2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P) | 978-1-6654-6512-0/23/$31.00 ©2023 IEEE | DOI: 10.1109/EuroSP57164.2023.00011

Understanding, Measuring, and Detecting Modern Technical Support Scams

Jienan Liu
University of Georgia
jienan82@gmail.com

Pooja Pun
University of New Orleans
poojapun90@gmail.com

Phani Vadrevu
University of New Orleans
phani@cs.uno.edu

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

***Abstract*—Technical support scams (TSS) are social engineering attacks that aim to exploit users that have limited knowledge about technology, such as the elderly, causing significant financial loss to vulnerable citizens. The security community has attempted to respond to these web-based scams with different countermeasures. However, to the best of our knowledge, no robust countermeasures have been proposed thus far to defend against modern TSS campaigns that abuse web search engines to inflate their rankings in search results and lure many potential victims.**

To defend against these TSS attacks, in this paper we first study the TSS ecosystem, with particular focus on how modern TSS campaigns are operated and promoted on the web. Then, we capitalize on our findings by proposing a novel detection system named TASR that can be used to differentiate TSS websites from legitimate technical support websites in a *topic-agnostic* way, by leveraging features that capture key traits of how TSS web pages are promoted. Our cross-validation tests show that TASR can detect 94.5% of the TSS links in web search results at a false positive rate of less than 1%, significantly outperforming previous work.

1. Introduction

Technical support scams (TSS) are a class of social engineering attacks that aim to exploit users who tend to have limited knowledge about technology, such as elderly people [23]. In their most prevalent form, these attacks start with the user stumbling upon a TSS web page that pretends to offer legitimate technical support services and persuades the victim to call a support phone number (typically, a toll-free number). The phone call is then answered by a scammer agent who employees social engineering tactics to coerce the victim into paying significant amounts of money for fabricated or unneeded technical services. In aggregate, such scams can cause tens of millions of dollars in monetary loss per year in the USA alone and often have a significant financial impact on vulnerable citizens [23].

Prior work [40] has focused on early versions of TSS attacks that relied on aggressive social engineering tactics. These typically start with the user visiting a web page that embeds malicious JavaScript (JS) code (e.g., delivered via malicious ads embedded in an otherwise legitimate web page). The JS code often exploits a number of subtle browser bugs to "lock" the browser (e.g., with "infinite" `alert()` boxes) and mislead victims into believing that their system (e.g., a computer or mobile device) has critical issues that can only be fixed by calling a purported technical support center. We refer to these attacks as *aggressive TSS*. Over the past few years, the security community responded fairly successfully to aggressive TSS tactics with multiple browser countermeasures. For instance, by searching across web browser bug repositories, we found that several "browser lock" issues have been resolved specially to deal with the social engineering tactics mentioned above (e.g., throttling usage of APIs such as `alert()` [3], [5], [4], `history.pushState()` [8], [11], and script-driven file downloads [7], [6]). Furthermore, ad networks such as Google have also begun to lay out strict content policies that prohibit "fake system dialogs" and other web API abuses [1]. As a result, the early versions of TSS web pages have mostly been relegated to low-ranked websites that use unconscientious ad networks with lax content policies [50].

In response to these countermeasures, TSS scammers have unfortunately evolved to use very different and potentially more insidious tactics. Namely, scammers often build a set of websites that mimic legitimate technical support businesses pretending to offer a variety of services related to fixing printers, popular software suites, email issues, security software, etc. Using *blackhat search-engine optimization* (BHSEO) techniques [38] and search ads, these websites are then promoted on web search engines [45] by artificially inflating their search results rankings related to specific technical support search keywords. We refer to TSS attacks that use this approach as *passive*, because they passively wait for users to stumble upon a TSS website through legitimate web searches. The question of how to mitigate this cunning evolution of TSS pages remains largely unanswered. Consequently, this has driven a growth in scope and size of passive TSS scams, as we will show in later sections.

To the best of our knowledge, no robust countermeasures have been proposed thus far for detecting these modern TSS scams. While search engines such as Google and Microsoft Bing have announced blanket bans on third-party technical support search advertisements [30], [52] such policies have also impacted legitimate technical support service and have caused some backlash [20]. Furthermore, these bans only apply to sponsored search advertisement links and not to organic search results, which can still be poisoned by the TSS scam pages. Also, previous work [45] proposed a text-based TSS website detector as part of a TSS measurement system. However, the proposed text-based detection approach has two major limitations, when used as a defense:

1) Because these modern TSS scam pages delivered

DOI 10.1109/EuroSP57164.2023.00011

via search engines include text content that is intentionally highly similar to benign tech support pages, it is difficult to differentiate between benign and scam tech support websites solely based on text (see Section 6).
2) Furthermore, text-based models will require frequent re-training with a comprehensive set of new scam topic pages, to be able to detect the growing number of TSS topics that are targeted by scammers. In fact, we found that TSS scammers have begun to target many new tech brands and categories beyond the few identified in [45]. New TSS categories include IoT devices (Ring, Nest), financial accounting software (Quickbooks), digital streaming (Netflix, Hulu, Roku, Firestick), cash transfer apps (Chime, Paypal, Cash App) and even airlines (e.g., Delta).

The above observations motivate the need for a *topic-agnostic* detection approach that can capture key traits of modern TSS websites *beyond their text-based content and related keywords*. To this end, in this paper we first study the TSS ecosystem, focusing especially on understanding how modern TSS campaigns are operated and promoted on the web, and then we capitalize on our findings by proposing a novel detection system named TASR (Topic-Agnostic Scam Recognizer) that can be used to differentiate TSS websites from legitimate technical support websites by leveraging features that capture how TSS web pages are promoted on the web.

