we are the first to systematically investigate multiple data sources that may be leveraged to automatically learn phone blacklists, and to explore the potential effectiveness of such blacklists by measuring their ability to block future unwanted phone calls. Specifically, we consider four different data sources: user-reported call complaints submitted to the Federal Trade Commission (FTC), complaints collected via crowd-sourced efforts (e.g., 800notes.com), call detail records (CDR) from a large telephony honeypot [1], and honeypot-based phone call audio recordings. Overall, our results show that phone blacklists are capable of blocking a significant fraction of future unwanted calls (e.g., more than 55%). Also, they have a very low false positive rate of only 0.01% for phone numbers of legitimate businesses. We also propose an unsupervised learning method to identify prevalent spam campaigns from different data sources, and show how effective blacklists may be as a defense against such campaigns.

I. Introduction

Telephony used to be a relatively closed and trusted system. However, with its convergence with the Internet, cyber criminals are now using the telephony channel to craft new attacks [14]. Robocalling [13], voice phishing [25], [39] and caller-id spoofing [15] are some of the techniques that are used by fraudsters and criminals in these attacks. The number of scam/spam calls people receive are increasing every day. In the United States, more than 660,000 online complaints regarding unsolicited calls were reported in 2015 on websites that track phone abuse[18], and the Federal Trade Commission (FTC) phone complaint portal receives millions of complaints about such fraudulent calls each year [13]. In several scams that have been reported widely, the telephony channel is either directly used to reach potential victims or as a way to monetize scams that are advertised online, as in the case of tech support scams [32].

In response to the increasing number of unwanted or fraudulent phone calls, a number of call blocking applications have appeared on smartphone app stores, some of which are used by hundreds of millions of users (e.g., [41], [42]). Additionally, a recent update to the default Android phone app alerts users of suspected spam calls [40]. However, little is known about the methods used by these apps to identify malicious numbers and how accurate and effective these methods are in practice.

To fill this knowledge gap, in this paper we systematically investigate multiple data sources that may be leveraged to automatically learn phone blacklists, and explore the potential effectiveness of such blacklists by measuring their ability to block future unwanted phone calls. Specifically, we consider four different data sources: user-reported call complaints submitted to the Federal Trade Commission (FTC) [12], complaints collected via crowd-sourced efforts, such as 800notes.com and MrNumber.com, call detail records (CDR) from a telephony honeypot [1] and honeypot-based phone call audio recordings. To the best of our knowledge, we are the first to provide a detailed analysis of how such data sources could be used to automatically learn phone blacklists, measure the extent to which these different data sources overlap, explore the utility of call context for identifying spam campaigns, and evaluate the effectiveness of these blacklists in terms of unwanted call blocking rates.

In performing this study, we are faced with a number of challenges, which we discuss in detail throughout the paper. First, the data sources may contain noise. For instance, user-provided reports are often very short, written in a hurry (using abbreviations, bad grammar, etc.) and may contain incomplete or incorrect information, making it challenging to automatically infer the context of spam/scam calls. In addition, some data sources provide very limited information. For instance, due to privacy concerns, the user-reported FTC complaints are anonymized, and only report the time at which each complaint was submitted and the phone number the user complained about; the content or description of the complaints are not available to the public. Partial call context can be obtained from transcripts of recordings of calls made to honeypot numbers. However, recording calls faces legal hurdles, can be costly, and the quality of the recorded content may depend on the extent of caller engagement.

Because the utility of phone blacklists comes from blocking calls which are based on the calling phone number, another

Network and Distributed Systems Security (NDSS) Symposium 2018
18-21 February 2018, San Diego, CA, USA
ISBN 1-1891562-49-5
http://dx.doi.org/10.14722/ndss.2018.23243
www.ndss-symposium.org

important challenge is represented by caller ID spoofing. If spoofing was pervasive, the effectiveness of blacklists could be entirely compromised. Somewhat surprisingly, our measurements instead reveal that phone blacklists can currently be reasonably effective, thus suggesting that perhaps caller ID spoofing is not as widely used as one may think or expect. Certainly, this could be due to the fact that few countermeasures currently exist for blocking unwanted calls, and therefore there may be few incentives for spammers to spoof their phone numbers. It may therefore be possible that once phone blacklists become more widely adopted, caller ID spoofing will also increase, thus making blacklisting less effective. Fortunately, the FCC has recently requested that carriers work towards a solution [57]. Technical proposals have been published on caller ID authentication [64], [58], [63], and efforts have been put in place by telecommunications providers to make caller ID spoofing harder [35]. For instance, the industry-led "Strike Force" effort [56] has suggested numerous steps that can help mitigate spoofing (e.g., carriers can choose to not complete calls that originate from unassigned phone numbers). Also, removing the caller ID altogether can be detrimental for attackers, as users are less likely to answer calls coming from a "private" number. Because the cost of acquiring ever new, valid phone numbers is non-negligible, the effectiveness of phone blacklists could increase significantly, once these anti-spoofing mechanism are more widely deployed. Therefore, studying how phone blacklists can be automatically learned, and evaluating their effectiveness, is important for both current and future telephony security applications.

In summary, we provide the following contributions:

- We present the first systematic study on estimating the effectiveness of phone blacklists. We first analyze the characteristics of multiple data sources that may be leveraged to automatically learn phone blacklists, and then measure their ability to block future unwanted phone calls.
- We investigate a number of alternative approaches for building phone blacklists. In particular, we propose methods for learning a blacklist when call context (e.g., complaint description or phone call transcripts) is available, and when context is missing.
- We evaluate the effectiveness of the phone blacklists we were able to learn, and show that they are capable of blocking a significant fraction of future unwanted calls (e.g., more than 55% of unsolicited calls). Also, they have a very low false positive rate of only 0.01% for phone numbers of legitimate businesses.
- To link phone numbers that are part of long running spam campaigns, we apply a combination of unsupervised learning techniques on both user-reported complaints as well as from phone call audio recordings. We then identify the top campaigns from each data source, and show how effective blacklists could be as a defense against such campaigns.

II. DATA COLLECTION

A. Data Sources

Although commercial phone blacklisting services and apps do not openly reveal how their blacklists are constructed, some of the data sources they use to derive the blacklists are known or can be inferred. For instance, Youmail [42] appears to leverage user complaints submitted to the FTC[1], whereas Baidu.com [55] leverages online user complaints[2]. Other telephony security companies, such as Pindrop [43], leverage phone honeypot data [1], [5].

To estimate the effectiveness of phone blacklists, we therefore use a multi-source data-driven approach that aims to gather and analyze datasets that are similar to the ones collected by commercial phone security services. Specifically, we consider two main sources of telephony abuse information: (i) phone call records collected at a large phone honeypot, and (ii) voluntary user complaints. For each of these information sources, we assemble two different datasets (described below), which we divide into *context-rich* and *context-less* datasets. We say that a phone call record is *context-rich* if a recording or description of the *content* (i.e., the actual conversation that took place) of the phone call is available, along with metadata such as the caller ID, the time of the call, etc. Conversely, when only the metadata (i.e., no content) related to the phone call is available, we refer to the phone call record as *context-less*.

It is important to notice that, because we harvest phone call information from honeypots and user complaints, our datasets naturally contain only records linked to *abuse-related, unwanted, or otherwise unsolicited phone calls* (though a small amount of noise may be present, as discussed below).

B. Context-Less Phone Abuse Data

FTC dataset (FTC) - The Federal Trade Commission collects voluntarily provided user complaints about unwanted or abusive phone calls (e.g., robocalls, phone scams, etc.). Along with the reported call metadata, users can include a description of the related phone conversation. However, the data publicly released[3] by the FTC only contains the source phone number and the timestamp of the reported call, and does not provide any information about the destination number (i.e., the user's phone number) or the content of the call itself, due to privacy reasons. From February to June 2016, we collected around 1.56 million complaints regarding 300,000 different source phone numbers.

Honeypot call detail records (CDR) - The CDR dataset contains detailed information about the calls coming into a telephony honeypot. It records the source phone number that made the call, the destination to which the call was made, and the time of the call. However, it does not provide the context of the call. This dataset contains records for more than one million calls received between February and June 2016 from approximately 200,000 distinct source numbers to approximately 58,000 distinct destination numbers, which are owned by the honeypot operator.

C. Context-Rich Phone Abuse Data

Crowd-sourced online complaints (COC). This dataset contains the online comments obtained from popular online forums, such as 800notes.com, MrNumber.com, etc., between Dec 1, 2015 and May 20, 2016. However, we only considered comments made between February to June so that this period overlaps with the time frame of the honeypot transcripts

[1] The results of a reverse phone look-up via youmail.com include the number of FTC and FCC reports related to the queried number.

[2] For example, searching for Chinese spam-related phone numbers on baidu.com will return a brief report that includes the number of users who have complained about the queried number.

[3] Via a Freedom of Information Act request.

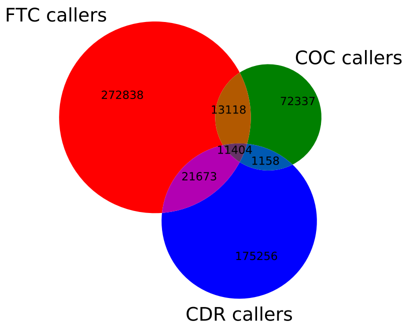


Fig. 1: Overlap of callers across data sources.

dataset (described below). The dataset contains about 600,000 actual "raw" complaints filed by users, containing the phone number that made the unwanted call, a timestamp, and a description of the content of the call. Since the comments are entered manually by frustrated users, the text describing the content of the call is typically quite noisy, as it contains many misspellings, grammatically inaccurate sentences, expletives, and in some cases apparently irrelevant information. It is also possible that the phone number and timestamp provided by the users could be mistyped.

Honeypot call transcripts (HCT) - This dataset contains about 19,090 audio recordings from 9,434 distinct phone numbers, extracted from a subset of calls made to the honeypot from February 17, 2016 to May 31, 2016. When a call is selected for recording, the responder software attempts to emulate human behavior, to elicit a short conversation from the caller, which may allow for inferring the nature of the call. However, we noticed that in many cases the conversation attempt by the honeypot responder is irrelevant, in that the caller often simply plays a recorded message (i.e., a *robocall*). The audio recordings were automatically transcribed using Kaldi [19]. Each dataset entry contains the time of the call, the source phone number, the destination phone number and the transcript of the call.

