# Who's Calling? Characterizing Robocalls through Audio and Metadata Analysis

*Sathvik Prasad*
*North Carolina State University*
*snprasad@ncsu.edu*

*Elijah Bouma-Sims*
*North Carolina State University*
*erboumas@ncsu.edu*

*Athishay Kiran Mylappan*
*North Carolina State University*
*akmylapp@ncsu.edu*

*Bradley Reaves*
*North Carolina State University*
*bgreaves@ncsu.edu*

## Abstract

Unsolicited calls are one of the most prominent security issues facing individuals today. Despite wide-spread anecdotal discussion of the problem, many important questions remain unanswered. In this paper, we present the first large-scale, longitudinal analysis of unsolicited calls to a honeypot of up to 66,606 lines over 11 months. From call metadata we characterize the long-term trends of unsolicited calls, develop the first techniques to measure voicemail spam, wangiri attacks, and identify unexplained high-volume call incidences. Additionally, we mechanically answer a subset of the call attempts we receive to cluster related calls into operational campaigns, allowing us to characterize how these campaigns use telephone numbers. Critically, we find no evidence that answering unsolicited calls increases the amount of unsolicited calls received, overturning popular wisdom. We also find that we can reliably isolate individual call campaigns, in the process revealing the extent of two distinct Social Security scams while empirically demonstrating the majority of campaigns rarely reuse phone numbers. These analyses comprise powerful new tools and perspectives for researchers, investigators, and a beleaguered public.

## 1 Introduction

The global telephone network serves more users than the Internet, is designed with higher availability guarantees, and is commonly relied upon for mission critical real time communications, including 911 service and emergency notifications [1, 2]. Despite its global importance, the phone network faces a number of problems. Prime among them are so-called "robocalls" — a catch-all term for automated or semi-automated unsolicited calls, often for fraud or telemarketing purposes [3]. Much like spam threatened the usefulness of email, unsolicited phone calls threaten to make voice calling unusable between two unknown parties. Moreover, because of the historic trust users have placed in the phone network, these scams surprisingly steal millions of dollars of revenue each year [4, 5].

Despite the clear importance of the problem, much of what is known about the unsolicited calling epidemic is anecdotal in nature. Despite early work on the problem [6–10], the research community still lacks techniques that enable rigorous analysis of the scope of the problem and the factors that drive it. There are several challenges that we seek to overcome. First, we note that most measurements to date of unsolicited volumes, trends, and motivations (e.g., sales, scams, etc.) have been based on reports from end users. In addition to the potential for selection bias, this information is often non-specific, unreliable, and/or incomplete. Second, most prior work on the problem has relied on analysis merely of the claimed number of the caller, neglecting to address the rampant (but previously unmeasurable) problem of number spoofing. Third, like modern cybercrime, robocalling is a commercial activity perpetrated at scale by professional operators [4, 5, 11]. Identifying the largest botnets and black markets has enabled targeted takedowns that reduce overall abuse on the Internet. Prior to this work, similar techniques for unsolicited calls have been out of reach. Such techniques could inform measurements, but also facilitate effective enforcement of the worst actors, leading to a decline in unsolicited calls.

In this paper, we operate a telephone honeypot that receives unsolicited calls over an 11-month period to up to 66,606 telephone lines. Our key innovation is the combined analysis of extensive and detailed call metadata with call audio. We combine this with novel techniques to identify similar calls efficiently allowing us to characterize whole campaigns of operation and detect fraud and abuse. While our purpose in this paper is to characterize a pernicious phenomenon, we note that our measurement techniques can provide valuable, actionable threat intelligence to carriers. In so doing, we provide a perspective on the problem that has been until now unavailable to researchers, regulators, and even carriers.

We use this new perspective to deliver 24 findings addressing three pressing questions:

- *How frequent are robocalls and is the problem getting worse?* We find that our lines can expect to receive a

robocall once every 8.42 days. Surprisingly, we learn that weekly call volumes are neither better nor worse over the observation period. We also discover and characterize rare "storms" of sudden unexplained bursts of unsolicited calls, providing support that anecdotal reports of high call volumes by individuals do occur.

- *Is it even safe to answer the phone?* Regulatory agencies and the press regularly warn of the risks of answering or returning calls from unknown callers. Shockingly, we discover no evidence that answering unsolicited calls increases daily call volume in a randomized single-blind study. We also develop heuristics to detect and measure wangiri call-fraud scams, finding no evidence of such a scam in 35 days across 2,949 highly-called lines.

- *Who is calling and how do they operate?* We develop and evaluate techniques to tightly cluster call audio to associate individual calls into broader campaigns with high precision. We then provide the first estimates of the number of operational campaigns and analyses of their number spoofing and line rotation practices and identify the top scam campaigns collected by our honeypot. All of these scams target vulnerable populations, including the elderly and recent immigrants, while operating over long time scales with impunity.

## 2 Background

To understand why unsolicited calling is such a challenging problem, we first need to review how the modern phone network operates. A call is said to "originate" at the caller's equipment, facilitated by the caller's carrier. It is the job of this carrier to "terminate" the call, which has the counter intuitive meaning of "establishing the connection", not ending it. If the originating carrier provides service to the called party, termination is straight forward. If however, the called party is served by another network, the originating carrier must route the call signalling and media through one or more intermediate carriers to reach the terminating carrier.

Carriers terminate calls using signalling protocols. In the PSTN[1], the most common protocol is Signaling System No. 7 (SS7). In VoIP, the most common protocol is Session Initiation Protocol (SIP). Carriers interconnect by establishing network gateways, which can operate over traditional PSTN trunks (called "TDM" in the industry) or VoIP, and often translate both signalling protocols (e. g., SS7 to SIP) and media encoding (e. g., PCM to Speex). It is important to note that when customers purchase VoIP-based telephone service from a provider, the customer does not actually place calls on an end-to-end basis with the called party. Instead, when the customer places a VoIP call, their local VoIP client software, physical phone, or phone gateway terminates the call at a proxy maintained by the provider. This provider-controlled proxy then routes the call to a peering partner's proxy, which forwards to another provider, and so on until the called party's provider receives the call and delivers it to the called party.

This state of affairs may seem surprising, but it is to prevent abuse of the network. Further, carriers are not allowed to listen to call audio of subscribers to protect their privacy. Instead, the call recipient must make a complaint, or the carrier must identify a malicious operator by call metadata. Carriers are required by law to maintain records on all calls they originate or route, but they are not required to make this information public. As a result when fraud specialists identify a fraudulent call, they must coordinate with every carrier in the entire call path to identify the origin. This entirely manual process is known as "traceback." A single call traceback can take dozens of hours to complete, making it largely infeasible.

### 2.1 Identity in the Phone Network

The principal identifier in the phone network is the phone number. While different countries and regions have different formats, all are unified in the ITU E.164 general format for numbers for unambiguous interpretation. Blocks of phone numbers are assigned to carriers according to the North American Numbering Plan (NANP), which covers all of the United States, Canada, and 23 other countries or territories. Carriers then assign numbers to subscribers. A valid NANP number has a country code (e.g. "1" for USA and Canada), three digit Numbering Plan Area code (NPA), three digit Exchange code (termed "NXX") and a four digit line number. There are fine-grained restrictions on NPA, NXX and the line numbers which determine if a phone number is valid, assigned, toll-free, or charges a premium rate when called.

The feature known as "caller ID" actually takes several forms in the PSTN. The first form, Calling Line Identification (CLI) includes the phone number of the calling party in the signalling information to setup the call. The second form is a 15-digit alphanumeric string to identify the caller known as Caller ID Name (CNAM). CNAM is not carried in the signalling of the call. Instead, the terminating provider performs a lookup for the CNAM associated with a CLI by querying an authoritative database in the telephone network.

Caller ID in SIP calls is more complicated. Identity info can be carried in the "To:" and "From:" fields of an INVITE message, the first signalling message to set up a VoIP call. These fields are populated by the SIP client controlled by the end customer. Some providers optionally append an additional identity header called a "P-Asserted-Identity" header. This header is meant to indicate a "true" identity to be used by the originating provider or its peers to traceback a source. Recently, a new standard to authenticate phone calls, STIR/SHAKEN [12], has been developed and is in the earliest stages of deployment. In this protocol, originating providers append a signature to the SIP header indicating that they indeed originated the call.

[1] Public Switched Telephone Network

This is also intended to facilitate traceback of abusive calls to their original source. *When deployed, STIR/SHAKEN will be the first and only widely-used cryptographic authentication mechanism anywhere in the telephone network.*

Operations that make large amounts of unsolicited calls, especially those doing so illegally, have a strong incentive to obscure their source phone number. They may do this to entice callers to answer, to avoid easy blocking based on caller ID, and/or to frustrate attempts to prosecute callers. There are a number of methods they can use to accomplish this. The first is to ask the terminating provider to block the caller ID to prevent it from being delivered to the called party. In the United States, callers can precede their call with the prefix "*67" to do this. In practice, this provides little anonymity because all carriers on the path see the true identity.

The second method is to purchase VoIP service from a provider who does not check outbound "From" fields for correct values. Many providers allow arbitrary "From" fields *as a feature* for customers who wish to present a main business number (e.g., a customer support number) that may not be owned by that provider. This is the most common form of caller ID spoofing. A special form of caller ID spoofing aims to match the caller's first six digits (NPA-NXX). This practice is termed "neighbor spoofing" as it is meant to entice victims to answer a phone call believing it is a neighbor or local organization (such as a school). The final method is to simply purchase a very large pool of phone numbers and rotate through them, often keeping them for only a short time. Operators have informed us this is an occasional practice by mass unsolicited callers. We note that from our viewpoint of measuring unsolicited phone calls, caller ID spoofing and simply having a large, rapidly changing pool of numbers is indistinguishable. As such, in this paper, we call the practice of changing numbers frequently "line rotation" regardless of mechanism.

## 2.2 Unsolicited Calls

Unsolicited calls may be known by many different terms, including "robocalls", "phone spam," and "vishing". Not all unsolicited calls are illegal or undesirable. Examples include public safety announcements for evacuations or school closures.

Most unsolicited calls are undesired yet may be legal. In the United States, calls made by political campaigns are legal. Some telemarketing calls are also legal, provided they are not targeted at cell phones, the called party has not subscribed to the FTC's "Do Not Call" list, or the caller has given permission for the call. Not only do most individuals not care for such calls, often these sales calls are for undesirable products criticized by consumer advocates, like auto warranties.