To study the TSS ecosystem, we collected and analyzed several months of data from underground TSS marketplaces and conducted an approximate cost analysis of the operations of these components in order to understand how the scams can remain profitable. To the best of our knowledge, ours is the first systematic analysis of the TSS ecosystem grounded in actual conversations among different TSS actors captured from social media channels. This analysis revealed that TSS websites that advertise scam phone numbers are typically setup and operated by a group of malicious actors that sell calls to TSS call centers. Such TSS websites serve as a key entry point to the scams and are often promoted using BHSEO techniques to artificially inflate their rank in the web search results even above official-brand support websites (e.g., Apple, Microsoft, HP, etc.). After studying how TSS websites are operated and promoted, we leverage their fundamental traits, such as the way that TSS phone numbers are promoted, the type of backlinks used to inflate web search rankings, etc., to build our *topic-agnostic* TASR detection system. Ultimately, our system could be deployed by web search engines to proactively detect and prune (or at least de-rank) TSS websites from their search results and search ads. In summary, we make the following main contributions:

1) *TSS ecosystem*: We conduct the first systematic study of the TSS ecosystem by collecting and analyzing messages and announcements published in underground TSS marketplaces; this allowed us to understand how TSS are orchestrated and promoted on the web. In addition, we perform an approximate cost analysis to understand how different actors profit from such illegal operations.
2) *TSS websites analysis*: We conduct an analysis of how TSS websites are built to promote TSS phone numbers and what are the topic-agnostic features that are used to inflate their search results rankings.
3) *Defenses*: We propose a novel detection system called TASR that can differentiate between modern TSS websites vs. legitimate technical support websites by leveraging key traits of TSS sites. Our cross-validation tests show that TASR can detect 94.5% of TSS links in search results at a false positive rate of less than 1%. We also released all the source code artifacts and datasets used in this paper[1].
4) *Abuse Disclosure*: We disclosed our findings to both the affected web search engines and social media platforms (Facebook and WhatsApp) used by TSS actors for their underground market communications.

2. TSS Ecosystem

To understand modern TSS operations, in this section we analyze information we collected by investigating TSS underground markets over several months. To the best of our knowledge, ours is the first work to study and report on details related to TSS underground markets.

2.1. Social Media Groups

During our study of TSS, we discovered that there exist several groups on Facebook (e.g., "Tech Support World", "Tech Support Business", etc.) that are actively used by TSS actors to conduct their shady business. Given the egregious nature of the information we found on these groups, it was surprising to see that these were operating openly on Facebook and required no private membership to view their posts. Interestingly, after we joined a few groups that we manually found, Facebook's recommendation engine suggested other similar TSS scammer groups, which we also joined. In total, for this study we joined and collected data from 10 such Facebook TSS groups and performed a detailed analysis of their posts.

While analyzing the Facebook posts, we also came across invitations to join private messaging groups on WhatsApp. We therefore also joined those groups to increase the diversity of information about TSS operations. We observed that a majority of phone numbers that post on these groups (about 71%) are from India, which is not entirely surprising given previous findings that more than 85% of the IP addresses used for remote desktop connections made by TSS scammers were from India [40] and the existence of known TSS call center operators in that country [17]. Unlike Facebook groups, we found that WhatsApp groups have more stringent membership policies, with some group administrators periodically purging members whose phone numbers do not include India's country code. As we used a US phone number, we were booted from some TSS WhatsApp groups but were able to remain part of and collect data from 13 groups.

The results we present are related to a quantitative and qualitative data analysis of posts we collected from 10 TSS Facebook groups in a 30-day period and from 13 TSS WhatsApp groups in a 9-month period between 2020 and 2021. In addition, we also repeated this analysis with data

1. Available at https://github.com/NISLabUGA/TSS_ESP23

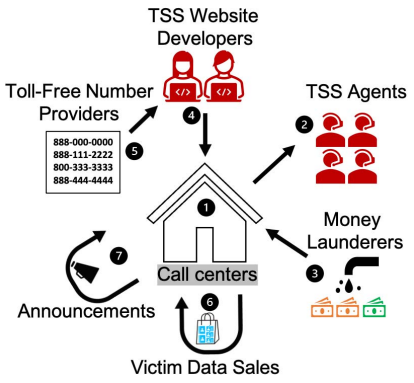


Figure 1: Components of web-based TSS operations (derived by analyzing social media posts).

collected in March 2022. Because the data analysis did require non-negligible manual effort, this latter experiment was performed at a smaller scale. However, the recent data we collected in 2022 shows that TSS-related groups continued to thrive unabated, and the analysis we present in the rest of this section represents how TSS scams are still organized and operated today.

Due to space constraints, a detailed analysis of all the data we collected from TSS groups on Whatsapp and Facebook is presented in Appendix A. To summarize, we collected 22,043 posts from Facebook groups and 43,307 posts from WhatsApp. The TSS Facebook groups included around 7,000 members, on average, with more than 500 authors (i.e., accounts that posted messages) per group. Interestingly, several of these groups were established more than 4 years earlier. This seems to indicate that Facebook is likely not aware of these scam-related activities, even though the posts appear to be in violation of Facebook's content policies [26]. We recently disclosed our findings to Facebook and our technical report was forwarded to their e-crime team who are working on analyzing this issue.