D. Data Volume and Overlaps

In the following, we present a number of high-level insights that can be gained from the four datasets described earlier. These insights help us understand how each dataset contributes to intelligence about telephony abuse, and what data source may first observe certain types of abuse.

Figure 1 depicts the overlap observed among callers across the FTC, CDR, COC datasets. Notice that, by definition, the source numbers in the HCT dataset are a small subset of the CDR dataset (see Section II-C). Interestingly, we found that many phone numbers that call into the honeypot are never seen in the COC or FTC datasets. We suspect this may be due to the fact that many of the honeypot phone numbers were previously business-owned, which were returned and repurposed. Hence, the scammers/spammers targeting businesses tend to be captured more frequently, whereas spam targeting individual users is less commonly observed by the honeypot. This hypothesis is supported by a manual analysis of the transcripts obtained from the HCT dataset, which revealed the prevalence of business-oriented abuse. At the same time, since complaints collected by the FTC and COC datasets come from individuals, they tend to mostly reflect scammers/spammers targeting generic users (more details are provided in Section IV-C1).

Figure 2 reports the data volume (i.e., the number of calls or complaints) over time, across the four datasets. The periodic drops are due to lower call volumes during weekends.The drop in the HCT traffic between April and May is because call recordings were stopped due to operational issues during that particular period. Similarly, operational issues affected the collection of COC data towards the end of May.

Figure 3 shows that a large fraction of source numbers receive only one or few complaints, or perform only few honeypot calls. This may be due to a combination of factors, including caller ID spoofing and noise due to misdialing, with spoofing being the most likely and prevalent culprit.

Figure 4 shows the difference in days between when the honeypot received the first call from a given source phone number and the time when the first complaint about that same phone number appeared in the COC dataset. Among the phone numbers that are observed in both the CDR and COC datasets, 20% of them were seen on the same day, whereas 35% of of the numbers were seen in the honeypot before they were complained about by users.

E. Gathering Ground Truth

Ideally, the datasets we collected should be free of noise. Indeed, both the honeypot records and the voluntary user complaints are by nature related to abusive or unwanted calls. However, as mentioned earlier, the datasets may contain noise, for instance due to misdialed calls to the honeypot or mistyped phone numbers in the complaint reports.

Establishing the true nature of source phone numbers that appear in our datasets is challenging, as no single authoritative entity exists that can certify whether a certain phone number is being used for legitimate or malicious activities. We, therefore, chose to take a conservative, best effort approach for ground truth collection based on multiple third party providers. Specifically, we leverage reverse phone lookup services provided by Whitepages [29], YouMail [42], and TrueCaller [41], to obtain independent insights about the nature of a fraction of the phone numbers we observed.

Query results from the above third parties contain information on whether a number is believed to be a spam/scam-related number. While we have no detailed information about how these third-party services classify phone numbers, public documentation suggests that they leverage user complaints, along with other data points. As these are third-party commercial systems with a large user base (millions of mobile application downloads), we believe it is reasonable to assume that they have checks in place to limit false positives to a minimum, because high false positives may otherwise deter app/service adoption and revenue. Therefore, if a source number is reported by these services as *spam*, we consider the label to be correct (unless disputed via other means).

Whitepages additionally provides information such as whether a phone number is likely a VOIP, toll free, mobile or landline number; it also indicates whether the number is used for commercial purposes, and provides owner information such as name, street address, etc., when available.

In addition to information on phone numbers likely involved in spam activities, we also collect a large set of phone numbers that can be considered as legitimate (i.e., non-

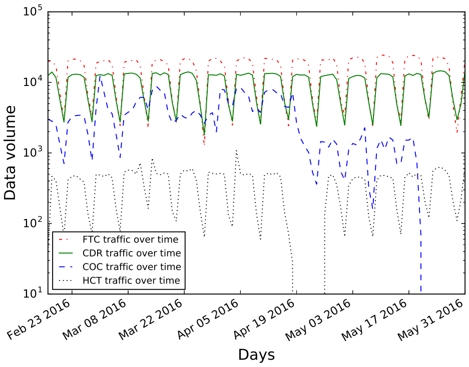


Fig. 2: Data volume over time

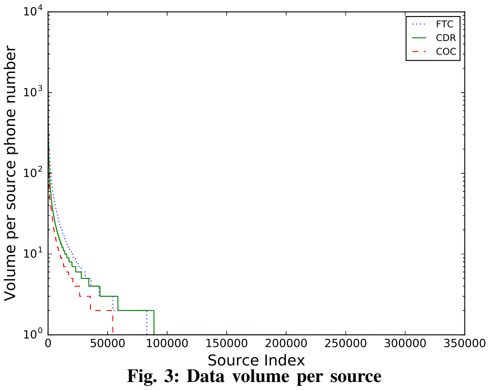


Fig. 3: Data volume per source

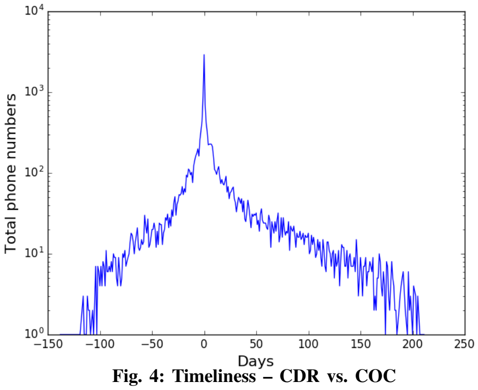


Fig. 4: Timeliness – CDR vs. COC

spam), by crawling the YellowPages.com phone book. We later leverage this set of phone numbers to estimate the false positive rate of our phone blacklists, as explained in Section IV-F.

In addition to collecting ground truth from third-parties, in some cases we attempted to verify the nature of phone numbers that are candidates for blacklisting by calling them back. For instance, for blacklisted numbers for which context is available (e.g., for numbers related to call transcripts), calling back allows us to verify whether the content of our call is similar to the previously recorded context.

F. Ethics

The telephony honeypot is operated by a commercial entity, and raw CDR data was accessed under non-disclosure agreements. The honeypot is programmed to record only (a small subset of) phone calls that meet rigorous legal requirements, according to US federal and state laws.

The FTC data was obtained in response to a freedom of information act (FOIA) request, and does not contain any sensitive information. For instance, the FTC does not disclose destination phone numbers and user complaint descriptions, to protect the privacy of the reporting users.

The ground truth data collected from third-party sources, such as YouMail, TrueCaller, and Whitepages, is limited to publicly accessible information. To increase the number of available queries, we used the Whitepages Premium service. For all Whitepages reverse phone lookups, we carefully refrained from collecting sensitive information from background reports (i.e., we never analyzed or stored any information about bankruptcies, liens, arrest records, family members, etc., which is available from Whitepages Premium).

When calling back numbers that are candidate for blacklisting, we only called those that asked to be called back, according to the honeypot transcripts in the HCT dataset. Furthermore, when calling back we never interacted with a real human. Every call we made went through an automated interactive voice response (IVR) system.

We did not involve human subjects in our research. The honeypot calls were recorded by a third-party company, while abiding by US laws (e.g., single-party consent requirement). Calls made by us were limited to interactions with automated IVR systems. Because of these reasons, we did not seek explicit IRB approval.

III. Phone Blacklisting

Blacklisting has been extensively studied as a way to defend against Internet abuse [4], [27], [48], [49]. For instance, domain name, URL, and IP address blacklists are commonly used to defend against email spam, phishing, and malware infections [47], [45], [46], [4]. Recently, phone blacklisting has started to make its way into real-world applications [42], [41]. However, to the best of our knowledge, the effectiveness of blacklisting approaches to defend against abuse in the telephony domain has not yet been systematically studied. Therefore, in this paper we focus on measuring the effectiveness of blacklists for blocking unwanted phone calls, as described below.

A. Example Use Case

We consider a scenario in which smartphone users[4] install an app that implements the following functionalities: the app is notified every time a phone call is received, it checks the caller ID against a phone blacklisting service[5], and informs the user whether the calling phone number is believed to be used for phone spam/abuse activities. This use case is similar to currently available popular apps [42], [41], [40].

Depending on user preferences, the app may strictly enforce the blacklist, and immediately block the call [44] (while still notifying the user of the event, for example). Alternatively,

[4] Blacklisting services may also be used by telephone networks to defend landline phones, for instance. While in the past there existed strict regulatory constraints that may have prevented carriers from using blacklists to block calls, such restrictions seem to have been recently relaxed [44].

[5] We also assume that queries to the blacklisting services can be done securely and in a privacy-preserving way, similarly to URL blacklists such as Google Safebrowsing.

the user may opt for a *soft blacklisting* enforcement, whereby the user is provided with information about if/why the calling number was included in the blacklist and will have to decide whether to pick up the call or not [40]. For instance, the user may be informed that the calling number was previously complained about by other users (e.g., via the FTC complaints service). If context is available (see Section II-C), the app may also provide information about a specific (set of) spam campaign in which the number has been involved.

In absence of public details on how current phone security apps work internally, in this section we explore different approaches for building a phone blacklist. Specifically, we build five different blacklists using the four datasets described in Section II, and then evaluate their effectiveness in blocking future unwanted or abusive calls in Section IV. Our blacklisting system architecture is shown in Figure 5. As we will show in Section IV, we empirically verified that our blacklists resemble third-party blacklists, and can therefore be used as a proxy for evaluating the effectiveness of proprietary phone blacklists.

B. Context-less blacklisting

As discussed in Section II, the FTC and CDR datasets do not include the context of a call. To build a blacklist based on these context-less data, we therefore focus on identifying anomalies in calling patterns.

Before we describe our blacklisting approach, it is important to remember that, by nature, the FTC and CDR datasets contain only information about unwanted or abusive calls. While we cannot exclude the presence of small amounts of noise (e.g., due to misdialed calls captured by the honeypot, or numbers incorrectly reported to the FTC), it is reasonable to assume the fraction of noisy reports/calls is small. We leverage this as the main observation to guide our approach to phone blacklisting in absence of context.

1) Blacklisting using the CDR data: Because the CDR dataset may (by chance) collect misdialed phone calls, we first apply a pre-filter step by removing phone numbers that, during the training data collection period, made less than θ_c calls to less than θ_d destination honeypot numbers. In other words, we only consider a phone number for the next processing steps if it made more than θ_c calls and contacted more than θ_d different destinations within a predetermined observation time (in our experiments, we primarily use $\theta_c = 5$ and $\theta_d = 3$, but also perform additional experiments that show how the blacklist effectiveness varies with these parameters). Notice that this pre-filtering step is fairly conservative, and that source phone numbers actively involved in spam activities will tend to pass this simple filter.