A small fraction of unsolicited calls are illegal scams. These scams may impersonate law enforcement or government agencies for taxes or benefits. They may also impersonate or fraudulently claim to be representatives of respected brands, as in tech support scams [13, 14] or fraudulent vacation sales [11].

Two categories of unsolicited calls are not intended to be answered. The first is voicemail spam. Rather than enticing their targets to listen to a recorded message in realtime, voicemail spam "injects" the recording into the voicemailbox of the target [3, 15]. Spammers will place two simultaneous calls to the target so that the second call finds the line busy and is redirected to voicemail. When the second call is connected, the first is disconnected by the caller, often before it rings.

The second type of unanswered call is known as a "one-ring" or "wangiri" scam, derived from a Japanese term which translates to "One (ring) and cut". In this scam, the perpetrator first obtains a premium rate number that bills all callers at a high rate (e.g. five dollars per minute). The perpetrator then calls a large number of victims indiscriminantly, hanging up just after it starts to ring. These calls are effectively free for the perpetrator because incomplete call attempts are not billable. However, the victim sees a missed call, and many victims will attempt to return the call, discovering they were billed only after their phone bill arrives. This scam is especially effective in North America if the premium rate number is obtained in certain Caribbean countries that are part of the North American Numbering Plan, as those phone numbers appear to be domestic and are not obviously charging a premium.

# 3 Data Collection

In this section, we explain the design principles of our honeypot, discuss the history of phone numbers used in our experiments, describe our data collection methodology, highlight ethical and legal considerations of our work and finally share details about a secondary data set used in our study.

## 3.1 Designing a Telephony Honeypot

A honeypot owned by a researcher allows adversaries to interact with a set of resources in an isolated environment. A telephony honeypot collects information about the entities that operate in the phone network. To collect such information, we assign a set of phone numbers to a honeypot. These phone numbers were provided to us by our service provider, Bandwidth Inc. In this paper, we refer to these phone numbers as *inbound lines* [2]. Such a setup allows us to conduct controlled experiments, collect data, and characterize the phone calls.

We explain key design decisions of the deployment, configuration, testing and operation of our honeypot.

**On-premises deployment**: A local deployment of our honeypot provided fine-grained control over its design and ensured that we stored all the sensitive data on servers we own.

[2] *Inbound lines* : A set of virtual VoIP phone lines and not physical PSTN lines.

**Phone Numbers**: We worked with a telecommunication service provider who owned the phone numbers used in our experiments. We built our honeypot using Asterisk [3], which is an open-source software implementation of a Private Branch Exchange (PBX). With a setup like an enterprise VoIP consumer, our honeypot received and processed phone calls.
**Configuring the Call Processing System**: Like routing tables and routes of a router, the dial-plan and dial-rules of a PBX determine how it handles a phone call. By developing appropriate dial-plans, our honeypot automatically answers and records calls made to one set of lines, while the honeypot rejects any calls made to a different set of inbound lines.
**Reliability of the Call Processing System**: We used over 66,000 inbound lines for our experiments. With 66,000 configuration entries, Asterisk exhibited inconsistent behavior resulting in frequent crashes. After many iterations, we estimated that a single Asterisk instance can handle approximately 15,000 unique dial-plan entries under realistic load of phone calls. To operate a stable honeypot, we reduced the dial-plan's size by reusing dial-plan subroutines for each experiment and automating dial-plan generation.

## 3.2 History of Inbound Lines

The total number of inbound lines terminated on our honeypot varied at different stages of our study because our service provider dynamically added inbound lines to our honeypot. We kept track of any additional inbound lines added to our honeypot through periodic snapshots and updates to a local database. We account for this incremental addition of numbers to our honeypot throughout our experiments and normalize our measurements when appropriate. Based on the history of the inbound lines, we categorize them into two types:
**Abuse Numbers**: As reported by our service provider, abuse phone numbers had a history of abuse. Some of these numbers were returned by their previous owners due to high volume of unsolicited calls. This pool also included phone numbers previously used by spammers and robocallers to generate unsolicited phone calls. Abuse numbers are an invaluable resource for our honeypot because these numbers were owned by adversaries in the past or were victims of high volume of unsolicited calls. We started with 6,754 abuse numbers at the beginning of our study and obtained additional abuse numbers in April 2019, resulting in a total of 9,071 abuse numbers.
**Clean Numbers**: A set of phone numbers owned by our service provider which were intended for distribution among new users. This pool contained a combination of numbers which were newly procured by our service provider and numbers which were rotated from prior customers. These numbers did not have a reported abuse history. We obtained a total of 57,535 such clean phone numbers at the end of July 2019.

A combination of clean and abuse numbers allowed us to systematically measure and report our observations of the two extremes of the phone network. To the best of our knowledge, we are the first to develop a comprehensive telephony honeypot with both clean and abuse numbers.
**No Seeding**: Throughout our study, we do not seed either the clean or the abuse numbers on any online portals, forums, denylists or mobile apps which claim to block robocalls. By definition, the calls collected and processed in our experiments were unsolicited calls. We did not initiate any outbound calls using any inbound lines.

[3] https://www.asterisk.org

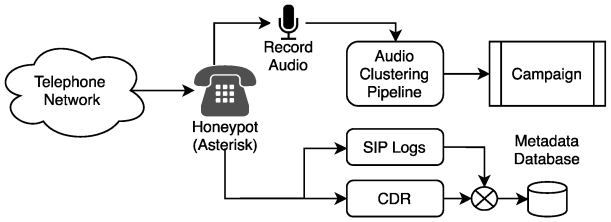


Figure 1: Honeypot Architecture and Data Collection Flow

## 3.3 Call Meta-data and Call Audio Collection

After designing and deploying our honeypot, we collect the call meta-data, which includes Call Detail Record (CDR) and SIP header logs. From CDR logs, we extracted the calling number, CNAM, called numbers, timestamp and optional call duration, if the honeypot answers the call. From SIP logs we get P-Asserted-Identity, a SIP header field which can contain different identification information.

Call audio is essential to characterize different spam and robocalling campaigns in the telephone network. To obtain a representative sample of call audio content, we initially selected 3,000 random lines from our pool. We refer to this set of lines as **Recording Lines 1 (RL1)**. We setup the dial-plan and configure our honeypot to answer any unsolicited call made to these 3,000 numbers and play a recording after a delay of 2 seconds. We use a default Asterisk audio recording as the source for the audio prompt, which says "*Hello*" in a female voice with an American accent.

On 21st December 2019, we analyzed the data collected thus far and identified the inbound lines which received an average of one or more calls per day. A total of 2,949 inbound lines met this criteria. We assign these inbound lines to a new Asterisk PBX and configure it to ring for 10 seconds before answering every call made to these lines. We call this set of lines as **Recording Lines 2 (RL2)**.

We configure our honeypot to record any unsolicited call made to an inbound line which belongs to Recording Lines 1 or 2. The honeypot records and stores every call as three separate audio streams — incoming (calling party to the honeypot), outgoing (honeypot to the calling party) and a combined recording. Separate recording streams allowed us to prevent issues caused by overlapping speech signals or locally generated noise or audio. We ensured that multiple simultaneous

calls made to the same inbound line generated separate recording files with appropriate timestamps. Finally, we rejected any unsolicited call made to a *non-recording inbound* line with a *403 Forbidden* SIP response code. We observed that certain SIP clients which initiate unsolicited calls retry calls multiple times when they receive a rejection from the called side. To address this, we identify and remove any duplicate calls which have the same calling and called number within a 30 second window. We do not consider these duplicate calls in any results in this paper.

A majority of service providers allow callers to mask their details by dialing with a prefix. In the United States, most subscribers can prefix the called number with *67 to ensure that the called party does not see the calling part's caller ID. By doing so, the caller ID shown to the user changes from the actual caller ID to a string like "Restricted", "Private" or "Anonymous". In our honeypot, we observed that there were multiple instances where the actual caller ID was replaced with string like "Restricted", "Private" or "Anonymous". We confirmed that our service provider's system was not manipulating the caller ID and instead, in some cases the actual caller ID was transparently passed from upstream service providers to our honeypot in the "P-Asserted-Identity" SIP header. Since neither us nor our service provider had control over caller ID information, we do not have caller ID information in the "P-Asserted-Identity" SIP header for all calls. Also, one of the key limitations of telephone networks is the lack of end-to-end caller authentication. Thus, the attested caller ID propagates across different boundaries in the phone network on best effort basis. Due to this, we do not assume that the caller ID information is complete or accurate.

While our study lasted over 11 months, Table 1 in the Appendix shows the exact dates when RL1 and RL2 were setup to collect call audio, maintenance downtime, power outage and the duration of t-test discussed later in Section 4.3.

## 3.4 Ethical and Legal Considerations

Our university's Institutional Review Board (IRB), our university's office of general counsel, and our provider reviewed and approved our experiments. We understand that our research may involve human subjects even though our main intention is to study automated phone calls. It is possible for a live human to call one of our inbound lines due to mis-dialing or while trying to reach the previous owner of the numbers. As responsible researchers, we take all the necessary actions to ensure that our research is within the legal and ethical boundaries. Before the start of our research, we ensured that we were compliant with ethical and legal restrictions imposed by the university, our state and the federal laws of United States. Specifically, we sought the approval of our IRB to address the ethical considerations of our study. We also worked closely with our university's Office of General Counsel to make sure that our actions are within the bounds of state laws of our state and the federal laws of United States.

Throughout our study, we ensured that our actions do not inflict harm to human subjects. We worked closely with our IRB before the start of our research to describe our experiments and the associated limitations. As part of this review process, we submitted a detailed report to our IRB. As explained in Section 3, our principal data collection methodology is to wait for the arrival of calls on the inbound lines owned by us. Our methods are similar to research studies that perform public observation of humans, except that we observe the behavior of humans in a virtual environment. In such a setting, we are neither targeting nor recruiting participants to take part in our study. We do not reach out to any participants. We strictly refrain from advertising the phone numbers of our inbound lines in spam portals, social media or through any other mechanisms. We do not initiate any outgoing calls to any phone numbers throughout our study. After a thorough review of our proposal, the above facts were carefully considered and the IRB determined that our research was exempt from further review on the basis that effectively, we are performing a public observation study.

In the United States, call audio is considered private information. Thus, recording a phone conversation is strictly regulated by state and federal laws. Our honeypot was setup in a state where single party consent is sufficient to record phone calls. In situations where a phone call spans across one or more state boundaries, federal law takes precedence over the state law. Federal law also mandates that at least a single party needs to consent for the phone call to be legally recorded. Throughout our study, all the calls that we recorded were made to the inbound lines we owned. Furthermore, we terminated these inbound lines on the Asterisk PBX which we operated. Since we explicitly consent to being recorded, we satisfied the single party consent requirement.