2.2. Overview of TSS Ecosystem Components

A qualitative analysis of the posts we collected from TSS groups revealed a vibrant ecosystem of scammers. Specifically, we found that a majority of the posts had a specific business objective and were used to advertise and sell services to other scammers. We thus learned that the TSS ecosystem is organized into multiple sub-businesses specializing on different TSS operations, as shown in Figure 1 (the arrows indicate the source and intended destination of the posts). Please note that, due to space constraints, the details of how these posts were labeled, including the codebook we used and inter-rater reliability metrics, are presented in Appendix A.

In the following, we provide an overview of each TSS component.

❶ **TSS call centers.** Call centers are the focal point for most of the posts. TSS call centers are responsible for handling the live scam calls from victims, and by analyzing the posts we found that they tend to operate in office spaces that are located in several metropolitan cities in India such as Delhi, Kolkata, Mumbai, Pune, Chandigarh, Hyderabad and Bangalore.

❷ **TSS agents.** TSS call centers need agents who can social-engineer the victims into paying for their pretend services [40]. These agents are the only people in the ecosystem who actually interact with the victims. To hire these agents, call center operators frequently post job ads, often including details such as salary and "benefits" (see Figure 13, Appendix B.1).

❸ **Money launderers.** The FTC has issued multiple warnings that payments extorted by TSS agents from victims can be in unconventional forms like gift cards, due to their ease of use [22] and difficulty to trace [41]. In our analysis, we found frequent posts by some groups of scammers who specialize in laundering gift cards, as well payments via credit cards, online financial services (e.g., PayPal, Venmo, CashApp, Zelle, etc.), wire transfers and even mail services such as FedEx and UPS. These groups sell their services to TSS call center operators.

❹ **TSS webmasters.** To monetize their TSS operations, call centers need to "advertise" their scams and lure potential victims to calling their TSS agents. While multiple vectors can be used to promote the scams (including email spam, social media spam, etc.), we found that a very common tactic in modern TSS is to build and promote websites that mimic legitimate technical support services and prominently display the phone number that victims will need to call. Because such websites are the entry point to many modern TSS campaigns, in this paper we focus on studying their properties.

From our social media posts analysis, we found that TSS websites are typically developed by a group of malicious actors, which we refer to as *TSS webmasters*, who specialize solely on building TSS websites and selling their services to call centers. In fact, from the posts we discovered that it is common for *TSS webmasters* to directly acquire and operate phone numbers used to monetize the scams. This was at first a bit surprising, as we assumed TSS call centers would be directly operating the phone numbers advertised on TSS websites. Instead, we found that *TSS webmasters* often use the TSS phone numbers they promote as a telephony proxy that receives victims' calls and redirects them to a call center, allowing them to *sell victim calls* rather than directly selling website content and web hosting services to the call centers. An example post is presented in Figure 3. The post mentions the targeted brands advertised on the TSS websites to attract victims. It also describes the prices (in Indian Rupees) of each call. Table 7 in Appendix A shows the victim call prices for different brands targeted by the scammers. Most of these calls individually cost between $5 and $10 USD.

❺ **Toll-free number providers.** To maximize the number of calls from potential victims, TSS scammers prefer to use *toll-free* phone numbers, presumably because this makes the phone numbers look more "official" and more similar to legitimate technical support centers for popular product brands. Toll-free numbers are acquired from phone number providers, who appear to act as re-sellers of phone numbers owned by legitimate telephone companies. An example of a Facebook post from such providers is shown in Figure 15 in Appendix B.1 . The post mentions that a new phone number will be provided within 5 minutes, if an existing phone number is blocked.

This is important to keep the TSS operations running, as phone numbers may get blocked by telephone companies [47] after receiving abuse complaints.

Alongside other interesting details, we also found that it is common for TSS call centers to set-up an auxillary **victim data sale ❻** business, for example to enable *refund scams* [24]. In such sales, we observed that it was common practice to explicitly target older aged populations (see Fig. 16 in Appendix B.1) corroborating prior reports [27]. Finally, scammers frequently post **TSS announcements ❼** that they believe will help other scammers. For instance, if they find out that a police raid is going to take place in a particular location, they alert the other group members to help them evade the police.

In Tables 1 and 2, we show the number of social media posts that were made pertaining to the above TSS operation components. It is to be noted that the posts for hiring agents and sales of prior victim data are made by the call centers themselves while the remaining category posts are made by other operators as depicted in Figure 1. Both the tables indicate that posts from money launderers and TSS webmasters are the most popular ones. We observed that these posts are repeated daily in different groups by the post authors. We also saw that these posts are typically made every weekday morning (as per US time zones) in order to potentially attract call centers that are beginning their daily operations. Facebook groups are richer in textual content compared to WhatsApp groups as Facebook allows for "comment threads" to exist below each post. In Table 1, we also report the number of these response comments for each category as it serves as a quantitative indicator of reader interest. We can see again that the response rate for posts from money launderers and TSS webmasters is high indicating a high daily demand from call centers for these services. Interestingly, the relative response rate for hiring TSS agents is the highest (1005 comments for 1322 posts) ominously showing that these TSS agent jobs are very much in active demand.