To build the CDR-based blacklist, we analyze the behavior of the remaining source phone numbers. For each of the source phone numbers, p_i, we compute a *blacklist score*:

$$s(p_i, \Delta t) = \alpha \times vol(p_i, \Delta t) + \beta \times nod(p_i, \Delta t) \qquad (1)$$

where $vol(p_i, \Delta t)$ is the number of calls made by p_i within time Δt, whereas $nod(p_i, \Delta t)$ is the number of destination numbers called by p_i in the same time period, and α and β are tunable parameters.

As spammers typically tend to reach a large number of potential victims, we set the value of β greater than α (in our experiments, we set β=0.2 and α=0.1). Any number p_i whose blacklist score $s(p_i, \Delta t)$ is greater than a threshold θ_b, which is learned from past observations, is added to the blacklist.

To learn the blacklist, we use a *one-class* learning approach [52], [51]. This choice of learning paradigm is guided by the challenges in collecting ground truth labels (see Section II-E), especially for benign phone numbers. To identify spam-related phone numbers within the CDR dataset, which we then leverage for training the blacklisting threshold, we proceeded as follows. Given the set $\mathcal{P}_{CDR}$ of all source phone numbers calling into the honeypot (excluding the pre-filtered numbers), we find the intersection between these numbers and the phone numbers reported in the FTC and COC datasets during the observation period Δt. Because these datasets are collected in a completely independent way (honeypot calls vs. user complaints), we assume that phone numbers that appear in two or more datasets are the most likely to actually be spam-related. For instance, if a number p_j called into the honeypot multiple times (enough to pass the pre-filter), and multiple users, in a completely independent way, complained about the same p_j number via the FTC portal, we label p_j as *spam*. We then use this *one-class* labeled subset of spam numbers, $P_s \in (\mathcal{P}_{CDR} \cap \mathcal{P}_{FTC})$, to learn the θ_b threshold. Specifically, we sort the number in P_s by their blacklist score (see Equation 1), and set θ_b so that the top 99% of all numbers, by score value, are added to the blacklist. In the spirit of one-class learning [52], [51], the remaining 1% of numbers are considered to be *tolerable* false negatives, and have the benefit of making the decision threshold sufficiently "tight" around the bulk of spam-labeled data to filter out the possible remaining dataset noise (i.e., potentially benign numbers that accidentally called into the honeypot).

2) Blacklisting using the FTC dataset: The FTC dataset is the largest in terms of volume of reported phone numbers, compared to the other datasets. As mentioned in Section II, the information provided by the FTC is very limited, as it only contains the user-reported phone numbers and a timestamp for each complaint report. Unlike the CDR dataset, no information is provided regarding the destination numbers.

Like the CDR dataset, the FTC dataset may also contain small amounts of noise, for instance due to a calling number being typed erroneously into a user complaint. To filter out this possible noise, we exclude all phone numbers that have been reported in less than θ_c complaints (notice that this parameter is similar to the θ_c filtering threshold used for the CDR-based blacklisting). All remaining numbers are then simply added to the blacklist. The reason why we do not perform any additional filtering is that the FTC dataset contains official complaints that users send to the FTC; as such, this dataset intuitively tends to contain lower amounts of noise, compared to the CDR dataset.

3) Context-less blacklisting using the COC dataset: For comparison purposes, we apply the same process described above for the FTC dataset to the COC dataset, pretending that no context is available. In other words, from the user complaints in the COC dataset we only extract the timestamp and the reported phone number (i.e., the source numbers users complained about), and apply the filtering approach described above for the FTC complaints. In the remainder of the paper, we refer to this blacklist as the COCNC blacklist, where NC stands for *no-context*.

C. Context-rich blacklisting

The context of a call, when available, can be used to understand the nature and content of the conversation, and provide more definitive indication on whether a call is potentially an

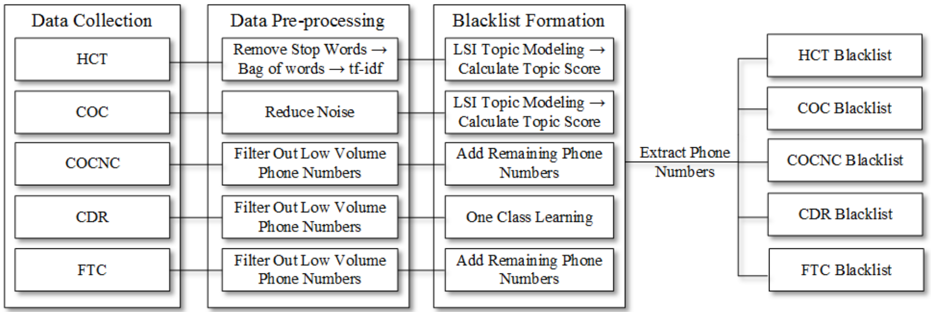

Data Collection
HCT
COC
COCNC
CDR
FTC
Data Pre-processing
Remove Stop Words → Bag of words → tf-idf
Reduce Noise
Filter Out Low Volume Phone Numbers
Filter Out Low Volume Phone Numbers
Filter Out Low Volume Phone Numbers
Blacklist Formation
LSI Topic Modeling → Calculate Topic Score
LSI Topic Modeling → Calculate Topic Score
Add Remaining Phone Numbers
One Class Learning
Add Remaining Phone Numbers
Extract Phone Numbers
HCT Blacklist
COC Blacklist
COCNC Blacklist
CDR Blacklist
FTC Blacklist

Fig. 5: System Architecture

unwanted or fraudulent one. For example, calls with similar context can be clustered together to discover spam campaigns, and the phone numbers related to the campaigns can be added to the blacklist, along with contextual information. Therefore, when a user receives a call from a number that belongs to a context-rich blacklist, we could not only inform the user that the incoming call is likely unwanted or abusive, but also provide a short description (e.g., via the example app described in Section III-A) of what kind of spam campaigns the number has been involved within the past. This information may be particularly useful when a *soft blacklisting* approach is selected, as it may help the user make a decision on whether to pick up the call or not.

1) Blacklisting using the HCT dataset: To derive a phone blacklist based on the honeypot call transcripts (HCT) dataset, we take the following high-level steps: (i) we perform transcript text analysis using topic modeling via latent semantic indexing (LSI) [6], to extract possible campaign topics; (ii) we then label transcripts based on their most similar topic, and group together calls that likely belong to a common spam campaign; (iii) finally, phone numbers belonging to a spam campaign are added to the blacklist. Below, we provide more details on this blacklisting process.

We first use a data pre-processing phase, which aims to filter out possible noise from the transcripts (e.g., noise due to imprecise speech transcription). To this end, we use the following steps: (1) *stop-words* are removed and a dictionary of the remaining terms is extracted from the transcripts' text; (2) each transcript is then converted into a bag of words, and each word is assigned a score using TF-IDF [10]. These two steps transform each call transcript into a vector of numerical features (i.e., a feature vector).

TABLE I: LSI topic modeling on HCT – top 10 topics

topic 0	google, listing, front, page, business, verify, press, removed, searching, locally
topic 1	cruise, survey, bahamas, awarded, correctly, included, participate, short, congratulation, selected
topic 2	listing, verify, front, google, page, updated, record, show, end, list
topic 3	verification, address, name, phone, number, cancel, flagged, map, notice, business
topic 4	hotel, pressed, exclusive, telephone, husband, marriott, detail, announcement, pre, star
topic 5	hotel, exclusive, husband, marriott, star, stay, placed, complimentary, further, telephone
topic 6	electricity, bill, per, system, stop, increase, energy, renewable, soon, coming
topic 7	optimize, found, date, order, indicate, critical, online, updated, show, end
topic 8	system, interest, eligibility, cost, account, rate, credit, notice, card, lower
topic 9	business, interest, eligibility, thousand, application, loan, rate, bad, system, qualifies

We then use a topic modeling approach on the feature vectors obtained from the steps mentioned above. Let Δt be a data observation window, and $\mathcal{H}(\Delta t)$ be the set of call transcript feature vectors obtained during Δ_t. We use LSI, a natural language processing technique that leverages SVD [11] to map documents (i.e., transcripts, in our case) from a syntactic space (the bag of words) to a lower-dimensional semantic space represented by a (tunable) number τ_{hct} of *topics*. In concrete terms, each topic is represented by a set of representative keywords that may be interpreted as describing a campaign theme. Table I shows the top 10 topics (sorted by eigenvalue) extracted from our HCT dataset (more details on the experimental results are provided in Section IV).

At this point, each transcript can be represented as a weighted[6] mix of topics, rather than a set of words [6]. Among these, we can identify the topics with the highest weight, which can be interpreted as indicating what spam campaigns the calling number recorded in the transcript is involved with.

The LSI algorithm requires as a parameter the desired number of topics to be kept in the SVD decomposition. Choosing the best value for the number of topics is often done either manually, by leveraging domain knowledge, or by measuring topic coherence [54]. However, coherence measures are themselves still a subject of research in the machine learning domain, and don't always lead to satisfactory results in practice. Therefore, in this paper we revert to manual selection driven by empirical results, and leave a fully automated selection of the optimal number of topics to future work. In our experiments, we first set the maximum number of LSI topics to 50. Then, once the topics are extracted, we manually analyze them and mark the ones whose keywords more clearly indicate a spam campaign, whereas the other topics are effectively discarded from a transcript's topic mixture vector. As a byproduct, this manual analysis also has the advantage that it allowed us to associate a human-interpretable campaign *theme* to each of the remaining topics. For instance, we summarize `topic 0` in Table I as the *Google Listings* spam campaign (notice that when analyzing a topic, not only we can refer to the topic's keywords, but also to the full text of the transcripts that are associated with that topic with a high weight).

At this point, we have a set of topics, $\mathcal{T}$, that are labeled with a relevant campaign theme, and we aim to do two things: (1) decide what source numbers for the transcribed calls should be blacklisted; and (2) leverage the topic model to group together call transcripts, and related source phone numbers, that likely belong to the same spam campaign. To this end, we first compute a *topic similarly* score $S_{i,j} = S(tr_i, \tau_j)$ that indicates how strongly a transcript tr_i is associated to each topic $\tau_j \in \mathcal{T}$. We calculate the topic scores by nor-

[6] We consider absolute values for the topic weights.