Many robocallers or spam campaigns make automated phone calls based on a "hitlist", which is a list of active phone numbers maintained and sold by third parties. As a result, the campaigns attempt to reach large groups of unknown recipients, seldom with the intention of reaching a known individual. Since these campaigns make unsolicited phone calls to unknown parties, it is reasonable to assume that the callers do not consider the call content especially private or sensitive. Not obtaining explicit consent of the caller (live human or automated call) prior to being recorded does not affect their rights or welfare. This is because the caller does not have a reasonable expectation that their calls are not being recorded. Further, these callers do not have a reasonable privacy expectation since they make unsolicited phone calls to a vast number of users.

The goal of our study is to develop a deeper understanding of the adversaries who operate in the telephone network, and not to identify details about individuals or specific callers from the data available in our honeypot. We designed our

experiments to limit the recording duration to 60 seconds. There are possibilities where a non-adversarial caller may make a phone call to one of the inbound lines configured for recording. By capping the recording duration to 60 seconds and by gracefully terminating the call at the 60 second mark, we minimize the amount of data gathered in such scenarios.

### 3.5 Industry Robocall Blocking Data

To evaluate our methodology in Section 5, we use a second corpus of phone calls provided to us by a company that builds services to help block robocalls. This data set consisted of the audio recording of the call, calling party number, timestamp of the call and the transcript of the call. Since we did not collect the data directly, we do not know the exact setup of the honeypot used for data collection or transcription.

## 4 Individual Call Characterization

In this section, we provide an overview of the data collected throughout our experiments. We delve into the temporal characteristics of call volume and highlight operational characteristics of unsolicited calls. We develop a method to identify and characterize high call-volume events. We statistically evaluate the effects of answering a phone call on the number of unsolicited calls received per inbound line. Next, we propose a heuristic to identify voicemail spam calls. We share a detailed analysis of caller ID spoofing in the wild and discuss how unsolicited callers reuse Caller ID Name (CNAM). We develop and apply a heuristic to identify wangiri scam and estimate the scale of wangiri scam observed in our honeypot. Finally, we delve into the characteristics of the call audio which sets the foundation for the subsequent section on campaign identification.

**Finding 1**: *Unsolicited phone calls are rampant in the United States.* Using the telephony honeypot described in Section 3, we collect 1,481,201 unsolicited phone calls over a span of 11 months, without seeding our phone numbers to any source. We observed an average of 4,137.43 unsolicited calls per day, across all the inbound lines used in our honeypot. Each inbound line received an average of 0.12 call per day, which translates to one call every 8.42 days.

Throughout our study, we track the state of clean and abuse lines assigned to our honeypot, since these were dynamically added to our honeypot by our provider. We owned a total of 66,606 unique inbound lines of which 57,535 were clean lines and 9,071 were abuse lines, as explained in Section 3.2.

**Finding 2**: *Clean Numbers received 77.83% of all unsolicited calls in our honeypot without any form of seeding.* Among all the inbound lines, 87.08% (57,535) were clean inbound lines and 12.92% (9,071) were abuse inbound lines. The clean inbound lines with no history of abuse received an average of 0.11 call per day per inbound line, which translates to one call every 9.35 days. The abuse inbound lines received an average of 0.11 call per day per inbound line, which translates to one call every 9.44 days. The scale of our findings shows that it is not necessary for a phone number to have a prior history of abuse calls in order to receive unsolicited phone calls.

**Finding 3**: *75.10% of clean lines and 100% of abuse lines received at least one unsolicited phone call during our study.* We found that only a small fraction (24.90 %) of all the clean numbers never received an unsolicited call during our study. It took an average of 8.01 weeks for an inbound line to receive the first unsolicited call after being added to our honeypot.

Since calls arrive into a honeypot at a fairly low rate, only a fraction of these calls actually contain audio. This justifies our design decision of having adding a very large set of numbers as inputs to our honeypot. In particular, this shows that prior research [6, 16, 17] which relied on only a few hundred inbound lines was ultimately unlikely to see large portions of the problem.

### 4.1 Temporal Characteristics

The normalized daily call volumes per line over the 11 month study period is shown in Figure 2. We observed outliers that caused a spike in call volume during April of 2019, which we characterize in detail in Section 4.2.

**Finding 4**: *We observed a stationary call volume of unsolicited calls over our study period.* Since our study spanned 11 months, we were able to observe the cumulative call volume on both clean and abuse numbers over extended period of time. We fit a linear model to our weekly average call volume observed in our honeypot after discarding the two weeks affected by server downtime, finding a slope of -0.0002, indicating almost no change in the rate of unsolicited calls over the study period. We also fit a model after also discarding the anomalous storm peak in April, finding an even smaller slope of $-9 \times 10^{-5}$. In addition to its significance for the phone network, this is also an important result for our evaluating methods. While we do not know the history of the numbers before we possessed them, on the whole we see approximately the same volume of calls months after we take possession. This implies that recent activity before we take ownership of the line is unlikely to skew our results.

**Finding 5**: *The call volume of unsolicited calls had a periodicity of one calendar week.* The call volume increased on Mondays and remained high during weekdays. The call volume decreased on Saturday and remained low on Sunday. We observed this pattern in every week of our data collection. To measure the extent of periodicity, we compute the auto-correlation score — a score from 0 to 1 which measures the similarity of a signal with itself at different time lags between the two copies of the signal. For daily unsolicited call volume, we observed a maximum auto-correlation of 0.87 at a time lag of 7 days.

**Finding 6**: *Our honeypot received 83.36% of all unsolicited phone calls during local working hours and 92.71% during*

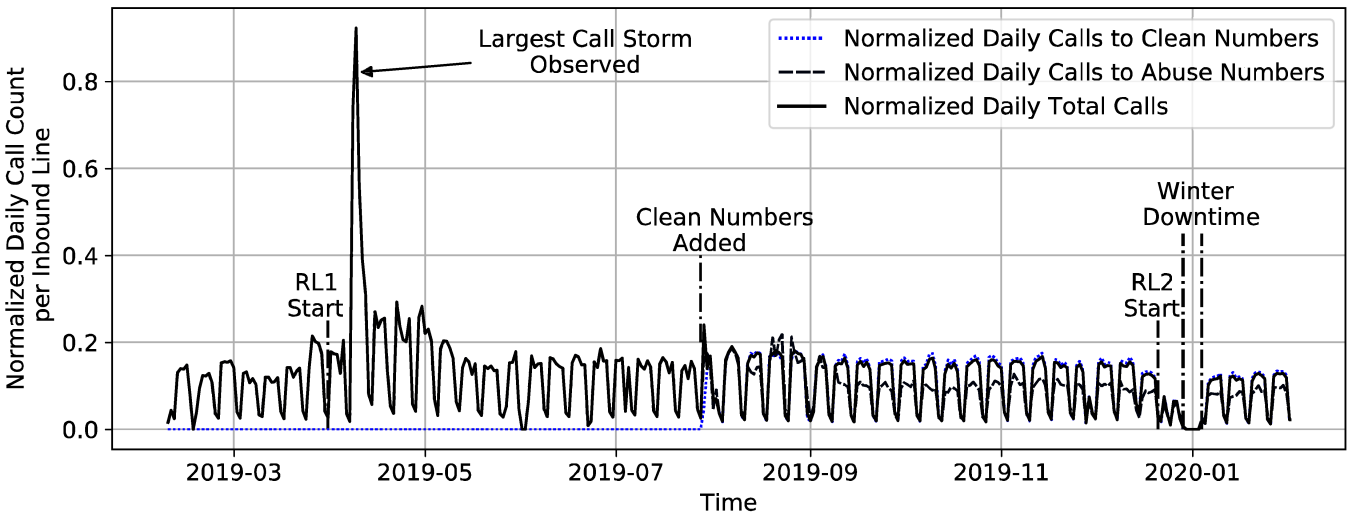


Figure 2: In this plot of normalized average calls, we observed a stationary call volume distribution of unsolicited phone calls on clean and abuse numbers with a weekly periodicity during our 11-month study. Major events in our honeypot are labelled.

*weekdays*. Intuitively, we would expect to receive significant amount of unsolicited calls when users are available to answer their phones. Weekends and non-working hours would seem to maximize the user's availability. Our honeypot received 83.36% of calls between 9 AM to 5 PM, as per the local timezone of our honeypot (Eastern Standard Time), which is roughly the local working hours. Furthermore, 92.71% of all calls were received during weekdays.

## 4.2 Storms: High Call Volume Events

When we observed an abnormally high number of calls in April of 2019, as seen in Figure 2, we delved deeper into the distribution of these calls over our inbound lines. We discovered instances when a disproportionately large number of calls were received on specific inbound lines. Using average call volume of each inbound line is not sufficient to identify such outliers. Inbound lines (e.g. abuse numbers) that regularly receive a significantly large number of unsolicited calls would naturally have a higher average call volume, but does not qualify as an outlier.

In this paper, we refer to such instances of high call volume occurrence as *storms*. To systematically identify storms, we wanted a uniform mechanism to compare call volumes in relative terms across all the inbound lines. To address this problem, we used $z$-score. The $z$-score is defined as $z = (x-\mu)/\sigma$, where $x$ is a data point in the distribution, $\mu$ is the mean and $\sigma$ is the standard distribution. We computed the $z$-score distribution of daily call volume per day, for each individual inbound line. A $z$-score of 1 for a specific day indicates that the call volume on that day is a single standard deviation away from the mean call volume of the inbound line. A higher $z$-score indicates that the measured value is farther away from the mean. We use a conservative heuristic and set a $z$-score of 4 as the limit to identify calls that received abnormally high calls per day during our study. A $z$-score greater than 4 indicated that the call volume on the specific day was 4 standard deviations higher than its mean call volume. Such behavior is an intuitive indication of an outlier.

Even though $z$-score allowed us to develop relative comparison, it includes inbound lines which has very low average call volume with sporadic calls. To remove these inbound lines with low call volume, but with significant high $z$-scores due to an occasional call, we set a threshold of a minimum call volume per day of 24 calls. A 24 calls per day threshold translates to one call per hour — this is a significant amount of unsolicited call volume. We identified inbound lines which received more than 24 calls on any single day, and had a $z$-score of greater than 4 during our study. By so doing, we identify inbound lines that received a significantly high call volume and characterize this phenomenon as a storm.