TSS category	# Posts	# Unique posts	# Post authors	# Comments
TSS agent hiring	1322	516	245	1005
Money launderers	6677	1435	541	3097
TSS webmasters	5167	1411	525	1908
Phone # providers	1129	258	107	190
Victim data sales	951	311	121	129

TABLE 1: Counts of posts for various categories of TSS operations based on Facebook data

TSS category	# Posts	# Unique posts	# Post authors
TSS agent hiring	161	104	57
Money launderers	8887	2138	787
TSS webmasters	9444	2826	815
Toll-free number providers	3104	511	226
Victim data sales	2757	519	204

TABLE 2: Counts of posts for various categories of TSS operations based on WhatsApp data

2.3. Impact of TSS Underground Economy

In this section, we focus on the TSS social media posts that are related to financial transactions and discuss their relation with possible TSS defensive measures.

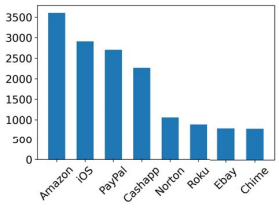


Figure 2: TSS post counts for different brands

Figure 3: A Facebook post selling TSS calls; prices are in INR

2.3.1. Selling TSS Calls. As discussed earlier, TSS webmasters make regular posts advertising TSS call sales to call centers as shown in Fig. 3. Fig. 2 shows the eight most frequently targeted brands[2] we found, while Table 7 shows the call prices. It is important to note the appearance of many new tech brands such as CashApp, Roku, Chime, etc none of these were seen in prior work [45]. Even airlines services such as Delta and United Airlines, were seen in our collected data. This suggests a fairly significant evolution of TSS scammers to target new brands and points to the need for *topic-agnostic* TSS mitigation tactics that have not been explored in the past.

2.3.2. TSS Operating costs and its effects. Table 7 shows that the call centers pay a hefty sum of between \$5 and \$10 USD per each call forwarded to them by TSS webmasters regardless of whether or not the scam is successful. In addition to TSS call sales, we also measured the market prices for other TSS ecosystem services. Using these prices, we were able to estimate the operating costs of a TSS call center. In addition, we assumed the median revenue of a successful TSS call to be \$250 (according to [40] in 2017). Using this cost and revenue model, we were able to perform a back-of-the-envelope calculation to estimate the target success rate for TSS call centers to be profitable. This yielded a surprisingly formidable value of 15%. Additional details for this can be found in Appendix B. Considering recent research on telephony scams which showed that about 1.17% of recipients of telephony scam calls get actually scammed by the calls [49], the success rates required for profitable TSS scams seem very high and indicate a high financial pressure on them to break even.

Meanwhile, the unbridled growth in the number and impact of TSS scams has also fostered the development of internet vigilantes known as *scam baiters* (e.g., [17]) whose goal is to verify and label newly discovered TSS websites and waste TSS agents' time by calling the advertised TSS number pretending to be a victims (e.g., using tools such as [28]), thus reducing the scammers' success rates and their morale. Although this may indeed discourage some scammers, the steep operating costs discussed above (and thereby, the required high target success rate) may also motivate other scammers to make up for the lost time by demanding more and more funds from true

2. Table 10 depicting several brands we found in the TSS sites we ultimately collected for this project is in the Appendix.

victims. For example, recent news reports describe several TSS agents attempting to steal thousands of dollars from victims [18], [48]. As an alternative to scam baiting, in this paper we propose a system that can automatically discover new TSS websites and block them at the web search engine level. Because TSS call centers purchase victim calls from TSS webmasters, and because TSS websites are the gateway to a large number of modern TSS scams, this can help significantly in countering TSS operations. In the remainder of the paper, we focus on studying this latter defense approach.

2.4. Key Takeaways for TSS Website detection

We conducted the first systematic study of TSS underground groups and their economy. This revealed, for the first time, a vibrant system of multiple independent operating groups such as call centers, web masters, money launderers, scam agents, and Toll-free number providers with each one specializing in different TSS operations. Some of these groups such as the money launderers, are likely involved in other cybercrime operations as well since their role is not inherently tied up to the core part of the TSS scams. Thus our findings can be vital for law enforcement officials in the future to being take-down operations on TSS as well as other scams. However, as discussed previously, for the rest of this paper, we will focus solely on applying the findings from this analysis to detect TSS websites. There are multiple ways in which our underground TSS study here has helped drive the design of our TSS detection methods. We discuss them below.

Blackhat promotion of TSS websites. Because TSS webmasters earn revenue on a per-call basis, namely by selling victim calls to TSS call centers, to maximize revenue they need to promote their TSS websites (and the TSS phone numbers within) as best as possible. Pertinent to this, we saw that the TSS community includes blackhat SEO (BHSEO) [38] experts who even advertise "educational" programs geared towards TSS webmasters in the underground groups. This shows that the webmasters often use BHSEO to abuse search engines and inflate TSS websites' search rankings. We can leverage this finding by looking for distinctive characteristics of BHSEO techniques in order to detect TSS websites. For instance, there might exist distinctive patterns in the nature of the *backlinks* used to inflate the rankings of TSS websites.

Explicit age-targeting. Our analysis revealed clear evidence of deliberate targeting of older-aged adults, with calls being advertised as all coming from potential victims in the age groups above a given threshold (e.g., 65+ years). Given such explicit targeting by the TSS community, it is possible that the TSS web masters are also taking these age groups into consideration when designing the websites. For example, they might be designing the websites with large font sizes in order to cater to the senior populations that often experience visual acuity issues. We will investigate this in later sections of the paper.