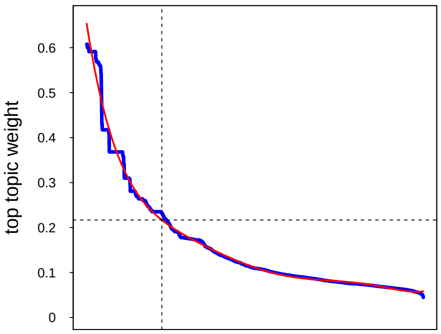


Fig. 6: HCT blacklist score threshold learning

malizing the topic weights output by the LSI topic modeling algorithm. Specifically, for every transcript tr_i and every topic τ_j, the topic modeling algorithm will assign a weight $w_{i,j} = w(tr_i, \tau_j)$. We compute the normalized weights for each transcript as $w'_{i,j} = |w_{i,j}|/\sum_j |w_{i,j}|$, and set the score $S_{i,j} = w'_{i,j} \in [0,..,1]$.

To decide whether a source phone number p_i responsible for call transcript tr_i should be blacklisted, we proceed as follows. We first compute the topic most similar to tr_i, namely $k = \arg\max_j(S_{i,j})$. Then, if $S_{i,k}$ is greater than a predetermined threshold θ_k, we assign p_i to the HCT blacklist. The threshold θ_k is learned as follows. Let S_i^* be the highest topic score for transcript tr_i. We first plot the distribution of scores S_i^* computed over all transcripts, as shown in Figure 6 and then compute θ_k by finding the "knee" of the curve (the knee finding process is explained in details in Appendix A).

Now, for every blacklisted number p_i, we have the topics that are most similar to the transcripts related to p_i, and can therefore label p_i with one or more campaigns themes (an analysis of campaigns themes and related blacklisted numbers is reported in Section IV-G).

2) Blacklisting using the COC dataset: Like honeypot call transcripts, user comments from online forums such as 800notes.com, MrNumber.com, etc., also provide us with the context of an unwanted call. However, transcripts and user comments datasets are different in nature, as user comments only provide a user's version – a subjective textual description – of the content of a call. To derive a blacklist using the COC dataset, we follow a process very similar to the one we used for the HCT data, with some small changes that take into account differences in the nature of the two data sources.

Via manual analysis of a few initial samples of online complaints data, we noticed that user-provided descriptions of unwanted calls tend to be noisier in nature than transcripts from call recordings. This is fairly intuitive: while the transcripts faithfully reflect the conversation (often represented by a well-played recorded spam message), user complaints typically consist of high-level descriptions of a call, in which abbreviations, bad grammar, and expletives are used to express discontent. Therefore, to reduce noise we use a more stringent pre-processing step, compared to the HCT dataset. First, we only consider phone numbers that were complained about at least θ_{coc} times ($\theta_{coc} = 10$, in our experiments). We also remove stop words and punctuation from the comments, and combine all comments about a single phone number into a single text document. This latter aggregation step is motivated by the following considerations: (1) complaints from different users that receive calls from the same phone number are often similar, because the number is used to perpetrate the same spam campaign by calling multiple destinations; (2) online complaints are often very short, making it difficult to automatically analyze them independently from each other; (3) by aggregating multiple complaints, we obtain larger documents that can be more easily analyzed using topic modeling, with a process similar to the one described in Section III-C1.

Let $\mathcal{C}(\Delta t) = \{c_{s_1}, \ldots, c_{s_n}\}$ be the set of complaint documents, where document c_{s_i} is an aggregate (i.e., a concatenation) of all user complaints about calling number s_i observed during period Δt. As for the HCT blacklist, we apply LSI on $\mathcal{C}(\Delta t)$ to derive a set of possible spam campaign themes, and at the same time associate each calling number (via the related complaints) to a mix of topics, as done for the HCT blacklist. We then decide what source numbers for the transcribed calls should be blacklisted by computing the topic scores, plotting the distribution of the maximum score, and computing the blacklisting score threshold by finding the "knee" of this distribution (see Section III-C1).

Although both context-rich and context-less blacklists consist of phone numbers, each phone number in the context-rich blacklist is associated with a label derived from its context. The label identifies the type of scam campaign for which the phone number was used. This label could be used to inform a user about the potential content of a suspected spam call.

IV. Evaluation

In this section, we evaluate the effectiveness of the phone blacklists we constructed following the methods described in Section III. We first show that our blacklists are representative of real-world phone blacklisting applications, by vetting them against two commercial third-party telephony security services. In addition, we perform an analysis of how blacklists formed from different data sources can complement each other. We then assess how well the blacklists we construct can help to block future spam (i.e., unwanted or abusive) phone calls, by evaluating each blacklist in terms of their *call blocking rate* (defined below). Finally, we analyze a few prominent phone spam campaigns, and demonstrate how effective the blacklists would be in blocking these campaigns.

A. Call blocking rate (CBR) definition

We evaluate the effectiveness of a phone blacklist based on its ability to block future unwanted call. Therefore, to enable the evaluation we first need to more formally define the concept of blocking rate. Given a blacklist $\mathbb{B}(\mathcal{D}, \Delta t) = \{p_1, \ldots, p_m\}$ containing m phone numbers learned from dataset $\mathcal{D}$ over a *training* observation period Δt, we consider the set $C(\lambda_t)$ of calls (or complaints, depending on the blacklist being analyzed) observed at a future *deployment* time period λ_t. We then compute the ratio $r(\mathbb{B}, \lambda_t) = N_{bl}(\lambda_t)/N_{tot}(\lambda_t)$ between the number of calls (or complaints) $N_{bl}(\lambda_t)$ that would have been blocked by the blacklist $\mathbb{B}$, and the total number of calls (or complaints) $N(\lambda_t) = |C(\lambda_t)|$ received during period λ_t.

In the remainder of this section, the set $C(\lambda_t)$ will represent either the set of calls received by the honeypot, as recorded in the CDR dataset, or user complaints from the FTC or COC datasets, depending on the blacklist that is being evaluated. In the first case, we refer to $r(\mathbb{B}, \lambda_t)$ as the *call blocking rate*, whereas in the second case we refer to it as the *complaint blocking rate* – both abbreviated as CBR.

In the case of the CDR data, we essentially pretend that the phone numbers belonging to the honeypot are owned by users, and consider a future honeypot call to be *blocked* if the related calling phone number was included in $\mathbb{B}(\Delta t)$. Therefore, the CBR estimates the fraction of future unwanted calls towards real users that would be prevented by the blacklist. In addition, in this case we can also measure how many users would be defended against spam calls, by counting the number of distinct destination numbers that thanks to the blacklist did not receive the unwanted calls.

In the case of the blacklists derived from the FTC and COC datasets, the CBR measures the fraction of future complaints that would be prevented by the blacklist. Computing the number of *blocked complaints* is motivated by this simple observation: if an app enforcing the blacklist was widely deployed, or telephone carriers directly blocked calls based on the blacklist, users would not receive any more unwanted calls from the blacklisted numbers, and would therefore stop complaining about them. Thus, the number of complaints that would be prevented (i.e., blocked) is a reflection of the effectiveness of the blacklist.

B. Experimental Setup

Our experiments for computing the CBR are performed as follow, for all the datasets and blacklisting approaches described in Sections II and III. Let $\mathcal{D}$ be a dataset of calls or complaint records (e.g., the CDR or FTC dataset). We start by setting an initial training period Δt_0 of one month, corresponding to the first month of data collected in $\mathcal{D}$. We use this first month of data to learn the first blacklist $\mathbb{B}_0 = \mathbb{B}(\mathcal{D}, \Delta t_0)$. We then consider one day, λ_{t_0}, of data from $\mathcal{D}$ collected on the day immediately after period Δt_0, and compute the CBR $r_0 = r(\mathbb{B}(\mathcal{D}, \Delta t_0), \lambda_{t_0})$.

We then set $\Delta t_1 = \Delta t_0 + \lambda_{t_0}$, thus extending the training period by one day, and compute $\mathbb{B}_1 = \mathbb{B}(\mathcal{D}, \Delta t_1)$ and $r_1 = r(\mathbb{B}(\mathcal{D}, \Delta t_1), \lambda_{t_1})$, where λ_{t_1} again represents the day after Δt_1. We repeat this process for all subsequent days of data available in $\mathcal{D}$, thus obtaining a series of blacklists $\{\mathbb{B}_i\}_{i=0..k}$ and related blocking rate estimates $\{r_i\}_{i=0..k}$. In other words, every day we extend our blacklist training set by adding one day of data from $\mathcal{D}$ to the previous training dataset, and then test the obtained blacklist against the following day of data in $\mathcal{D}$. This allows us to estimate how effective the blacklist would be in blocking future calls (or complaints) related to spam phone numbers we learned up to the previous day.

C. Characterizing Blacklisted Numbers

We now analyze the overlap among blacklists learned over different data sources, and discuss how our blacklists align with phone blacklists provided by third-party apps.

1) Overlap among our blacklists: Figure 7 shows the size (i.e., the number of blacklisted phone numbers) and overlap among the blacklists learned as discussed in Section III. These results are obtained by building the blacklists over the entire set of data available from each of our data sources. In other words, given a dataset $\mathcal{D}$, we consider the entire period of time Δt_{max} in which data was collected, and compute the related blacklist $\mathbb{B}(\mathcal{D}, \Delta t_{max})$.

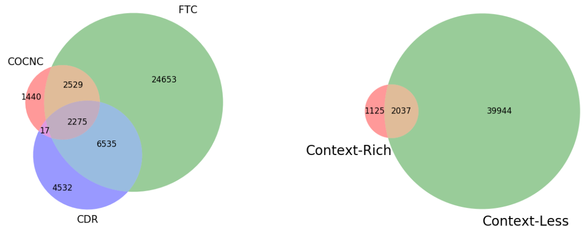


Fig. 7: Blacklists size and overlap

As we can see from the figure, the overlap among blacklists derived from different data sources is limited. Specifically, there exists only partial overlap between spam phone numbers observed in the three main data sources, namely the FTC complaints, online user complaints (i.e., the COCNC blacklist) and honeypot call records (i.e., the CDR blacklist). This shows that the different blacklists provide coverage for different sets of phone numbers, and are therefore complementary to each other.