**Finding 7**: *We observed 648 instances of storms spread across 223 inbound lines.* A 11 month long study helped us uncover numerous instances of storms. The largest storm comprised of over 1,400 unique unsolicited calls made to the same number on the same day. These calls seemed to originate from over 750 unique callers based on the number of unique caller IDs used. We note that in prior work, Gupta et al. [6] report 2 "TDoS" events over their 7 week observation period. Our findings indicate such events are rare, yet occur regularly. We also note, our term "storm" does not imply malicious intent, as we cannot attribute a course or source of these events. Throughout our study, we observed storms as early as March 2019 and as late as January 2020.

Our discovery of storms also confirms anecdotal reports where individuals seem to be deluged seemingly "out of the blue" by dozens of calls in a day. Most of our storm events oc-

cur on unrecorded lines. [4] As a result, it is unclear if the storms originate from a single operation or campaign, or if storms comprise a chance coincidence where one line is randomly targeted by many different campaigns.

## 4.3 Effects of Answering Unsolicited Phone Calls

One of the most common recommendations to tackling the problem of unsolicited calls is to not answer any calls originating from unknown numbers (numbers not in the user's contact list), under the hypothesis that answering will increase call volume. To understand if there is a significant impact of answering phone calls to the number of unsolicited phone calls received on an inbound line, we designed an experiment and statistically evaluate our measurements. For this experiment, we randomly selected 3000 inbound lines, which were the same lines initially referred to as Recording List 1 (RL1). Initially, we did not answer any unsolicited calls made to these 3000 inbound lines for 6 weeks. Next, we answered all calls received on these 3000 inbound lines and observed the call volume for 6 weeks. We calculated the average call volume of each line in RL1 during the first 6 weeks of not answering the phone call. We also computed the average call volume during the next 6 weeks, when we answered all calls made to these inbound lines. To understand if there is significant evidence that answering phone calls has an effect on the number of unsolicited phone calls, we apply a statistical test based on average call volume observed from 17th February to 12th April of 2019.

We use $t$-test for dependent populations to measure if the difference between the means of two populations is significant. We also select an alpha value of 0.01 to determine the significance of our statistical test. Our p-value should be less than alpha to indicate statistical significance.

Since we observe a peak in overall call volume, which we have associated to storms, we checked if any of the inbound lines of RL1 were victims of such huge call volume. We confirmed that there were no storms associated with any of the RL1 inbound lines. This steps ensures that there were no outliers when we perform the $t$-test.

**Finding 8**: *Answering unsolicited calls did not have a statistically significant effect on the average number of unsolicited phone calls received on a phone number.* We observed that average call volume when not answering calls was 0.1027 and average call volume when we were answering phone calls was 0.0944. Our $t$-test indicated the result was statistically insignificant ($p = 0.0708$). Through this result, we conclude that there is no evidence that answering phone call increases the number of unsolicited phone calls received. This finding contradicts the traditional wisdom and provides insight to operators in that our findings indicate that it would be safe for operators to monitor and use lines without the risk of further contamination.

[4] In the absence of evidence to the contrary, we assume this is simply due to the fact the majority of our lines are not answered.

## 4.4 Voicemail Spam

Unlike traditional landline or mobile phones, our inbound lines did not have the restriction of maintaining only one active call at a time. Such a configuration allowed us to observe multiple call attempts with the same calling and called numbers in quick succession — a classic behavior of voicemail spam. Since the successive call attempts maintained the same calling and called numbers, we identified groups which have a unique 3 tuple of the calling number, the called number and the date. We discard the groups which have a single call. Next, we calculate the time difference between successive calls in each group. Since our honeypot rejected a fraction of incoming calls with a 403 SIP Response code, we observed clients re-trying the same call within a short duration of time, as discussed in section 3. After referring to the SIP retransmission section in the SIP [18] RFC, we remove all duplicate retries within 30 seconds of each other.

**Finding 9**: *We estimate that 2.91% of all calls made to our honeypot were suspected voicemail injection attacks.* Most adversaries need to tune their campaigns through manual delay measurement and determine the ideal time difference between successive calls for executing voicemail spam. Such delay estimation vary depending on how a phone call is routed from the source to the destination. We performed test calls across multiple originating service providers to estimate the delay associated with call setup. By empirical estimation, we set a conservative window of 30 to 90 seconds as the time difference between successive calls to execute a successful voicemail injection. We identified 43,170 calls within this window which we believe are successful voicemail spam or voicemail injection attempts.

Our findings also indicate that voicemail spam is likely a significant problem. However, because our heuristics rely only on signaling information alone, it should be detectable by carriers. Though in magnitude similar, this would have the effect of eliminating an **entire class** of telephone fraud. While we have tried to design our heuristics to make it practical and usable, careful testing and validation with ground truth is essential before deployment in live networks.

## 4.5 Caller ID Spoofing

**Finding 10**: *We estimate that 6.12% of all unsolicited calls used neighbor spoofing techniques.* For calls where the calling number adheres to NANP, we compare the calling number with the called number to identify the length of the match. We compared the calling and the called numbers and found that 27.67 % (409,876) of all calls had identical area codes (NPA) between the calling and the called number. Further, 6.12% (90,648) calls had both, a matching area code and a matching

exchange code (NPA+NXX). Surprisingly, 0.05% (698) calls were made with the same calling number as the called number for that call. We also observed that for 0.07% (976) calls, the caller ID used by the calling side was one of the 66,606 phone numbers owned by us. We used *libphonenumber* [5] module and openly available information from *North American Numbering Plan Administrator's* [6] website to parse and validate the non-US and US phone numbers respectively. We highlight neighbor spoofing as one example of a particular robocalling strategy. As callers continue to evolve their tactics we can use similar techniques to identify other trends and patterns.

Regulatory changes made by the Federal Communications Commission (FCC) in November 2017 [19] authorized telecom operators to block calls which seem to originate from unassigned, unallocated or invalid phone numbers. It also allowed providers to maintain a Do Not Originate list and block calls which seem to originate from a number on this list. These changes did not address scenarios where legitimate numbers were used to spoof the caller ID or when caller ID was not spoofed at all. The FCC acknowledged these limitations and allowed more flexibility to block calls by empowering the providers through its more recent regulatory changes in June 2019 [20].

**Finding 11**: *We found that only 3.2% (47,910 calls) of all the unsolicited calls made to our honeypot could have been outright blocked by providers.* We observed that only 5.97% (8,633) of all unique calling numbers seen in our honeypot met the criteria of call blocking. These percentages are a lower bound on the effectiveness of provider based call blocking, mainly because we cannot measure or collect information about calls which were blocked by the upstream providers.

As described in Section 3.3, calling parties can mask their identity by dialing with specific prefixes, like *67. In our honeypot, we collected SIP logs from which we extracted the caller ID information of unsolicited calls attempting to dial with a prefix, and in-turn mask their original caller ID.

**Finding 12**: *Out of 72,197 unsolicited calls which attempted to mask their caller ID by dialing with *67 as a prefix, 79.16% (57,151) were successful.* A small fraction (20.85%) of these unsolicited callers leaked their actual caller ID through the "P-Asserted Identity" SIP header, but most calls that dialed a call using the *67 prefix successfully masked their caller ID. This observation is an example of how unsolicited callers can use existing features in the phone network to evade detection.

As described in Section 2.1, CNAM is a feature through which a set of 15 characters can be sent to the called party. When CNAM information is available, it represents the name of the owner of the calling phone number.

**Finding 13**: *A large number of callers used a small pool of caller names (CNAM) when making unsolicited phone calls.* From the data collected in our honeypot, we observed that there were 811,262 unique calling entities who had made an unsolicited call. Each calling entity is uniquely identified by a combination of calling party's phone number and the Caller ID Name (CNAM). Of these 811,262 (100%) calling entities, we observed that there were 801,466 (98.79%) unique phone numbers (caller IDs) and 239,210 (29.49%) unique CNAMs, which indicates rampant reuse of CNAMs.

[5] https://github.com/daviddrysdale/python-phonenumbers

[6] https://nationalnanpa.com/number_resource_info/index.html

## 4.6 Wangiri Scam Estimation

We studied wangiri scam attempts on 2,949 inbound lines (RL2) which were configured to ring for 10 seconds and answered any unsolicited call. We defined a heuristic and empirically estimate the scale of wangiri scams. Since all our inbound lines were located in the United States, the ringing tone cadence as per ITU specifications [21] was 2 seconds ring and 4 seconds silence. A single ring lasted for a duration of 6 seconds.

In order to compute the estimate of wangiri scam calls in our honeypot, we identified any calls that were disconnected before being answered. Next, we computed the fraction of these calls which disconnected from the calling side before the beginning of the second ring — all the calls that disconnected at or before 6 seconds after the call setup. Since a successful wangiri scam involves an International or a premium rate number as the caller ID, we also analyzed the caller ID for all calls disconnected on or before 6 seconds from the call attempt.

**Finding 14**: *We found no concrete evidence of wangiri scams* We found that there were 3,213 calls among all the calls which were prematurely disconnected within 6 seconds. We analyzed the caller ID for calls that were disconnected before answering and observed that there were 29 unique instances of numbers not matching the standard NANP format and were likely a premium rate number used for Wangiri scams. There were 4 invalid caller IDs (e.g. "Restricted, *86") and 2,296 numbers matched the NANP format. Since we found that the caller IDs for these calls did not match well-known wangiri NPA — 900, 976 or other Caribbean countries, we report that there were no instances of wangiri scams observed in our honeypot.

## 4.7 Call Audio Characteristics

Among all the data collected in our honeypot, call audio is crucial in understanding the intent of the call. As explained in Section 3.3, we record and store call audio from unsolicited calls on a subset of our lines. Now, we discuss the characteristics of call audio collected in our honeypot.

Some robocalls have a pre-recorded message while other calls have large sections of audio that are silent. In situations where an actual person dialed one of our inbound lines, it is typical for the user to wait for a response from our side to continue the conversation and hang up after some time. To categorize such calls, we calculate the duration of call

recording which has audio and the duration for which there is silence. These two values help us identify the calls which have a large fraction of audio, which are clear indications of a robocall.

To measure the amount of audio in a call recording, we use py-webrtcvad, [7], a Python interface for WebRTC VAD project. [8] By performing this pre-processing step, we identify and measure the position and duration of non-speech signals. Using these measurements, we compute the total audio duration, the total silence duration and percentage of audio for every call recording. We empirically select two thresholds to determine the calls which have significant amount of audio in the recording — calls should have at least 5 seconds of pure audio and at least 10 % of the entire call should be pure audio. We prune the calls which do not meet these two thresholds before we perform campaign identification using call audio.