Effects of TSS compartmentalization. One of the most important findings of our underground study is the compartmentalization of TSS scams which revealed that the TSS web masters who develop the sites are separate from the call centers. The web masters obtain phone numbers and insert them into their sites (see Fig. 1). In an effort to cast a wider net for victims as well as conserve the costs of phone numbers, it is possible for the web masters to reuse the same phone number in multiple sites. We will also explore this idea in later sections when designing our TSS website data collection methodology. Furthermore, the phone number replacement services advertised by the toll-free number providers may translate in noticeable periodic changes in the phone number advertised by a given TSS website. This can also be used for developing detection methodologies.

We will corroborate all these insights in later sections, where we will present how they can be leveraged to derive three different groups of detection features that ultimately detect TSS websites in a way that is agnostic to the topic of the specific TSS scams.

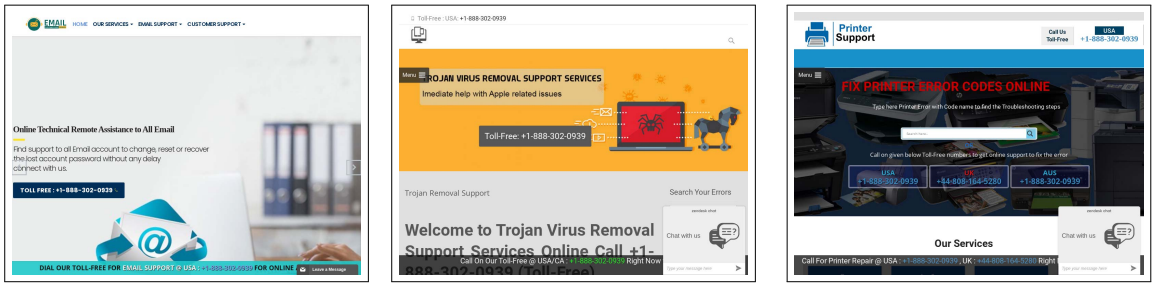
3. Collecting Tech Support Websites

As explained in Section 2, TSS websites function as a primary entry point to TSS campaigns. In this section, we focus on collecting *ground-truth data*. Specifically, we describe our approach for collecting both malicious (i.e., TSS) and benign tech support websites.

Ground-Truth Collection Challenges. Collecting and labeling TSS websites presents a number of challenges. First, modern TSS websites are often built to visually mimic legitimate technical support businesses (e.g., see examples of confirmed TSS sites in Figure 4), making ground-truth labeling difficult. Furthermore, TSS webmasters leverage a large variety of technical support topics to attract a broad audience of potential victims, thus requiring a *topic-agnostic* approach to discovering and labeling TSS websites. In addition, TSS websites are often promoted by using a combination of approaches, including BHSEO [38] and web search ads that allow them to appear in search results while blending-in with legitimate organic search results and search ads.

To build our TSS website collection and labeling system, we leverage lessons learned from our own investigation of the TSS ecosystem presented in Section 2 and develop a new and systematic way to label both TSS scams and legitimate tech support websites. In addition, we collect a large variety of metadata associated with these websites to enable a more in-depth analysis of how TSS webmasters are able to inflate their TSS websites' ranking to make sure their websites appear on (or near the) top of the list of web search results.

Avoiding search ads analysis. It is important to note that we intentionally avoid analyzing search ads. Crawling through ads to collect the content of the landing pages has important ethical implications, because automatically visiting search ad links may impact legitimate advertisers, since they will typically need to pay the search engine for every ad click. Because we don't know *a priori* which links are related to TSS vs. benign ads, "clicking" on legitimate ads would be unavoidable, and doing this at scale may impact many legitimate entities. In addition, search ads can be expensive (i.e., with a high cost per click), when they target specific search terms and/or populations. Therefore, in this paper, we avoid an analysis of search ads and leave a large scale detection/analysis of TSS-related search ads to a separate future work.



(a) contactemailssupport.com (b) applesupport247.com (c) epsonconnectservices.com

Figure 4: Example of three confirmed TSS pages. These TSS sites share the same scam phone number and cover three different tech support *topics* (email, malware/Apple devices, and printers).

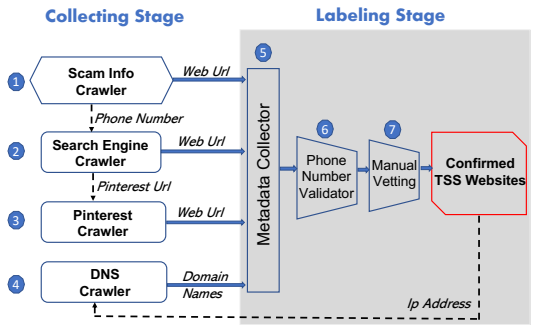


Figure 5: Topic-agnostic TSS website collection system.

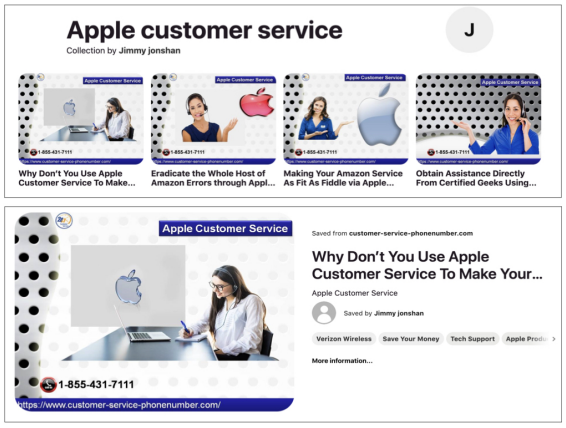


Figure 6: Example of pinned TSS images on Pinterest

3.1. Topic-Agnostic Data Collection

We begin by seeding our data collection system with phone numbers and URLs that have been previously reported as TSS-related in crowd-sourced scam alert websites (e.g., Scammer.info). To maximize the number of TSS websites and remove potential noise, we built a subsystem that consists of four crawler modules targeting different data sources, plus a TSS verification process, as shown in Figure 5 and described below. Notice that all modules below gather TSS data in a *topic-agnostic* way, meaning that they are independent from the specific TSS "topic" or products/brands abused in the scams.