The differences between honeypot calls and user complaints is likely due to the particular nature of the honeypot-owned phone numbers. While user complaints mostly reflect spam campaigns that target ordinary users, many of the honeypot-owned phone numbers were previously owned by businesses, and tend to attract business-oriented phone abuse. We verified this by analyzing the honeypot transcripts in the HCT dataset, many of which are related to *Google business listing*, *business loans*, and other business oriented phone spam campaigns. On the other hand, user complaints tend to reflect more "traditional" spam campaigns, including *IRS* scam, *tech support* scam, *payday loans* scams, etc. (see Section IV-G for more details about spam campaign analysis).

We conducted further analysis to better understand the limited overlap between the COCNC and CDR blacklists. There are 2,292 overlapping phone numbers in these two blacklists, and we found that these numbers are not among the most complained about or heavy honeypot callers. This seems to refute the intuition that phone numbers that make the most honeypot calls are also more likely to be complained about by users. Again, this may be due to the different, business-oriented nature of the honeypot-owned numbers, as discussed above.

The FTC is the largest dataset in our analysis, and the blacklist constructed from the FTC complaints is the largest one, in terms of number of blacklisted phone numbers. Comparing the FTC blacklist to all other four blacklists combined shows an overlap of less than 50%. On the other hand, the context-rich blacklists are significantly smaller than the context-less ones. The main reason is that in the context-rich case a phone number is added to the blacklist only if it can be associated to an identifiable spam campaign (see Section III-C).

2) Overlap with third-party blacklists: As discussed in Section II-E, we leverage third-party phone security and information services to gather independent (partial) ground truth on spam-related activities and characterize the phone numbers in our blacklists, as described below. We first assess the overlap between our blacklists and phone abuse information available from Youmail [42] and Truecaller [41], and then use the Whitepages [29] reverse phone lookup service to gather further insights into a subset of the numbers.

To estimate the overlap between our blacklists and third-party blacklists, we selected a random sample of 12,500 source phone numbers from all of our datasets, and performed reverse phone lookup queries. We found that 2.4% of the queried numbers were labeled as *spam* by Youmail. To determine the overlap between our blacklists and Youmail's, we proceeded as follows. If Youmail labeled a queried phone number as *spam*, we checked if the number was also present in our blacklists or not, and found that 87% of the phone numbers blacklisted by Youmail were also present in one or more of our blacklists. Most of the numbers labeled as *spam* by Youmail that were not included in our blacklists are present in our FTC dataset, but they were not included in our blacklist because they had a very low number of complains. This is in accordance with our attempt to be conservative, and filter-out possible noise in the user complaints, as explained in Section III-B2. On the other hand, it appears that Youmail labels a number as *spam* even if only one user complained to the FTC about that number. If we had added all FTC-complained callers to our blacklist, we would have a 98% match of our blacklisted numbers against Youmail's blacklist. We also found that among the numbers that were not labeled as spam by Youmail, about 1% of them were present in our blacklists. These results show that, combined, our blacklists are representative of a commercial blacklisting app such as Youmail.

To compare our blacklists with Truecaller, we took a similar approach. In this case, we found that 38% of the numbers we queried were labeled as *spam* by Truecaller, and that only 13% of all the numbers labeled as *spam* by Truecaller were contained in our blacklists. The reason is that Truecaller seems to be labeling a number as abusive even if only one Truecaller user reported it as such. In fact, by labeling as *spam* only numbers that have been reported as abusive to Truecaller by at least 5 users, we found that 75% of these numbers are present in our blacklists. As in the previous analysis of Youmail, we found that of the numbers that were not labeled as spam by Truecaller, only 13% were present in our blacklists. The majority of this 13% of numbers matches our FTC blacklist, and are therefore reported in multiple user complaints.

The above results confirm that our blacklisting approach aligns fairly well with real-world, commercial phone blacklists (especially with the Youmail app), and can therefore be leveraged as a proxy to estimate how effective third-party phone blacklists are in defending real users from unwanted or abusive calls.

3) Analysis of phone number types: To further characterize the phone numbers included in our blacklists, we turned to the Whitepages [29] reverse phone lookup service. Whitepages is a third-party provider that gathers comprehensive information about phone numbers, including detailed phone ownership information, and whether the phone number *type* falls within one of the following categories: *VoIP*, *toll-free*, *landline*, or *mobile*.

As Whitepages' public querying interface only allows for a limited number of queries, we first started by selecting a sample of 150 phone numbers in the overlapping region across HCT, COC and CDR blacklists, and analyzed their query results. Because these numbers appeared in three different blacklists, they are among the highest confidence spam numbers in our datasets. We found that 67% of these numbers are VoIP numbers for which no owner information was available. This is not surprising, as it is common for abusive calls to originate from VoIP numbers [33], [37]. The lack of owner information also suggests that these phone numbers are unlikely to belong to legitimate users. In addition, 5% of these numbers did not appear to have been assigned by any carrier. Surprisingly, only 22% of the numbers we queried were flagged as scam/spam callers by Whitepages itself. This suggests that Whitepages may be missing a large fraction of numbers that can be labeled as *spam* with high confidence.

TABLE II: Whitepages reverse phone lookup results

	cdr	coc	hct	hct/coc	hct/cdr	coc/cdr
VoIP	69%	37%	57%	80%	76%	70%
toll-free	9%	2%	0%	0%	2%	0%
landline	20%	16%	26%	20%	22%	30%
mobile	2%	45%	17%	0%	0%	0%
owner info	7%	10%	12%	2.5%	2%	5%

We then expanded our sample of phone numbers by randomly drawing a total of 400 numbers from all blacklists and performing an analysis of the reverse lookup results. Out of all the phone numbers we queried from Whitepages, 71% of them were present in our FTC blacklist. Table II summarizes some of the results we obtained, where the *cdr*, *coc*, and *hct* represent results related to phone numbers that were included only in the CDR, COC, or HCT blacklist, respectively. Columns *hct/coc*, *hct/cdr*, and *coc/cdr* represent a random sample of the phone numbers that belong to the intersection between pairs of blacklists. Table II also report the percentage of phone number for which ownership information was present. As we can see, only a relatively small fraction of the (potentially spoofed) numbers appear to be owned by a valid user or business. In addition, we found that some owner information (e.g., owner name and location) is highly likely forged.

D. Evaluating Context-Less Blacklists

We now evaluate the call (or complaint) blocking rates, which we defined in Section IV-A, for the CDR, FTC, and COCNC blacklists (see Section III-B).

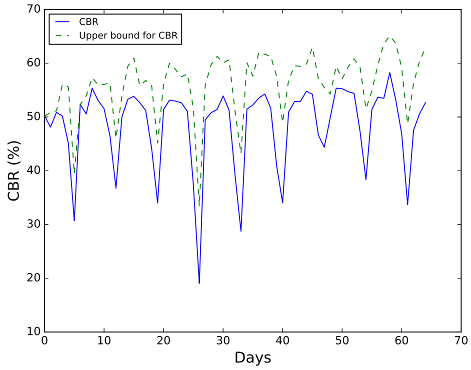


Fig. 8: CDR call blocking rate

Figure 8 shows the call blocking rate when applying the CDR blacklist to future calls into the honeypot. The y axis depicts the percentage of calls to the honeypot on a given day that would have been blocked by using the most recent blacklist (i.e., the blacklist trained on data collected up to the previous day, as detailed in Section IV-B). The blue curve show the results for our CDR blacklist (see Section III-B1), whereas the green line shows the upper bound for the effectiveness of the blacklist. This upper bound was obtained by adding *all* source phone numbers previously observed in the honeypot (during the training period Δt) to the blacklist, without applying any noise filtering or threshold learning, and then testing against the calls received on the next day (i.e., λ_t). The periodic call blocking rate drops in this graph are due to periodic appearance of new callers (every 7 days or so).

Although the blocking rate decreases during some days, by updating (i.e., re-training) the blacklist daily, over 55% of honeypot calls could be blocked using our CDR blacklist, on average. This shows that the blacklist can be fairly effective in blocking unwanted calls. At the same time, the upper bound curve clearly demonstrates that every day 40% or more of all calling numbers observed from the honeypot are always new (never seen before). Although we need to consider that noise (e.g., misdialings) may be recoded in the dataset and that spammers also churn through new valid numbers, it is likely that the large volume of new numbers is for the most part due to spoofing.

Figure 9 reports the complaints blocking rate (defined in Section IV-A) for the FTC blacklist. These results were computed for three different values of the θ_c blacklisting threshold defined in Section III-B2. Naturally, the lower the threshold, the higher the blocking rate, because more numbers will be added to the blacklist. In Figure 10, we report analogous results for the COCNC blacklist (Section III-B3), including results for $\theta_c = 1$, which provide an upper bound for the effectiveness of the blacklist.

As we can see, from Figure 9, the CBR for the FTC blacklist is much higher than in the CDR and COCNC cases. On possible explanation for this is that that users may be reluctant to report an unwanted call to the FTC unless they receive multiple calls from the same number, given the official nature of the complaint to a government agency. In this case, they complaints would likely include more "stable" source phone numbers, and naturally filter most of the spoofed calls. Another factor to consider is that not all users will complain to the FTC; namely, for a number to appear into the FTC dataset, it is likely that several users received a call from the same number, but only one or few users decided to complain.

Unlike the FTC dataset, the CDR dataset includes all (unanswered) calls to the honeypot, even if a number called only one time to one of the many honeypot destination numbers. This explains the lower effectiveness of the blacklist shown in Figure 8. On the other hand, given the more "casual" nature of the online user complaints collected in the COC dataset, it is not surprising to see a CBR reaching between 50-60%, when setting $\theta_c = 5$ for the COCNC blacklist, as shown in Figure 10. However, towards the end of our data collection period, we see a large drop in the CBR, including in the upper bound (i.e., $\theta_c = 1$) case. This means that the vast majority of numbers in the complaints collected every day after the drop were never seen before. After manual investigation, we found that many of the new complaints with never-before-seen source numbers seemed to be related to an IRS scam. Considering that the drop started in the weeks before the US tax filing deadline of April 15, it is possible that the drop is caused by a new large IRS scam campaign that relies heavily on caller ID spoofing.

E. Evaluating Context-Rich Blacklists

Context-rich blacklists (see Section III-C) tend to be much more conservative, compared to context-less blacklists, as clearly shown in Figure 7 (Section IV-C1). Unlike the context-less case, only numbers that can be attributed to one or more human-identified spam campaigns are added to the blacklist. In addition, the HCT dataset only contains a small subset of the CDR data (i.e., only recorded calls). As expected, Figure 11 shows that the overall blocking rates for the HCT and COC blacklists are fairly low.