## 5 Campaign Identification

In this section, we describe common traits of robocalling and spam calling operations and how we exploit this similarity to develop a clustering algorithm to identify campaigns. We note that number rotation eliminates the possibility of using the calling number to group similar calls.

While number rotation is simple and inexpensive, using significantly different audio prompts for each call is computationally and economically expensive for the caller. Our key insight is that a specific operation will use the same audio to make unsolicited calls, and similarity allows us to group calls with similar audio to identify a group of calls as a *campaign*.

In order to group similar calls, we use raw audio signals present in the call recording to generate audio fingerprints and use these fingerprints to cluster similar audio files. While other researchers [16, 22, 23] have applied Natural Language Processing (NLP) and Machine Learning techniques to audio transcripts in order to analyze calls and cluster them, such techniques involve error and loss of information during transcription. Our audio fingerprinting based clustering approach is versatile and has numerous advantages as described below.

First, our approach is language and speaker agnostic, allowing us to process calls in any language without any modification to our pipeline. Second, our clustering approach is capable of matching audio files which are not identical, but have significant portions of audio that *are* identical. This is important in our case because many campaigns use text-to-speech systems to dynamically insert the name of the called party as part of the robocall. For example, a sample audio snippet could be *"Hello* ***<name>****, this is a call from the Social Security Administration."* Third, our specific technique is resistant to noise, compression, and packet loss.

[7] https://github.com/wiseman/py-webrtcvad
[8] https://webrtc.org/

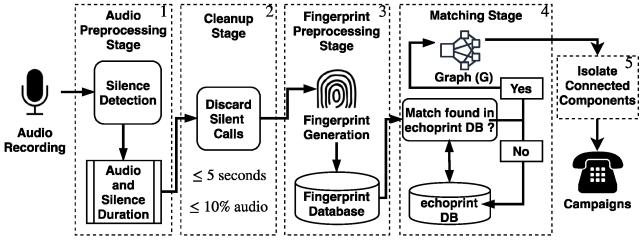


Figure 3: Robocalling Campaign Identification Process through a Five Stage Audio Clustering Pipeline

It is important to keep in mind that what we are characterizing as campaigns is audio, not operators. Multiple operators may collude and use the same audio files as one campaign. Likewise, a single operator might use many different audio files, each a different campaign.

### 5.1 Fingerprinting and Clustering

The architecture of our multi-stage audio clustering pipeline is shown in Figure 3. First, the recorded phone calls go through the audio preprocessing stage which computes the amount of audio in each call recording, as explained in Section 4.7. In stage two, the cleanup stage, we discard any audio files with less than or equal to 5 seconds of audio or less than or equal to 10% of audio, since we are unable to group silent audio into particular campaigns. These two threshold values were empirically determined based on how long it took the authors to convey a single meaningful sentence.

Thirdly, the fingerprint preprocessing stage takes each audio sample as the input, generates the fingerprint of the audio file and stores it in the fingerprint database. In the context of this paper, a *fingerprint* [24, 25] refers to a compact representation of a file such that, with high probability, the fingerprints of two similar files are similar (but not necessarily equal), and the fingerprints of two non-similar files are different. Such fingerprinting techniques are applied to audio files [26] to index songs and perform real-time audio search (e.g., Shazaam [27]). We use audio fingerprinting techniques to identify similar call recordings and cluster them together to identify robocalling campaigns.

We use `echoprint` [28], an open source framework for audio fingerprinting and matching. We choose `echoprint` instead of other audio fingerprinting frameworks since it uses a robust fingerprinting primitive that is well suited for phone call recordings. Since we do not claim the design of `echoprint` as a contribution, we discuss its design and operation in detail in Appendix A. We use raw audio for all the above computation. Using a lossless Waveform Audio File Format (WAV) to store call audio instead of a lossy compressed format like MP3 reduces the probability of error [28] in `echoprint`. Using WAV files and discarding silent audio

calls, as done in stage 2, significantly improves the performance of `echoprint`.

Fourthly, the fingerprints of the filtered audio files go through the matching stage. We query the echoprint database for each new audio fingerprint to check if there is a similar audio file already in the database. If there are no matches, then we add the current audio fingerprint to the database. If we find a match, then we add an edge between the two audio files, where each node represents an audio fingerprint. These nodes and edges are a part of an undirected graph G.

After processing all the audio fingerprints, the undirected graph G has nodes with edges that connect similar audio files. The final stage identifies the connected components of G, where each connected component is a robocalling campaign.

## 5.2 Clustering Evaluation

It is important to evaluate our clustering methodology. However, precision in this context is not clearly defined. To evaluate precision, we define and compute two custom metrics — cluster perfection and intra-cluster precision — to measure the effectiveness of our audio-based clustering methodology. Cluster perfection is defined as the ratio of the number of clusters without misplaced calls to total number of clusters analyzed. Intra-cluster precision is defined as the mean of the ratio of number of correctly placed calls in the cluster to the total number of calls in the cluster. We note that computing recall is impossible given no ground truth on the total count of campaigns in our data.

We use the Industry Robocall Blocking Dataset to evaluate our methodology, since we already have good quality transcript for these calls, as explained in Section 3.5 We thus used the transcripts to assist in labeling correct clustering assignment. We randomly select 20,000 audio samples from the Industry Robocall Blocking Dataset and apply our clustering pipeline. We identified 1,188 clusters and clustered a total of 8,290 audio samples. Out of all these clusters, we selected 30 random clusters and manually listened to a total of 160 audio samples to compute Cluster Perfection and Intra-cluster Precision. We found that there were 2 clusters among the 30 clusters with at least one misplaced call in each of them, resulting in an overall Cluster Perfection rate of 93.33%. The overall Intra-cluster Precision for these 30 clusters was 96.66%.

## 5.3 Campaign Characterization

In this subsection, we characterize campaigns identified using our clustering mechanism. We apply the campaign identification methodology described above to our data set of call recordings collected from our honeypot and identify robocalling campaigns operating in the real world. We define and compute metrics which help us characterize the robocalling campaigns systematically.

**Finding 15**: *91,796 (62.75%) call recordings did not have sufficient amount of audio to be considering for clustering.* We found that 61,528 (42.05%) call recordings had less than 1% audio in the entire duration of the call. Furthermore, 70,916 (48.47%) calls had a total duration of less than one second. A possible explanations for a large fraction of silent calls could be that the campaigns are interested in identifying the phone numbers which are active and are capable of answering a phone call. Another reason could be that the campaigns use voice activity detection features that triggers the payback of a recorded message once the calling side is confident that the call has been answered by an actual person. Since we used a simple greeting while answering a phone call and remain silent post the greeting message, such call answering behavior may not be categorized as a live human in sophisticated outbound calling campaign systems.

It is practically infeasible to convey meaningful information in such short duration and by using a small fraction of speech throughout the call. Also, it is unlikely for an active caller who may have mis-dialed the called number, to disconnected within a fraction of a second after we answer the call. At the outset, such a large number of call audio recordings not containing substantial amount of audio may seem surprising. This high rate may be explained by hit list generation.

Additionally, we observed that few calls (0.01 %) among all the recorded calls were disconnected by our honeypot, which was configured to terminate the call after 60 seconds. The rest were disconnected by the calling side. This observation indicates that a 60 second recording duration is sufficient to record significant portions of unsolicited phone calls.

After filtering out the calls which lack substantial audio to be clustered into a campaign, we performed clustering to identify similar audio as described before in Figure 3.

**Finding 16**: *We found that out of 54,504 call recordings with substantial audio content, 34,150 ( 62.65%) call recordings were identified to be a part of one of the many campaigns.* Of all the calls we processed, we observed that 62.65% were grouped into one of the campaigns. Such high percentage of calls being grouped into clusters indicate that our clustering approach is capable of identifying campaigns and is successful in grouping similar calls into clusters. By analyzing complete campaigns we give providers the tools to choose which operations to target and help them find their weakest points. For example by doing traceback only on the calls in a campaign that are originated by peers.

**Finding 17**: *We discovered 2,687 unique robocalling campaigns operating in the wild.* The largest campaign cluster had 6,055 unique call recording with an average call duration of 47.71 seconds. The calls in this top campaign had an average of 84.88% audio content, which signifies that the campaign was indeed playing a dense recorded message. Furthermore, the average cluster size of the top five campaigns was 2,372.2,

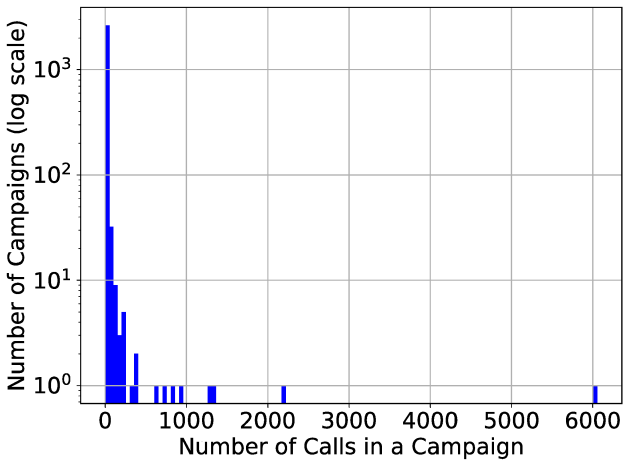


Figure 4: This campaign size histogram indicates we received only a few calls from most campaigns. We received fewer than 27 calls from 95% of campaigns. The largest campaign had 6,055 calls.

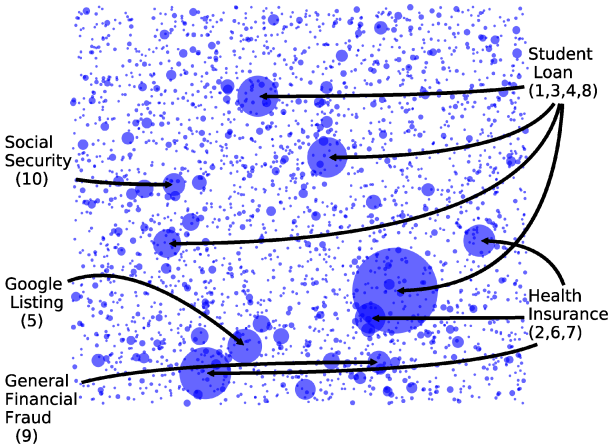


Figure 5: Top 10 Robocalling Campaigns with Radius of the Circle indicating the Relative Campaign Size

which reaffirms our key insight — campaigns that operate at scale reuse the same audio prompts or use audio prompts with slight modifications.