❶ **Scam Info Crawler**. While studying the TSS ecosystem (see Section 2.3), we learned that there exists an online community of *scam baiters* who scout the web for scam websites, manually vet them, and report malicious ones to crowd-sourced databases. For instance, Scammer.info includes reports of a variety of online scams organized under a preset taxonomy. Reports regarding TSS are listed under a category appropriately labeled as *Tech Support Scam*. We therefore implemented a web crawler that automatically collects TSS posts and extracts TSS-related URLs and phone numbers.

❷ **Search Engine Crawler**. Additionally, we built a web search results crawler that takes TSS phone numbers discovered by our Scam Info Crawler and queries Google with each number as a search term. The related search results typically include: (i) TSS websites that advertise the phone number; (ii) websites that report phone numbers related to unwanted phone calls (e.g., 800notes.com); and (iii) popular websites abused by TSS scammers to advertise and support their operations (e.g., Pinterest.com, Wix.com, and Medium.com, etc.).

❸ **Pinterest Crawler**. Among the popular websites abused to promote TSS websites, we found that Pinterest.com has become a highly preferred venue for scammers, given the high number of TSS-related URLs it unwittingly hosts. Often, TSS scammers use Pinterest.com to create one or more boards related to a variety of technical support topics to which they pin images embedding a TSS phone number (see Figure 6), with an associated TSS URL link that helps to establish a high-reputation *backlink* to the scam website. To collect such data, we built a Pinterest Crawler that collects pinned images and extracts URLs and phone numbers from TSS-related posts.

❹ **DNS Crawler**. To expand the set of candidate TSS websites we leveraged the fact that different TSS websites often share the same hosting network infrastructure [45]. As shown in Figure 5, our DNS Crawler takes as input the IP addresses of confirmed TSS websites and queries a passive DNS datasets to collect domain names that share those same resolved IPs. The obtained domains will then be added to the list of sites to be vetted.

❺ **Metadata Collection**. The output of the four crawlers described above is a set of URLs that are likely related to TSS websites. However, the crawlers can obviously also collect URLs that are unrelated to TSS. To filter out such URLs, we use a manual vetting process (de-

scribed later in this section). To assist this vetting process, for each URL output by the crawlers we automatically gather additional information. For instance, given a URL we extract its web page content, the (effective) second-level domain name, and the content of up to 100 pages under this domain (screenshot, text, etc.).

❻ **Phone Number Validator**. Also, for each candidate TSS website we use OCR (with Tesseract [35]) to extract potential phone numbers, check whether the extracted strings are syntactically valid phone numbers, and filter out those websites that do not include a valid number.

❼ **Manual Vetting**. First, we collect the set of all TSS-related URLs, U_{SI}, and phone numbers, P_{SI}, found on Scammer.info. Then, starting from the websites collected via modules ❶-❻ that embed a phone number, to vet and label them we manually filter out those whose theme and content is unrelated to technical support services. For each remaining website, we extract its URL, u, and if $u \in U_{SI}$ we label the website as *TSS*. We refer to this set of TSS websites as TSS_U. Additionally, we extract the set of phone numbers, P_U, from the TSS_U websites, and analyze the remaining not-yet-labeled sites; if a website includes a number $p \in (P_{SI} \cup P_U)$ we label it as *TSS*.

Besides the above process, we also consider other heuristics for our manual ground-truth labeling. For instance, we noticed that, unlike legitimate technical support sites, TSS websites usually do not provide valid physical address on their web pages. Therefore, during our manual vetting we checked the candidate websites and attempted to verify the reported physical addresses by using Google Maps, and labeled as *TSS* those websites that either did not provide any physical address for their business or whose address was invalid, anomalous or non-existent.

As this step involved manual vetting, it was repeated twice by different authors independently to ensure label reliability. Between the two labellers, there were disagreements on only 13 websites which have been discussed and settled in a multi-hour meeting using further information such as online reviews, search engine results for the domain names in question.

Dataset. Using this approach, we were able to label 806 TSS websites, of which 202 sites can be attributed to the Scam Info Crawler (TSS_U), while the rest of the sites have been added with the help of the remaining TSS-data collection pipeline components described above. We refer to this dataset of 806 TSS web sites as `TSS-TA`.

3.2. Topic-Based Data Collection

To further expand on the collection of TSS websites, we also built a ground-truth collection subsystem that can be seeded with the keywords extracted from the TSS sites collected in a *topic-agnostic* way (see Section 3.1). This additional *topic-based* data collection module, summarized in Figure 7, is partially inspired by previous work [45]. However, because [45]'s code is not openly available (we confirmed this with [45]'s main author), we built our own system and took the opportunity to introduce several important technical improvements, including introducing *topic modeling*, *geographical diversity*, and *age diversity*, as detailed below.