Consider that during training only a small fraction of the source phone numbers can be attributed to a distinguishable campaign. To see why this would be the case, let's consider the COC data source as an example. Figure 3 shows that a large number of source phone numbers are complained about only once or very few times and never again. This means that, in a particular day, many of the COC user complaints are related to numbers that were never seen before (and the will never be seen again). Since most user complaints contain only very short text, it is difficult to attribute these numbers to a campaign, and will therefore be excluded from the COC blacklist.

To see how effective the context-rich blacklists are at blocking specific spam campaigns, below we analyze two representative campaigns discovered as part of the HCT blacklist learning (see Section III-C1). Specifically, we explore the *Google Listings* and *Free Cruise* campaigns, and compute the CBR for calls from numbers that are assigned (via topic analysis) to these two campaigns, which are reported in Figures 12(a) and 12(b). In addition, Figures 12(a) and 12(b) also report the fraction of calling source numbers blocked and the fraction of destination numbers that are "protected" from the spam campaigns. We can notice that the CBR drops significantly on some days, when many new source phone numbers appeared that were never seen before. However, the blacklist adapts fairly quickly to the new sources, by including these numbers at the next daily blacklist update, thus increasing the campaign CBR. On average, the CBRs for the two campaigns were 70% and 84%, respectively. These results suggest that while the spammers running these campaigns (especially the Free Cruise campaigns) do periodically churn through new phone numbers, they do not seem to employ caller ID spoofing as aggressively as one may think.

Figure 13(a) and Figure 13(b) show the CBR for two prevalent campaigns, the *IRS* and Tech support scams, that can be identified from the COC data. The figures show that the source numbers used in these campaigns are more volatile than what we observed in the *Free Cruise* campaign in Figure 12, for example. This suggests that the spammers running these campaigns may be more aggressively using caller ID spoofing, or frequently churning through new phone numbers. However, the average CBR is above 60% showing that the COC blacklist can still effectively block a meaningful fraction of calls belonging to these campaigns.

F. Measuring False Positives

In the previous subsections we have shown how effective blacklists are at blocking potential spam calls. Naturally, a high blocking rate should be paired with a low false positive rate,

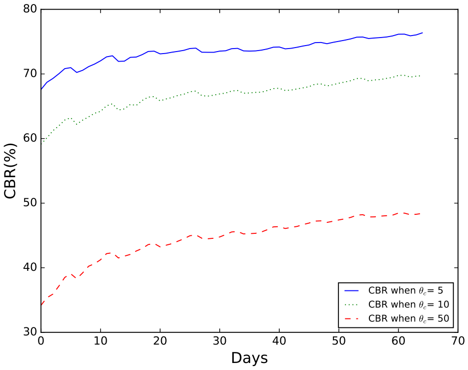


Fig. 9: FTC complaints blocking rate

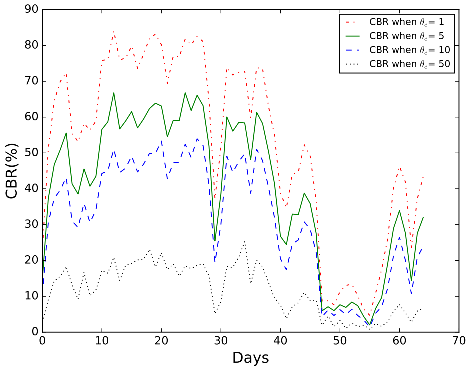


Fig. 10: COCNC complaints blocking rate

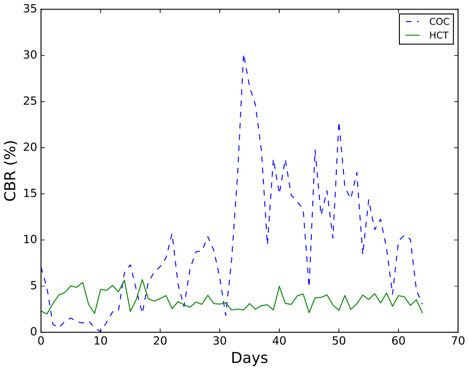


Fig. 11: Overall CBR for COC and HCT

for a blacklist to be useful in practice. Ideally, to measure a blacklist's false positive rate we would need access to a large whitelist of legitimate phone numbers that have never engaged in spamming activities. Unfortunately, we are not aware of the existence of any such whitelist.

Because no ready-to-use phone whitelist is available, to estimate the false positive rate of our blacklists we proceeded as follows. We first built an instrumented browser (using Selenium WebDriver) capable of crawling the YellowPages directory [60], which lists the phone numbers of businesses around the US. The assumption is that the vast majority of businesses that advertise on YellowPages are legitimate entities unlikely to engage in phone spam activities.

Using our crawler, we gathered around 100,000 phone numbers listed across 15 different major US cities and 10 different business categories, including doctors, plumbers, insurance, restaurants etc. We then checked each of these numbers against our blacklists, and found that only 10 of them were present, yielding a false positive rate of only 0.01%. We further investigated the phone numbers that resulted in these false positives. We found that 7 of these phone numbers appeared in the FTC blacklist with 20 complaints per phone number on the average. The remaining 3 phone numbers appeared in the CDR blacklist and on the average, they made 14 calls to 7 destinations. We also found that all of these 10 phone numbers have been complained about on 800notes.com for making unwanted or annoying calls.

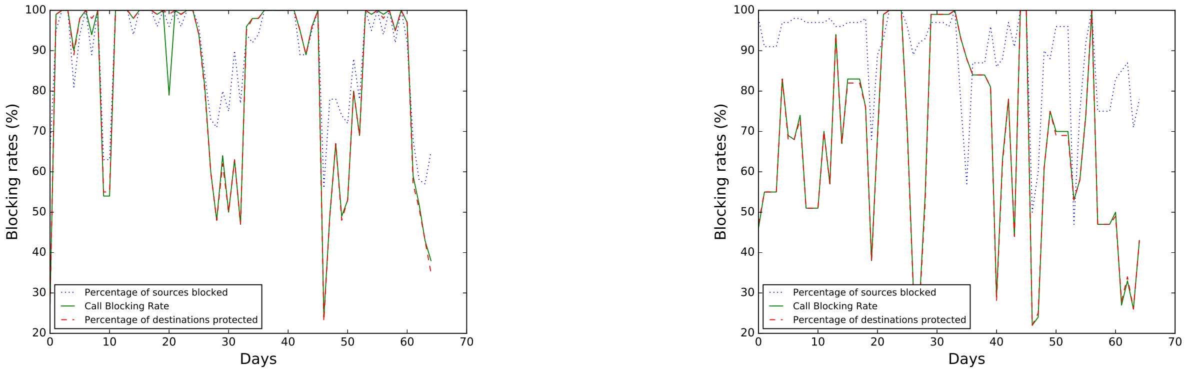
According to YellowPages, the phone numbers which led to false positives belong to medical centers, plumbing businesses, locksmiths, auto repair shops and grocery stores. In 800notes.com reports, users mentioned that these numbers were involved in making robocalls about an insurance scam or the caller claimed to be an Amazon associate asking for money. Some complaints mentioned the calls came from annoying telemarketers and debt collectors. One possible explanation for this is that, while belonging to seemingly legitimate businesses, these numbers may have been spoofed by spammers as part of a phone spam campaign. If this is true, the very low false positive rate suggests that such spoofing is not common.

To assess the FP rate, we used the data described in Section-II.E. Specifically, we used 100,000 benign phone numbers of businesses that were randomly chosen from YellowPages. While not complete, we believe this set of numbers is sufficiently large to yield a meaningful and reasonably accurate estimate of the FP rate.

G. Phone Abuse Campaigns

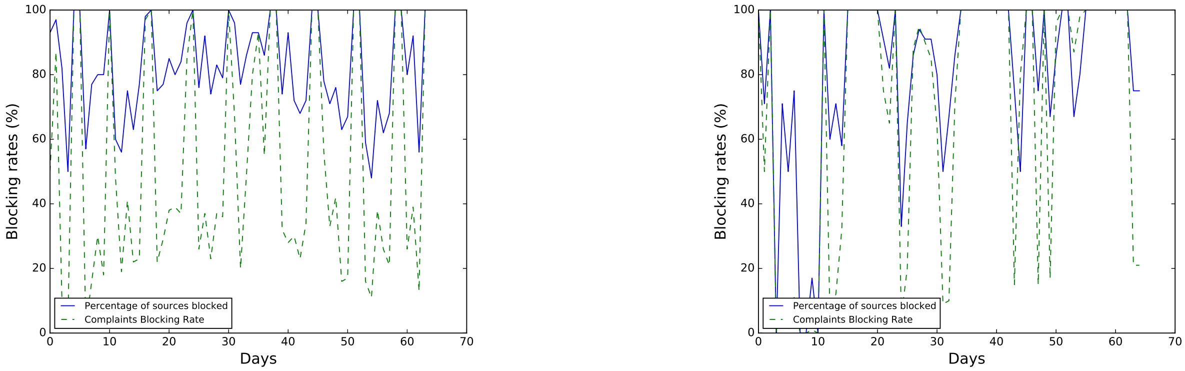
A natural question is whether the same phone numbers are used across different spam campaigns. In such cases, including a phone number on a blacklist due to abuse related to one scam could also protect users from other scams. Also, are the campaigns seen across multiple datasets or do they have higher visibility in a specific data source? We explore these questions by studying several prominent campaigns (derived from LSI topic modeling of our COC and HCT datasets). We found that the *Free Cruise* scam, shown in Figure 12, is seen in both the COC and HCT data sources, whereas the *Tech Support* and *IRS* scams shown in Figure 13 are largely seen only in the COC dataset and the *Google Listing* scam is seen in the HCT dataset. Figure 14 shows traffic over time for the top four campaigns in HCT and COC datasets

We used the COC dataset to further explore various abuse campaigns. For example, we conducted a pairwise analysis of a number of additional campaigns, including *Home Security* and *Pay Day Loan* scams, and we found a considerable amount of overlap in source numbers involved in separate campaigns. For example, 231 of the 500 phone numbers used in the *Free Cruise* scam (see Figure 12) are also used in the *Pay Day Loan* scam (notice that both campaigns target consumers). Similarly, we found around 90 phone numbers that were used for both IRS and Tech Support scams. While it is possible that different scammers may use caller ID spoofing and the same phone number can be spoofed in two unrelated scams, this is highly



(a) ***Free Cruise*** **(b)** ***Google Listings***

Fig. 12: Campaign call blocking rates over time.