**Finding 18**: *We observed that on an average, a campaign has 12.70 calls.* As shown in Figure 4, we can infer that among all the 2,687 campaigns, a large fraction of campaigns were relatively small in size and a few campaigns have significantly large size.

## 5.4 Campaign Metrics

To systematically evaluate various operational characteristics of the campaigns, we define and calculate metrics to measure the behavior of robocalling campaigns.

**Campaign Size**: Number of calls in each campaign, where a campaign is represented by a cluster of audio recordings.

**Source Distribution**: Ratio of the count of unique caller ID used by the campaign to the campaign size. A 100% source distribution indicates that the campaign used a different caller ID for every call. This metric quantifies the rate at which campaigns spoof caller ID or rotate between calling numbers.

**Spread**: Ratio of the count of unique destination numbers to the campaign size. A 100% Spread indicates that every call from this campaign was to a different inbound line. This metric helps us understand if a campaign is targeting a specific set of inbound lines or tends to distribute calls across a wide range of called numbers.

**Toll-Free Number Usage**: A count of unique toll-free numbers used as the caller ID.

**NPA-NXX Matching Percentage**: Calls which had identical NPA and NXX for calling and called numbers. This is a measurement of neighbor spoofing.

After we defined various metrics, we compute them for each of the 2,687 campaigns. Now, we interpret the metrics to understand how these campaigns differ from each other.

**Finding 19**: *Robocalling campaigns had an average source distribution of 84.17%, which indicates that most campaigns use a large pool of numbers as caller ID.* We observed that the largest robocalling campaign with a campaign size of 6,055 had a Source Distribution of 99.93%. The top 10 campaigns had an average source distribution of 95.50%. Such high source distribution rate indicates that the campaigns are likely spoofing the caller ID. If the campaign is not spoofing caller IDs, then the campaign might own a large pool of phone numbers using which it generates unsolicited phone calls. The findings from the source distribution indicate that well-known call blocking techniques that use allowlists or denylists will not effectively detect or block calls from many campaigns. In future work we hope to analyze the distribution and relative usage of lines by campaigns, and in so doing potentially examine patterns that could be used to predict and block robocalls based on their line rotation strategies.

**Finding 20**: *Robocalling campaigns had an average spread of 78.30% with a few top campaigns targeting specific inbound lines.* We observed that the top campaign had a spread of 19.60%, which indicates that there were multiple calls from the same campaign to a set of inbound line. Such behavior could also indicate that the campaign is using a list of phone number to target their calls. It could also indicate that they selectively target the inbound lines which answer the previous calls made by the campaign. If so, the number of campaigns using this technique must be small in order to be consistent with finding 9. An average spread of 78.30% indicates that most campaigns target a wide range of phone numbers. In future work we hope to analyze the distribution and relative usage of lines by campaigns, and in so doing potentially examine patterns that could be used to predict and block robocalls

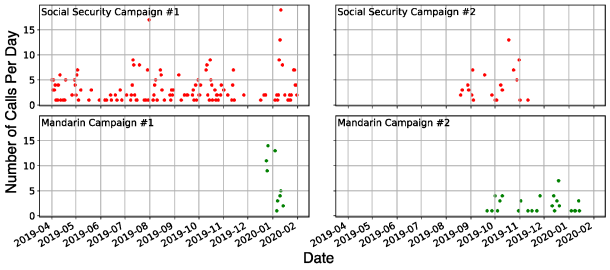


Figure 6: Many of our case study scam campaigns operate over long periods

based on their line rotation habits.

**Finding 21**: *We find that 2.86% (77) campaigns used neighbor spoofing by matching the NPA-NXX of the calling number to the called number.* Of all the campaigns that did employ NPA-NXX based neighbor spoofing, we found that on an average 48.5 % of calls made by these campaigns used neighbor spoofing. There were 14 (0.52%) campaigns with 100% neighbor spoofing rate, indicating that they matched the NPA-NXX for every call. These results indicate that operators regularly evolve spoofing strategies within a single campaign. This finding describes the neighbor spoofing behavior observed among recorded phone calls which belong to a specific campaign, whereas Finding 10 describes an aggregate view of neighbor spoofing among all calls made to our honeypot.

## 5.5 Case Studies

To get an overview about the contents of a cluster, we randomly selected 3 calls from the top 40 clusters and listened to them. This allowed us to label the top clusters as seen in Figure 5 and also helped us to selectively delve into campaigns with unique characteristics. We discovered that of the different telemarketing campaigns, many of them use potentially misleading tactics to encourage victims to engage.

**Finding 22**: *We uncovered two fraudulent robocalling campaigns which impersonated the Social Security Administration (SSA) office, a United States agency.* Our honeypot discovered two separate large scale fraudulent campaigns which clearly violates multiple federal and state laws. Both these campaigns used different audio recordings. The first SSA Campaign (SSA Campaign #1) was the 10th largest campaign in our honeypot with a campaign size of 396. This campaign extensively used 224 unique toll-free numbers as the caller ID to generate unsolicited calls. We observed that this first SSA Campaign operated throughout the duration of our study – April 2019 to February 2020. The second SSA Campaign (SSA Campaign #2) had a campaign size of 75 and operated from August 2019 to November 2019.

We observe that, IRS impersonation (anecdotally one of the most common scams) has given way to different tactics focused on immigration and social security. We suspect that these changes have arisen because taxes are seen as a seasonal issue where other issues are relevant year-round.

**Finding 23**: *We observed that SSA campaigns prefer to use toll-free numbers as the caller ID and are highly targeted to specific users.* We found that the SSA Campaign #1 used 224 unique toll-free numbers with a source distribution of 89.39%, which indicates that only a few calls reused a caller ID. This campaign had a spread of 46.21%. The SSA Campaign #2 also extensively used a pool of 25 unique toll-free numbers. This campaign has an overall source distribution of 100% and a spread of 29.33%. Such low spread indicates that both the campaigns were selective in targeting specific inbound lines, and therefore called the same inbound lines multiple times. SSA Campaigns are known to target specific segments of population who are more vulnerable than the rest [29].

**Finding 24**: *We uncovered two large scale robocalling campaigns that selectively target the Mandarin speaking Chinese population in North America.* Our campaign identification mechanism uncovered two unique robocalling campaigns that operated in Mandarin and in turn was targeted towards Chinese population in the United States. Each campaign had a campaign size of 62 and 51. Both the campaigns impersonated the Chinese Consulate. The first campaign threatened the callers that there was an important document which had expired, and it needed immediate attention of the caller to press a specific digit. The second campaign mentioned that the caller had an urgent message which was time sensitive.

# 6 Discussion

**The future of robocall mitigation**: Current robocall mitigation techniques use caller ID and other heuristics to identify suspected robocalls. Using call traceback [30, 31] to investigate even a fraction of such suspicious calls is time consuming and does not scale well for the provider. Instead, providers can operate their own honeypots and use the campaign identification technique demonstrated in our paper. Providers can systematically identify fraudulent and abusive robocalling campaigns and surgically target the source of such operations. By prioritizing the takedown of specific campaigns, providers can better protect their subscribers.

**Will new initiatives and regulations reduce unsolicited calls?**: To improve enforcement against unsolicited calls, the United Stated passed the TRACED Act [32] into law on December 31st, 2019. Among other things, this act mandates the deployment of STIR/SHAKEN within a certain period and increases penalties for illegal calls. Unfortunately, by the time the regulatory agencies impose penalties on robocalling operations [33], the perpetrators have already generated billions of robocalls. We do not yet know if STIR/SHAKEN will be effective in addressing the problem of unsolicited phone calls, especially because calls that transit any TDM network will be unauthenticated. During our study, no providers were passing STIR/SHAKEN authentications to our provider. Therefore,

our data does not yet indicate if this mechanism will be effective or not.

**Recommendation to the public**: As explained in Section 4.3, we found no statistically significant effect of answering phone calls on the average number of unsolicited phone calls received. Despite this finding, we suggest the general public should still use caution in answering unsolicited calls.

**Limitations**: Like all measurement studies, our work does have some limitations. First, because we do not do any seeding of our numbers, our results may be biased towards campaigns that dial at random. However, the low spread values of our top campaigns indicate that some of our campaigns indeed specifically targeted our lines. Second, our estimates of voicemail spam, wangiri, and neighbor spoofing are based on heuristics, and may be subject to false positives. In particular, our neighbor spoofing estimates assume that callers are not purchasing lines to match the caller, and, given the difficulty of such an operation, we believe this is unlikely. Moreover, in general our methods cannot distinguish between "legitimate" line rotation and spoofing. Finally, because we do not analyze the content of our campaigns, we do not estimate how many of our unsolicited calls fall into the "good" category (e.g., public service announcements) and leave them for future work.

## 7 Related Work

Adversaries continue to thrive [34] in the era of modern telephone networks. Even though researchers [35] have been trying to make telephone networks more secure, end users are constantly bombarded with spam calls [3, 36] and robocalls. Some of the previously proposed techniques to combat spam and fraud [37] in telephone networks employ graph analysis [38–40], use decoys [41], apply machine learning [22, 42–45] and clustering techniques [46, 47]. Other researchers associate a custom metric for the calling number, like a trust value [48] or a reputation score [49] to detect malicious callers.

The absence of end-to-end mutual authentication in phone networks makes caller ID spoofing trivial. Tu et al. [50] demonstrated that spoofed caller ID is a key factor in tricking victims into revealing their private information, like their Social Security Number. Caller ID spoofing also allows the adversaries to operate without the fear of being tracked. To address this issue, researchers have proposed in-band authentication techniques [51], pre-call authentication [52], improving core SS7 protocol standards [53–55], developed mobile applications [56, 57],initiating a call [58] to the calling party during the ringing state, using a trusted third party [59] and coupling SMS with call timing [60] to detect caller ID spoofing. The IETF's STIR working group [12] has recently proposed the SHAKEN [61] framework which uses PASSporTs [62] and certificates [63] to authenticate caller ID [64] in SIP networks. But, these standards do not address the challenges in large segments of non-SIP, TDM and analog circuits which are still operational. By building on top of the Public Key Infrastructure (PKI) ideology, SHAKEN/STIR [65, 66] standards inherit the risks of PKI [67] system designs.