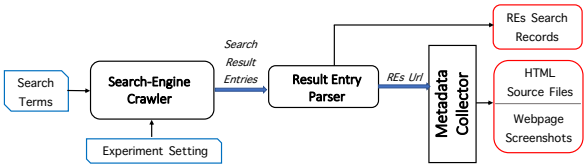


Figure 7: Search Crawling System Architecture

3.2.1. TSS-Related Search Terms and Crawling. TSS webmasters rely heavily on BHSEO techniques to inflate the ranking of their websites. This include tuning the text towards specific keywords that can be picked up by search engines. To automatically learn important keywords among the ground-truth TSS websites previously discovered by our topic-agnostic TSS collection system (Section 3.1), we use an LDA-based topic modeling algorithm [14]. We heuristically set the maximum number of topics for LDA to 100, and select the first four most important keywords from each discovered topic. Then we use these keywords to feed an automated web search crawler we built and collect the top 50 links for each search query results.

Through manual analysis of the topics discovered via topic modeling, we were easily able to recognize and extract 44 different brand/product names that are being actively targeted by TSS scammers, including *HP printer*, *Canon printer*, *Apple*, *Quick Books*, *Cash App*, etc., as well as IoT products such as *Amazon Firestick* and *Roku* devices. One important observation is that these 44 TSS-targeted brands represents a significantly broader set of scams, compared to the only 7 brands identified by [45] in 2018 (all of which were also re-discovered in our work). This clearly shows that TSS scams have continued to grow unabated in the past few years by expanding their targets.

Increasing Search Results Diversity. Because search results may vary depending on users' features, such as age and location, we programmed our web search crawler to include the following:

- *User Age Diversity*: We created two Google accounts with different user profiles: a 30-year old user and a 70-year old user. Then, our search crawler issues search queries using three different user profiles: anonymous (no user registration/login), a 30-year old, and 70-year old user.
- *Geographical Diversity*: We purchased a commercial VPN/Proxy service to programmatically change the crawler's source IP address to different geographical locations, including different US states (New York, Texas and California) and english-speaking countries (UK, Canada, Australia, New Zealand, and India).

Dataset Using the above setup, our crawler initiated queried Google.com using 276 different search queries from 8 different geographical locations. We then used our Scammer.info crawler (❷) and the filtering and manual vetting (❻ and ❼) described in Section 3.1 to label TSS websites among the search results. Overall, we collected web pages from 6,486 distinct effective second-level domains and were able to confirm that 456 were TSS scams. Collectively, these sites covered tech support topics related to all of the 44 brands we previously extracted. We refer to this dataset as `TSS-TB`.

3.3. Collecting Benign Tech Support Websites

To contrast TSS website features to the features of legitimate operations, we also collected information from *benign technical support* websites. These sites include *official* technical support web pages for the 44 brands we found to be targeted by TSS, as well as generic technical support services offered by *legitimate third-party* businesses. We refer to the dataset of benign tech support websites as `BTS` and divide the into two subsets: the set of official support sites, `BTS-O`, and the set of legitimate third-party support sites, `BTS-T`.

While collecting `BTS-O` examples is relatively straightforward, collecting and labeling `BTS-T` web pages is more challenging. To address this latter issue, we relied on the Better Business Bureau (BBB) [9]), a nonprofit organization that focuses on advancing marketplace trust by rating and accrediting local businesses. Specifically, we searched for *technical support* businesses (as labeled by BBB) in all 50 US states, and selected all listed businesses with an *A+* rating. We then labeled the related websites (linked on the BBB search results) as `BTS-T`. With this approach, we were able to collect data from 2438 legitimate tech support websites (38 `BTS-O` and 2400 `BTS-T`) that we can use as *negative* ground truth.

4. Analysis of Ground-Truth TSS Websites

By conducting a pilot experiment, we initially analyzed the web search ranking (using Google) of ground-truth TSS websites we collected using our search-engine crawler as described in Section 3.2.1. We found that 25% of the TSS sites appear in the first page of search results, and that TSS websites that abuse popular brands are often ranked higher than even the official support websites for those brands (see Figure 10). This corroborates previous observations [45] and confirms that TSS websites are still able to successfully and significantly inflate their ranking in the search results, despite attempts by web search engines to mitigate this issue.

To understand what drives this, we conducted an in-depth analysis of the methods used by TSS websites to promote their scams and inflate their rankings. Our goal is to identify a set of fundamental features used by these successful TSS websites that we can later leverage to build an accurate and reliable TSS website detection system. To perform this analysis, we used the ground truth TSS websites in our `TSS-CA` dataset (see Section 3.1). Our analysis is divided into three categories each of which is directly inspired by the takeaways we cited in the TSS ecosystem study (see Section 2.4).

4.1. Backlinks

Due to the nature of Google's page-rank algorithm [15], backlinks (i.e., hyper-links that point to a given website) are a predominant factor in search engine optimization (SEO) techniques. Therefore, to understand how TSS websites abuse backlinks, we measured the difference in the number and type of backlinks between TSS and benign tech support websites in our ground-truth datasets. To discover backlinks towards a given website, we made use of the Moz platform (moz.com). Along with

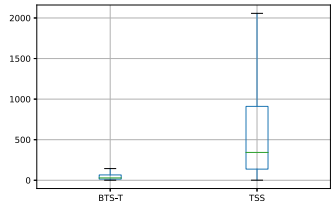


Figure 8: Backlink domains count

the each backlink URL towards a target site, we also collected metadata such as the anchor text associated with the link, the type of backlink (e.g., `nofollow` links [29]), the reputation of the domain associated with the page containing the link (on a scale of 0 to 100 assigned by Moz) and the number of other domains that page links to. For practical reasons, we considered only the top 5,000 most reputable (as per Moz's reputation score ranking) distinct backlink domains.