(a) ***IRS*** **(b)** ***Tech Support***

Fig. 13: Complaints blocking rate for top campaigns.

unlikely for two scams that independently spoof numbers roughly at random, given the size of the phone numbers space. Therefore, it is more plausible that the same spammers are responsible for multiple spam campaigns.

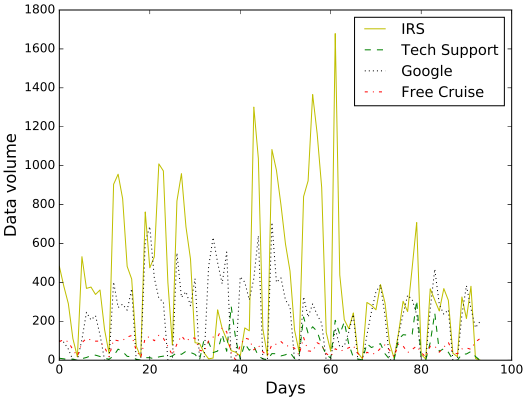


Fig. 14: Traffic over time

V. Discussion and Limitations

The CBR rates shown in Section IV were computed using cumulative blacklists. In other words, the blacklists were updated daily by adding new phone numbers, but old phone numbers were never removed. One may infer that CBR rates computed in such a way can be overly optimistic, as old phone numbers may get reassigned to legitimate users. Therefore, retiring old numbers from the blacklist is a reasonable practice. To demonstrate the effect of removing older phone numbers from the blacklist, we recomputed the CBR rates by updating the blacklist in a non-cumulative way. In other words, we define a window of size n and remove any phone numbers that were not seen in the last n days. Figure 15 shows that the CBR rates of COCNC drop by about 1%-15% depending on the window sizes. We get similar results with FTC CBR rates.

Our results show that phone blacklists offer meaningful mitigation for current telephony abuse. However, if such blacklists are deployed widely, scammers can utilize a number of techniques to evade them. The ease with which source phone numbers can be spoofed makes such evasion easy. In fact, we are already witnessing that scammers spoof a number that has the same area code as the victim to increase the likelihood that the call will be picked up by a targeted victim. An analysis of recent CDR data shows that there has been a 20% rise in such neighbor spoofing in 2017.

Although spoofing can make phone blacklists less effective, we believe that this will be more challenging for scammers in the future because of several recent initiatives [56]. For example, the Federal Communications Commission (FCC) has already proposed rules that allow carriers to block calls coming from unassigned or invalid phone numbers [57]. In absence of spoofing, the cost of acquiring a new spamming phone

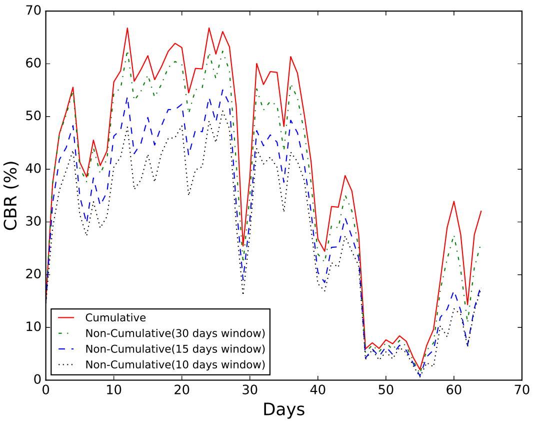


Fig. 15: COCNC CBR rates with different window sizes

number will be non-negligible. Besides the cost of purchasing a number, the attacker risks that the carrier from which the numbers are acquired could block all numbers purchased by an entity that has been found to engage in any spamming activities.

Scammers can also evade the blacklists by making fewer calls from a source phone number so the volume stays below a threshold such as θ_c used in the CDR blacklist. However, this will require that they utilize a larger pool of source phone numbers either by spoofing or by purchasing additional numbers. If spoofing can be made more difficult, evading the threshold will add cost to the phone scam operators. Our blacklists also rely on user provided data in the FTC and COC datasets. Scammers could target these data sources by injecting noise into them to reduce the effectiveness of blacklists. Also, a user may mistakenly provide incorrect information. For example, when a user receives multiple scam calls, they could incorrectly report the context of a call. Similarly, a more sophisticated attacker may check for liveness before providing call details in the HCT dataset. If such tactics become common in the future, the effectiveness of the blacklists could degrade unless techniques are employed to collect higher quality and more complete data.

VI. Related Work

Similar to online abuse and scams, telephony abuse has grown considerably and high volume scams have proliferated over the phone channel [61], [65]. Security researchers have explored such abuse and specific scams. For example, Costin et. al. [53] investigated how phone numbers are used in various scams based on analysis of crowd sourced web listings. A detailed analysis of the tech support scam, which is also explored in this paper, was presented in [32]. This work presented detailed insights into the online and phone infrastructure used by tech support scams and it also examined tactics of scammers by engaging them in calls. However, their results are for Tech-Support scams only, whereas our blacklists include a broader range of campaigns. Moreover, [32] mostly considered phone numbers related to outgoing calls, whereby the user contacted the Tech-Support number. Our study instead focuses on incoming calls, in which the users/honeypot received the calls from the scammers. The targeting of international students by the IRS scam has been explored in [62].

Our phone blacklists are based on several data sources, including a CDR dataset from a honeypot. Phoneypot [1], a phone honeypot, consisted of a pool of phone numbers and associated infrastructure to collect call detail records (CDRs) when honeypot numbers are called. Phoneypot demonstrated the feasibility of augmenting abuse information available in existing crowd sourced and self-reported datasets like 800notes and FTC complaint database. Although the data collected from the honeypot was analyzed to discover various abuse patterns (e.g., debt collection calls), blacklists were not explored by this work. Similarly, [2] presents an analysis of a large dataset of VoIP CDRs, analyzes different call patterns and groups callers using unsupervised techniques. The features mentioned in [2] suffice to group users, but they are not designed to differentiate between spammers and the legitimate callers. Clustering of transcripts recorded by a phone honeypot to identify and block calls from major bad actors was explored in [5] . However, since transcripts is the only abuse information source, it can only block calls from the campaigns that are seen at the honeypot.

Profiling of callers has been investigated by several researchers [16], [17], [36]. However, they assume access to large CDR datasets which have associated privacy concerns and telephony service providers do not make such datasets available due to privacy reasons. Yardi et al. [26] characterized behavioral patterns that disambiguate spammers from legitimate users. Call duration and social network connections are used to separate legitimate callers from spam/scam callers in [31] by developing a global reputation based system for callers Although low reputation can be placed on a blacklist, call duration information and social connections of phone users are not available. Furthermore, blacklists and their evaluation in not addressed by this work.

Blacklists have been investigated for domain names, IP addresses and other online resources such as URLs to combat email spam and malware infections [4], [19]. However, these papers typically utilize network, email content and other application-specific features to blacklist such resources which differ significantly from information available in our phone abuse datasets. SMS spam is related to phone abuse and has been investigated in the past [3]. However, unlike voice calls, SMS message content becomes available before it needs to be delivered which allows content-based detection.

To the best of our knowledge, this work is the first one to systematically explore phone blacklists using context-rich and context-less datasets. In addition, it provides insights into the effectiveness of such blacklists.

VII. Conclusions

Call blocking apps for smartphones are now becoming commonplace but little is known about the efficacy of such applications in protecting users from unwanted/scam calls. We present results of a data-driven study that utilizes multiple data sources to explore the feasibility and effectiveness of phone blacklists for blocking such calls . We demonstrate how phone blacklists can be learned both when the context of a call is known and when such context is not available due to a variety of reasons (e.g., privacy concerns, recording overhead etc.). Our results show that phone blacklists could block a meaningful fraction of unwanted/scam calls (over 55%). We also demonstrate that blacklists can be an effective defense against major phone abuse campaigns that have targeted consumers and businesses.

Currently phone blacklists can be effective against unwanted and scam calls but their effectiveness can suffer with increased level of caller ID spoofing. If spoofing increases, we will either need to detect it or make it more difficult to spoof phone numbers. We will explore efficacy of anti-spoofing techniques in our future research.

Acknowledgment

We thank Brad Reaves for serving as our shepherd, and the anonymous reviewers for their constructive comments and suggestions for improvement.

This material is based in part upon work supported by the National Science Foundation, under grants No. CNS-1514035 and CNS-1514052. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] Gupta, Payas, Bharath Srinivasan, Vijay Balasubramaniyan, and Mustaque Ahamad. ”Phoneypot: Data-driven Understanding of Telephony Threats.” In NDSS. 2015.

[2] Chiappetta, S., Claudio Mazzariello, Roberta Presta, and Simon Pietro Romano. ”An anomaly-based approach to the analysis of the social behavior of VoIP users.” Computer Networks 57, no. 6 (2013): 1545-1559.

[3] Xu, Qian, Evan Wei Xiang, Qiang Yang, Jiachun Du, and Jieping Zhong. ”Sms spam detection using noncontent features.” IEEE Intelligent Systems 27, no. 6 (2012): 44-51.

[4] Feamster, Nick, Alexander G. Gray, Sven Krasser, and Nadeem Ahmed Syed. SNARE: Spatio-temporal Network-level Automatic Reputation Engine. Georgia Institute of Technology, 2008.

[5] Marzuoli, Aude, Hassan A. Kingravi, David Dewey, Aaron Dallas, Telvis Calhoun, Terry Nelms, and Robert Pienta. ”Call me: Gathering threat intelligence on telephony scams to detect fraud.” Black Hat 2016.

[6] Deerwester, Scott, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. ”Indexing by latent semantic analysis.” Journal of the American society for information science 41, no. 6 (1990): 391.

[7] Ester, Martin, Hans-Peter Kriegel, Jrg Sander, and Xiaowei Xu. ”A density-based algorithm for discovering clusters in large spatial databases with noise.” In Kdd, vol. 96, no. 34, pp. 226-231. 1996.

[8] http://scikit-learn.org/stable/modules/generated/sklearn.cluster.DBSCAN.html

[9] https://radimrehurek.com/gensim/

[10] Ramos, Juan. ”Using tf-idf to determine word relevance in document queries.” In Proceedings of the first instructional conference on machine learning. 2003.

[11] https://en.wikipedia.org/wiki/Singular_value_decomposition.