Due to the inherent closed architecture of telephone networks, it is extremely challenging to collect real-world data about how adversaries operate in the wild. Lack of data further prevents us from applying spam detection and mitigation techniques popular in email [68] and SMS [69–75] ecosystems to telephone networks. To collect data and gain insights about how adversaries operate, researchers have scraped websites for audio transcripts [76, 77], used online text-to-speech services to mimic robocallers [78] and generated calls in a lab-controlled environment [42, 79]. We believe that such strategies are inadequate in representing a constantly evolving real-world adversary. Also, user reported details could be biased, inaccurate and under-represented. Numerous researchers [3, 36, 68] have emphasized the need for collecting and analyzing data from actual phone networks, which can inturn help in the development of robust mitigation techniques. Techniques presented by Balasubramaniyan et al. [80] can be useful to study the network path of a phone call as part of our future work. Actively engaging with the caller [13, 50, 77] has been an effective approach to gain deeper insights about the adversary's operational characteristics.

Honeypots [81] have served as a mechanism to collect data about adversaries. Honeypots have been used to study worms [82], email spam campaigns [83], SMS spam [71], social media campaigns [14, 84, 85], telephone networks [6, 86] and much more. Previously developed telephony honeypots have certain limitations and inherent assumptions. Gupta et al. [6] and Li et al. [43] do not collect and process call audio, while Balduzzi et al. [16] restrict themselves to specific geographic regions or languages, and Sahin and Francillon [17] use a small number of clean numbers. Previous work [6, 16, 22, 23, 43] either used transcripts to identify clusters of calls or did not account for caller ID spoofing, which is prevalent in an adversarial telephone network settings. Our data collection and campaign identification techniques extend far beyond each of them. The techniques proposed and used in this paper are agnostic to caller ID spoofing and language of the robocall. None of the prior work collect and analyze the call meta-data, call audio content and signaling information as a whole.

## 8 Conclusion

Robocalls and other forms of unsolicited phone calls have plagued the telephone network. Such calls are a long-standing problem to all people who use a phone. Despite anecdotal evidence of the prevalence of such calls, accurate information on the frequency of these calls is largely unknown. Through a data-driven study, we provide details about the scale at which unsolicited calling campaigns operate in the North American phone network. By experimentation and statistical validation, we find no evidence that answering unsolicited calls increases

the number of such calls received. We develop mechanisms to characterize voicemail spam, wangiri scam and different forms of caller ID spoofing techniques. We develop, evaluate and apply a robust campaign identification technique using call audio, and uncover 2,687 unique robocalling campaign in the wild. Based on our observation, we discuss the state of existing detection and mitigation techniques and call for more data driven studies of the phone network.

## Acknowledgments

The authors would like to thank Bandwidth Inc. and Tom Soroka, Dir. Fraud Mitigation at Bandwidth Inc. for their support and for providing VoIP service and phone numbers for the honeypot. We thank Aaron Foss and Nomorobo for providing call recordings and transcripts. We would like to thank Dr. William Enck and Bihan Zhang. We also thank our anonymous reviewers for their helpful comments. This material is based upon work supported by the National Science Foundation under grant number CNS-1849994. 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] J. Davidson. Here's How Many Internet Users There Are. *Time*, May 2015.

[2] Y. Wang. More People Have Cell Phones Than Toilets, U.N. Study Shows. *Time*, 2013.

[3] H. Tu, A. Doupé, Z. Zhao, and G. Ahn. SoK: Everyone Hates Robocalls: A Survey of Techniques Against Telephone Spam. In *IEEE Symposium on Security and Privacy*, 2016.

[4] S. Poonam and S. Bansal. Scare and sell: Here's how an Indian call centre cheated foreign computer owners. *Hindustan Times*, May 2017.

[5] K. P. Erb. Dozens Arrested In IRS Phone Scam Call Center Raids. *Forbes*, October 2016.

[6] 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*, 2015.

[7] C. Valli and M. A. Lawati. Developing VoIP Router Honeypots. In *Security and Management*, 2010.

[8] M. Nassar, R. State, and O. Festor. VoIP Honeypot Architecture. In *Integrated Network Mgmt.* IEEE, 2007.

[9] R. do Carmo, M. Nassar, and O. Festor. Artemisa:An open-source honeypot back-end to support security in VoIP domains. In *IEEE Integrated Network Mgmt.*, 2011.

[10] A. Costin, J. Isacenkova, M. Balduzzi, A. Francillon, and D. Balzarotti. The role of phone numbers in understanding cyber-crime schemes. 2013.

[11] FCC Enforcement Bureau. Abramovich Citation and Order, June 2017.

[12] IETF. *Secure Telephone Identity Revisited (STIR)*, 2019.

[13] N. Miramirkhani, O. Starov, and N. Nikiforakis. Dial One for Scam: A Large-Scale Analysis of Technical Support Scams. July 2016.

[14] S. Gupta, G. S. Bhatia, S. Suri, D. Kuchhal, P. Gupta, M. Ahamad, M. Gupta, and P. Kumaraguru. Angel or Demon? Characterizing Variations Across Twitter Timeline of Technical Support Campaigners. *The Journal of Web Science*, 2019.

[15] T. B. Mobarak and A. Han. Method and apparatus for forcing a call to a carrier provided voice mail facility, December 10 2013. US Patent 8,605,869.

[16] M. Balduzzi, P. Gupta, L. Gu, D. Gao, and M. Ahamad. MobiPot: Understanding Mobile Telephony Threats with Honeycards. In *Proceedings of ACM on Asia Conference on Computer and Communications Security*, 2016.

[17] M. Sahin and A. Francillon. On the Effectiveness of the National Do-Not-Call Registries. In *Workshop on Technology and Consumer Protection*, May 2018.

[18] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. SIP: Session Initiation Protocol. RFC 3261, June 2002.

[19] FCC Adopts Rules to Help Block Illegal Robocalls. https://www.fcc.gov/document/fcc-adopts-rules-help-block-illegal-robocalls-0.

[20] FCC Affirms Robocall Blocking By Default to Protect Consumers. https://www.fcc.gov/document/fcc-affirms-robocall-blocking-default-protect-consumers-0.

[21] ITU. Various Tones Used in National Networks. https://www.itu.int/ITU-T/inr/forms/files/tones-0203.pdf.

[22] A. Marzuoli, H. A. Kingravi, D. Dewey, and R. Pienta. Uncovering the Landscape of Fraud and Spam in the Telephony Channel. In *15th IEEE International Conference on Machine Learning and Applications*, Dec 2016.

[23] S. Pandit, R. Perdisci, M. Ahamad, and P. Gupta. Towards Measuring the Effectiveness of Telephony Blacklists. Annual Network and Distributed System Security Symposium, NDSS, 2018.

[24] U. Manber. Finding Similar Files in a Large File System. In *Proceedings of USENIX Technical Conference*, '94.

[25] S. Schleimer, D. S Wilkerson, and A. Aiken. Winnowing: Local Algorithms for Document Fingerprinting.

[26] C. Brinkman, M. Fragkiadakis, and X. Bos. Online music recognition: the Echoprint system.

[27] A. Wang. An Industrial Strength Audio Search Algorithm.

[28] D. P. W. Ellis, B. Whitman, and A. Porter. ECHOPRINT - An Open Music Identification Service. In *Proceedings of the 12th International Society for Music Information Retrieval Conference*, 2011.

[29] M. Bidgoli and J. Grossklags. "Hello. This is the IRS calling.": A case study on scams, extortion, impersonation, and phone spoofing. In *2017 APWG Symposium on Electronic Crime Research*, 2017.

[30] D. Frankel. Senate Hearing on Combating Robocall Fraud. https://www.aging.senate.gov/imo/media/doc/SCA_Frankel_7_17_19.pdf, 2019.

[31] USTELECOM Industry Traceback Group Report 2019. https://www.ustelecom.org/wp-content/uploads/2020/01/USTelecom_ITG_2019_Progress_Report.pdf.

[32] FCC. TRACED ACT or FS.151 - Pallone-Thune Telephone Robocall Abuse Criminal Enforcement and Deterrence Act. https://www.congress.gov/bill/116th-congress/senate-bill/151.

[33] FCC Proposes Record $225 Million Fine for 1 Billion Spoofed Robocalls. https://www.fcc.gov/document/fcc-proposes-record-225-million-fine-1-billion-spoofed-robocalls.

[34] M. A. Ali, M. Ajmal Azad, M. P. Centeno, F. Hao, and A. van Moorsel. Consumer-facing technology fraud: Economics, attack methods and potential solutions. *Future Generation Computer Systems*, 2019.

[35] A. D. Keromytis. A Comprehensive Survey of Voice over IP Security Research. *IEEE Communications Surveys Tutorials*, 2012.

[36] M. Sahin, A. Francillon, P. Gupta, and M. Ahamad. SoK: Fraud in Telephony Networks. In *2017 IEEE European Symposium on Security and Privacy*, April 2017.

[37] M. Sahin and A. Francillon. Over-The-Top Bypass: Study of a Recent Telephony Fraud. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 2016.

[38] H. K. Bokharaei, A. Sahraei, Y. Ganjali, R. Keralapura, and A. Nucci. You can SPIT, but you can't hide: Spammer identification in telephony networks. In *Proceedings IEEE INFOCOM*, April 2011.

[39] V. S. Tseng, J. Ying, C. Huang, Y. Kao, and K. 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*. ACM, 2015.

[40] M. Swarnkar and N. Hubballi. SpamDetector: Detecting spam callers in Voice over Internet Protocol with graph anomalies. *Security and Privacy*, 2019.

[41] S. M. A. Salehin and N. Ventura. Blocking Unsolicited Voice Calls Using Decoys for the IMS. In *2007 IEEE International Conference on Communications*.

[42] Y. Wu, S. Bagchi, N. Singh, and R. Wita. Spam detection in Voice-over-IP Calls through Semi-supervised Clustering. In *2009 IEEE/IFIP International Conference on Dependable Systems Networks*, 2009.

[43] H. Li, X. Xu, C. Liu, T. Ren, K. Wu, X. Cao, W. Zhang, Y. Yu, and D. Song. A Machine Learning Approach to Prevent Malicious Calls over Telephony Networks. In *IEEE Symposium on Security and Privacy*, May 2018.

[44] A. Leontjeva, M. Goldszmidt, Y. Xie, F. Yu, and M. Abadi. Early Security Classification of Skype Users via Machine Learning. In *Proceedings of the ACM Workshop on Artificial Intelligence and Security*, 2013.