[12] https://www.ftc.gov

[13] How does a robocall work? January 2015. https://www.consumer.ftc.gov/articles/0381-how-does-robocall-work-infographic

[14] Results of worldwide telecom fraud survey 2015. http://cfca.org/fraudlosssurvey/2015.pdf

[15] Mustafa, Hossen, Wenyuan Xu, Ahmad Reza Sadeghi, and Steffen Schulz. ”You Can Call but You Can’t Hide: Detecting Caller ID Spoofing Attacks.” In Dependable Systems and Networks (DSN), 2014 44th Annual IEEE/IFIP International Conference on, pp. 168-179. IEEE, 2014.

[16] Jiang, Nan, Yu Jin, Ann Skudlark, Wen-Ling Hsu, Guy Jacobson, Siva Prakasam, and Zhi-Li Zhang. ”Isolating and analyzing fraud activities in a large cellular network via voice call graph analysis.” In Proceedings of the 10th international conference on Mobile systems, applications, and services, pp. 253-266. ACM, 2012.

[17] Tseng, Vincent S., Jia-Ching Ying, Che-Wei Huang, Yimin Kao, and Kuan-Ta Chen. ”FrauDetector: A Graph-Mining-based Framework for Fraudulent Phone Call Detection.” In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 2157-2166. ACM, 2015.

[18] ?Directory of Unknown Callers,? http://800notes.com/.

[19] Povey, Daniel, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann et al. ”The Kaldi speech recognition toolkit.” In IEEE 2011 workshop on automatic speech recognition and understanding, no. EPFL-CONF-192584. IEEE Signal Processing Society, 2011.

[20] Ramachandran, Anirudh, and Nick Feamster. ”Understanding the network-level behavior of spammers.” In ACM SIGCOMM Computer Communication Review, vol. 36, no. 4, pp. 291-302. ACM, 2006.

[21] Provos, Niels. ”A Virtual Honeypot Framework.” In USENIX Security Symposium, vol. 173, pp. 1-14. 2004.

[22] Provos, Niels, and Thorsten Holz. Virtual honeypots: from botnet tracking to intrusion detection. Pearson Education, 2007.

[23] John, John P., Fang Yu, Yinglian Xie, Arvind Krishnamurthy, and Martn Abadi. ”Heat-seeking honeypots: design and experience.” In Proceedings of the 20th international conference on World wide web, pp. 207-216. ACM, 2011.

[24] Wyatt, Edward. ”Multinational Crackdown on Computer Con Artists.” New York Times (2012).

[25] Ollmann, Gunter. The phishing guide. Next Generation Security Software Limited, 2004.

[26] Yardi, Sarita, Daniel Romero, and Grant Schoenebeck. ”Detecting spam in a twitter network.” First Monday 15, no. 1 (2009).

[27] Prakash, Pawan, Manish Kumar, Ramana Rao Kompella, and Minaxi Gupta. ”Phishnet: predictive blacklisting to detect phishing attacks.” In INFOCOM, 2010 Proceedings IEEE, pp. 1-5. IEEE, 2010.

[28] Balasubramaniyan, Vijay A., Aamir Poonawalla, Mustaque Ahamad, Michael T. Hunter, and Patrick Traynor. ”PinDr0p: using single-ended audio features to determine call provenance.” In Proceedings of the 17th ACM conference on Computer and communications security, pp. 109-120. ACM, 2010.

[29] http://www.whitepages.com

[30] Blondel, Vincent D., Adeline Decuyper, and Gautier Krings. ”A survey of results on mobile phone datasets analysis.” EPJ Data Science 4, no. 1 (2015): 10.

[31] Balasubramaniyan, Vijay, Mustaque Ahamad, and Haesun Park. ”CallRank: Combating SPIT Using Call Duration.” Social Networks and Global Reputation. In CEAS (2007).

[32] Miramirkhani, Najmeh, Oleksii Starov, and Nick Nikiforakis. ”Dial One for Scam: A Large-Scale Analysis of Technical Support Scams.” In Proceedings of the 24th Network and Distributed System Security Symposium (NDSS). 2017.

[33] Dantu, Ram, and Prakash Kolan. ”Detecting Spam in VoIP Networks.” SRUTI 5 (2005): 5-5.

[34] Kim, Hyung-Jong, Myuhng Joo Kim, Yoonjeong Kim, and Hyun Cheol Jeong. ”DEVS-based modeling of VoIP spam callers? behavior for SPIT level calculation.” Simulation Modelling Practice and Theory 17, no. 4 (2009): 569-584.

[35] Secure Telephony Identity Revisited, IETF Working Group, https://tools.ietf.org/wg/stir/.

[36] Bai, Yan, Xiao Su, and Bharat Bhargava. ”Detection and filtering Spam over Internet Telephony?a user-behavior-aware intermediate-network-based approach.” In Multimedia and Expo, 2009. ICME 2009. IEEE International Conference on, pp. 726-729. IEEE, 2009.

[37] Endler, David, and Mark Collier. Hacking exposed VoIP: voice over IP security secrets & solutions. McGraw-Hill, Inc., 2006.

[38] Murynets, Ilona, and Roger Piqueras Jover. ”Crime scene investigation: SMS spam data analysis.” In Proceedings of the 2012 ACM conference on Internet measurement conference, pp. 441-452. ACM, 2012.

[39] Griffin, Slade E., and Casey C. Rackley. ”Vishing.” In Proceedings of the 5th annual conference on Information security curriculum development, pp. 33-35. ACM, 2008.

[40] https://play.google.com/store/apps/details?id=com.google.android.dialer

[41] https://www.truecaller.com

[42] https://www.youmail.com

[43] https://www.pindrop.com

[44] https://www.att.com/offers/call-protect.html

[45] Antonakakis, Manos, Roberto Perdisci, David Dagon, Wenke Lee, and Nick Feamster. ”Building a Dynamic Reputation System for DNS.” In USENIX security symposium, pp. 273-290. 2010.

[46] Felegyhazi, Mark, Christian Kreibich, and Vern Paxson. ”On the Potential of Proactive Domain Blacklisting.” LEET 10 (2010): 6-6.

[47] https://developers.google.com/safe-browsing/?csw=1

[48] Dietrich, Christian J., and Christian Rossow. ”Empirical research of ip blacklists.” In ISSE 2008 Securing Electronic Business Processes, pp. 163-171. Vieweg+ Teubner, 2009.

[49] Ramachandran, Anirudh, David Dagon, and Nick Feamster. ”Can DNS-based blacklists keep up with bots?.” In CEAS. 2006.

[50] Gmez Hidalgo, Jos Mara, Guillermo Cajigas Bringas, Enrique Puertas Snz, and Francisco Carrero Garca. ”Content based SMS spam filtering.” In Proceedings of the 2006 ACM symposium on Document engineering, pp. 107-114. ACM, 2006.

[51] Moya, Mary M., and Don R. Hush. ”Network constraints and multi-objective optimization for one-class classification.” Neural Networks 9, no. 3 (1996): 463-474.

[52] Tax, David Martinus Johannes. ”One-class classification.” (2001).

[53] Costin, Andrei, Jelena Isacenkova, Marco Balduzzi, Aurlien Francillon, and Davide Balzarotti. ”The role of phone numbers in understanding cyber-crime schemes.” In Privacy, Security and Trust (PST), 2013 Eleventh Annual International Conference on, pp. 213-220. IEEE, 2013.

[54] Mimno et al. “Optimizing Semantic Coherence in Topic Models.” In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP ’11), 2011.

[55] http://www.baidu.com/

[56] https://transition.fcc.gov/cgb/Robocall-Strike-Force-Final-Report.pdf

[57] FCC, “FCC SEEKS RELIABLE CALL AUTHENTICATION SYSTEM.” July 13, 2017. https://www.fcc.gov/document/fcc-seeks-reliable-call-authentication-system

[58] H. Tu, A. Doup, Z. Zhao and G. J. Ahn, ”Toward authenticated caller ID transmission: The need for a standardized authentication scheme in Q.731.3 calling line identification presentation,” 2016 ITU Kaleidoscope: ICTs for a Sustainable World (ITU WT), Bangkok, 2016, pp. 1-8.

[59] Joint ATIS/SIP Forum Standard- Signature-based Handling of Asserted Information Using TOKENS (SHAKEN), https://access.atis.org/apps/group_public/download.php/32237/ATIS-1000074.pdf

[60] https://www.yellowpages.com/

[61] Sahin, Merve, Aurlien Francillon, Payas Gupta, and Mustaque Ahamad. ”Sok: Fraud in telephony networks.” In Security and Privacy (EuroS&P), 2017 IEEE European Symposium on, pp. 235-250. IEEE, 2017.

[62] Bidgoli, Morvareed, and Jens Grossklags. ”Hello. This is the IRS calling.: A Case Study on Scams, Extortion, Impersonation, and Phone Spoofing.” In Proceedings of the Symposium on Electronic Crime Research (eCrime), Scottsdale, AZ. 2017.

[63] Reaves, Bradley, Logan Blue, Hadi Abdullah, Luis Vargas, Patrick Traynor, and Thomas Shrimpton. ”AuthentiCall: Efficient Identity and Content Authentication for Phone Calls.” In 26th USENIX Security Symposium (USENIX Security 17), pp. 575-592. USENIX Association, 2017.

[64] Reaves, Bradley, Logan Blue, and Patrick Traynor. ”AuthLoop: End-to-End Cryptographic Authentication for Telephony over Voice Channels.” In USENIX Security Symposium, pp. 963-978. 2016.

[65] Tu, Huahong, Adam Doup, Ziming Zhao, and Gail-Joon Ahn. ”SoK: Everyone Hates Robocalls: A Survey of Techniques against Telephone Spam.” In Security and Privacy (SP), 2016 IEEE Symposium on, pp. 320-338. IEEE, 2016.

Appendix

To find the “knee” of a topics weight curve (see Section III-C1), we use the following algorithm, which is visually represented in Figure 16:

1) Plot graph of sorted highest topic weights (blue curve);
2) Fit a low-order (e.g., order 4 or 6) polynomial onto the curve (red curve);
3) Compute the intersection between the left and right tangent lines (gray lines with negative slope);
4) Find the line that bisects the angle between the two tangents (gray line with positive slope);
5) Find the point in which the bisection line intersects the polynomial;
6) Project this point onto the y axis (dashed horizontal line).

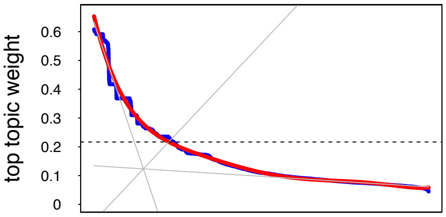


Fig. 16: Finding curve knee