[45] K. Ji, Y. Yuan, R. Sun, L. Wang, K. Ma, and Z. Chen. Abnormal Telephone Identification via an Ensemble-based Classification Framework. In *Proceedings of the ACM Turing Celebration Conference - China*, 2019.

[46] S. Subudhi and S. Panigrahi. Use of Possibilistic Fuzzy C-means Clustering for Telecom Fraud Detection. In *Computational Intelligence in Data Mining*, 2017.

[47] J. Zhang, J. Wang, Y. Zhang, J. Xu, and H. Wu. *A Novel SPITters Detection Approach with Unsupervised Density-Based Clustering*. 2018.

[48] N. Chaisamran, T. Okuda, G. Blanc, and S. Yamaguchi. Trust-Based VoIP Spam Detection Based on Call Duration and Human Relationships. In *2011 IEEE/IPSJ International Symposium on Applications and the Internet*.

[49] M. A. Azad, R. Morla, J. Arshad, and K. Salah. Clustering VoIP caller for SPIT identification. *Security and Communication Networks*, 2016.

[50] H. Tu, A. Doupé, A. Zhao, and G. Ahn. Users Really Do Answer Telephone Scams. In *28th USENIX Security Symposium*, 2019.

[51] B. Reaves, L. Blue, and P. Traynor. AuthLoop: End-to-End Cryptographic Authentication for Telephony over Voice Channels. In *25th USENIX Security Symposium*, 2016.

[52] B. Reaves, L. Blue, H. Abdullah, L. Vargas, P. Traynor, and T. Shrimpton. AuthentiCall: Efficient Identity and Content Authentication for Phone Calls. In *26th USENIX Security Symposium*, 2017.

[53] H. Tu, A. Doupe, Z. Zhao, and G. Ahn. Toward standardization of authenticated caller id transmission. *IEEE Communications Standards Magazine*, 2017.

[54] H. Tu, A. Doupé, Z. Zhao, and G. Ahn. Toward Authenticated Caller ID Transmission: The need for a Standardized Authentication Scheme in Q.731.3 Calling Line Identification Presentation. In *ITU Kaleidoscope: ICTs for a Sustainable World*, 2016.

[55] H. Tu. *From Understanding Telephone Scams to Implementing Authenticated Caller ID Transmission*. PhD thesis, Arisona State University, 2017.

[56] I. Sherman, J. Bowers, K. McNamara Jr, J. Gilbert, J. Ruiz, and P. Traynor. Are You Going to Answer That? Measuring User Responses to Anti-Robocall Application Indicators. Annual Network and Distributed System Security Symposium, 2020.

[57] G. W. Edwards, M. J. Gonzales, and M. A. Sullivan. Robocalling: STIRRED AND SHAKEN! - An Investigation of Calling Displays on Trust and Answer Rates. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 2020.

[58] H. A. Mustafa, W. Xu, A. Sadeghi, and S. Schulz. You Can Call but You Can't Hide: Detecting Caller ID Spoofing Attacks. In *44th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, 2014.

[59] J. Li, F. Faria, J. Chen, and D. Liang. A Mechanism to Authenticate Caller ID. In *Recent Advances in Information Systems and Technologies*, 2017.

[60] H. Mustafa, W. Xu, A. Sadeghi, and S. Schulz. End-to-End Detection of Caller ID Spoofing Attacks. *IEEE Transactions on Dependable & Secure Computing*, 2018.

[61] C. Wendt and M. Barnes. Personal Assertion Token (PaSSporT) Extension for Signature-based Handling of Asserted information using toKENs (SHAKEN). RFC 8588, 2019.

[62] C. Wendt and J. Peterson. PASSporT: Personal Assertion Token. RFC 8225, February 2018.

[63] J. Peterson and S. Turner. Secure Telephone Identity Credentials: Certificates. RFC 8226, February 2018.

[64] J. Peterson, C. Jennings, E. Rescorla, and C. Wendt. Authenticated Identity Management in the Session Initiation Protocol (SIP). RFC 8224, February 2018.

[65] J. McEachern and E. Burger. How to shut down robocallers: The STIR/SHAKEN protocol will stop scammers from exploiting a caller ID loophole. *IEEE Spectrum*, 2019.

[66] M. Chiang and E. Burger. An Affordable Solution for Authenticated Communications for Enterprise and Personal Use. In *2018 IEEE 8th Annual Computing and Communication Workshop and Conference*, 2018.

[67] C. Ellison & B. Schneier. Ten risks of PKI: What you're not being told about public key infrastructure. 2000.

[68] P. Patankar, G. Nam, G. Kesidis, and C. R. Das. Exploring Anti-Spam Models in Large Scale VoIP Systems. In *28th IEEE International Conference on Distributed Computing Systems*, 2008.

[69] B. Reaves, L. Blue, D. Tian, P. Traynor, and K. R.B. Butler. Detecting SMS Spam in the Age of Legitimate Bulk Messaging. In *Proceedings of the 9th ACM Conference on Security & Privacy in Wireless and Mobile Networks*, 2016.

[70] T. A. Almeida, J. M. G. Hidalgo, and A. Yamakami. Contributions to the Study of SMS Spam Filtering: New Collection and Results. In *Proceedings of the 11th ACM Symposium on Document Engineering*, 2011.

[71] N. Jiang, Y. Jin, A. Skudlark, and Z. Zhang. Greystar: Fast and Accurate Detection of SMS Spam Numbers in Large Cellular Networks Using Gray Phone Space. In *The Proceedings of USENIX Security Symposium*, 2013.

[72] H. Siadati, T. Nguyen, P. Gupta, M. Jakobsson, and N. D. Memon. Mind your SMSes: Mitigating Social Engineering in Second Factor Authentication. 2017.

[73] B. Srinivasan, P. Gupta, M. Antonakakis, and M. Ahamad. Understanding Cross-channel Abuse with SMS-spam Support Infrastructure Attribution. In *European Symposium on Research in Computer Security*, 2016.

[74] B. Reaves, N. Scaife, D. Tian, L. Blue, P. Traynor, and K. R. B. Butler. Sending Out an SMS: Characterizing the Security of the SMS Ecosystem with Public Gateways. In *2016 IEEE Symposium on Security and Privacy*, 2016.

[75] S. Gupta, P. Gupta, M. Ahamad, and P. Kumaraguru. Exploiting Phone Numbers and Cross-application Features in Targeted Mobile Attacks. In *Proceedings of the 6th Workshop on Security and Privacy in Smartphones and Mobile Devices*, 2016.

[76] Q. Zhao, K. Chen, T. Li, Y. Yang, and X. Wang. Detecting Telecommunication Fraud by Understanding the Contents of a Call. *Cybersecurity*, 2018.

[77] M. Sahin, M. Relieu, and A Francillon. Using Chatbots Against Voice Spam: Analyzing Lenny's Effectiveness. In *Proceedings of the Thirteenth USENIX Conference on Usable Privacy and Security*, SOUPS 2017.

[78] A. Lieto, D. Moro, F. Devoti, C. Parera, V. Lipari, P. Bestagini, and S. Tubaro. "Hello? Who Am I Talking to?" A Shallow CNN Approach for Human vs. Bot Speech Classification. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2019.

[79] P. Kolan and R. Dantu. Socio-technical Defense Against Voice Spamming. *ACM Trans. Auton. Adapt. Syst.*

[80] V. A. Balasubramaniyan, A. Poonawalla, M. Ahamad, M. T. Hunter, and P. Traynor. PinDr0p: Using Single-Ended Audio Features to Determine Call Provenance. In *Proceedings of 17th ACM Conference on Computer & Communications Security*, 2010.

[81] Niels Provos et al. A Virtual Honeypot Framework. In *USENIX Security Symposium*, volume 173, 2004.

[82] D. Dagon, X. Qin, G. Gu, W. Lee, J. Grizzard, J. Levine, and H. Owen. "HoneyStat: Local Worm Detection Using Honeypots". In *Recent Advances in Intrusion Detection*, 2004.

[83] N. Krawetz. Anti-honeypot technology. *IEEE Security and Privacy*, 2004.

[84] E. De Cristofaro, A. Friedman, G. Jourjon, M. A. Kaafar, and M. Z. Shafiq. Paying for Likes?: Understanding Facebook Like Fraud Using Honeypots. In *Proceedings of Conference on Internet Measurement Conference*, 2014.

[85] S. Gupta, D. Kuchhal, P. Gupta, M. Ahamad, M. Gupta, and P. Kumaraguru. Under the Shadow of Sunshine: Characterizing Spam Campaigns Abusing Phone Numbers Across Online Social Networks. In *Proceedings of the 10th ACM Conference on Web Science*, 2018.

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

## A Functionality of `echoprint`

At its core, `echoprint` generates a fingerprint of an audio file based on the time interval between successive amplitude peaks (called *inter-onset interval*). Phonemes in human speech — each unit of sound that distinguishes one word of a spoken language to another — creates amplitude variations in the call audio. Prosody in human speech introduces functions like rhythm and tone to the call audio. These features of human speech allow `echoprint` to generate fingerprints that represents a call audio recording.

To generate a fingerprint, `echoprint` marks the amplitude peaks and computes the inter-onset interval between these peaks across 8 independent frequency bands of the audio file. A combination of (i) inter-onset interval, (ii) the specific frequency band and (iii) time at which the inter-onset interval occurs in the audio file are used to generate a non-cryptographic hash value. Each second of audio generates approximately 48 hash values. Multiple hash values together form the fingerprint of the file. We store the fingerprint in the fingerprint database as a JSON object.

The matching operation of the `echoprint` framework works on the fact that inter-onset interval of similar audio files are identical [28]. When we query the echoprint DB with a new audio fingerprint, `echoprint` framework identifies a list of top 15 audio files which have matching hashes. These matches are sorted, starting with the best match and ending with the worst match. We get the top audio sample in this list as a match if its match score is significantly higher than the match score of all the other matches in the list. Otherwise, `echoprint` does not return a match. If we do not get a match, we add the audio fingerprint to the echoprint DB.

Table 1: Important Dates

| | Dates (dd-mm-yyyy) | |
|---|---|---|
| Name | Start | End |
| Study Duration | 17-02-2019 | 01-02-2020 |
| $t$-test Duration | 17-02-2019 | 04-12-2019 |
| Initial Recording | 31-03-2019 | 01-02-2020 |
| Second Recording | 21-12-2019 | 01-02-2020 |
| Power Outage Downtime | 05-04-2019 | 06-04-2019 |
| Winter Downtime | 29-12-2019 | 04-01-2020 |