Backlink counts. We first analyzed the number of backlinks for each ground-truth website in our `TSS-CA`, `BTS-T` and `BTS-O` datasets. Unsurprisingly, the vast majority of `BTS-O` websites, being the official brand technical support sites, tend to have a large number of backlinks (>5,000). On the other hand, `BTS-T` websites (i.e., legitimate third-party tech support sites), tend to have few backlinks. On the contrary, TSS websites tend to have a relatively high number of backlinks, though less than `BTS-O` sites. The distributions of of backlink counts are shown in Figure 8 for the `TSS-CA` and `BTS-T` websites.

Spam-infested backlink pages. Upon manual investigation, we noticed that many of the backlinks pointing to TSS websites reside in the (typically unmoderated) comments section of otherwise legitimate web pages that have been abused and are polluted with a significant number of links pointing to third-party origins. On the contrary, this is rarely the case for links pointing to benign tech support sites.

High-reputation backlink pages. `TSS` websites tend to abuse high-reputation web pages that do not properly set the `nofollow` attribute [29] for links embedded in user-provided content. For instance, websites such as microsoft.com, ibm.com, hp.com, gnu.org, etc., include forums or community sections in which users can create profiles and post comments. We found that in some cases user profiles can include a URL pointing for instance to a personal page. It turns out that the related links do not include the `rel="nofollow"` attribute, and thus tend to be abused by TSS webmasters who can create several fake profiles and include backlinks to their TSS websites from the profile URL section. Similarly, other cases simply rely on popular online forum services failing to properly sanitize third-party content or to automatically include `nofollow` for every link in the community discussions [29].

4.2. Websites Design Features

As mentioned earlier, TSS websites are often designed to have the look-and-feel of legitimate websites. This

makes them difficult to distinguish them from benign tech support websites, in particular if we restrict ourselves to analyzing their textual content, as done in previous work [45] (see also results in Section 6).

However, by analyzing samples taken from our initial ground-truth datasets, we observed a key differences in the design of TSS vs. BTS-T and BTS-O websites. Specifically, TSS websites advertise their (scam) support number much more "aggressively" than legitimate sites. This makes sense, since selling phone calls to TSS call centers is how TSS webmasters generate revenue (see Section 2.3.1). To quantify this design differences, we used visual analysis based on OCR to identify valid phone numbers within web page screenshots taken from both TSS and benign tech support sites. In summary, we found that about 99% of TSS sites in our dataset prominently display a phone number on their home page. Also, we found that most pages under TSS websites repeatedly advertise the TSS phone number, often multiple times per visited page. On the contrary, 83.9% of BTS-O pages did not display a phone number. Manual analysis showed that the reason for this is that many brands (e.g., Yahoo) do not offer phone-based technical support for their free customers (they only offer it as a premium service). In cases like Amazon, even paid customers have to first walk through a number of dedicated self-help tech support pages and a virtual chat assistant, before being able to call a support phone number. On the other hand, BTS-T sites are understandably somewhat similar to TSS websites, given that their key goal is to also drive (legitimate) revenue by increasing the number of support calls. Nevertheless, it is interesting to see from the table that BTS-T sites are not as aggressive as TSS in the way the phone numbers are reported in pages under a given website. Table 8 (in Appendix) presents a more detailed analysis to show the number of times the same phone number appears in different pages crawled from sites belonging to either TSS or benign categories.

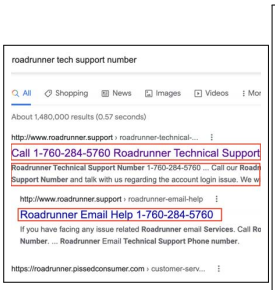
Figure 9: Example TSS website

Besides number of occurrences, we also measured the prominence of the displayed phone numbers, by measuring the area of the bounding box around the largest phone number appearance. Figure 9 shows an example, whereas Table 3 shows the phone number area sizes (in pixels) at different area size percentiles for the three datasets. These results clearly show that TSS websites stand out, with phone number areas as large as twice the others.

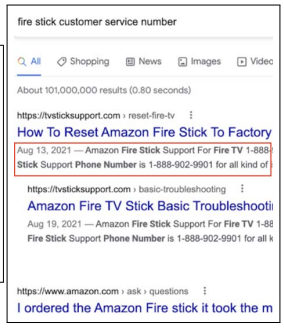
Another interesting thing we observed is that TSS webmasters tend to include scam phone numbers within the page title and search results snippets (e.g., using the meta tag), so that the phone number immediately appears

Percentile	25%	50%	75%	90%	98%
TSS	1,807	2,856	4,681	9,328	18,308
BTS-T	943	1,380	2,580	4,064	9,650
BTS-O	1,100	1,100	1,430	2,146	4,224

TABLE 3: Distribution of phone number page area size (in pixels)



(a) Roadrunner E-mail



(b) Amazon Firestick

Figure 10: TSS phone numbers in title and search snippets. Notice how the TSS sites appear above the BTS-O sites (such as www.amazon.com) in both cases.

in search results alongside the link to the TSS website, as shown in Figure 10. Our measurements show that more than 68% of TSS sites make use of these "tricks" to promote their scam phone numbers, as opposed to less than 5% of BTS-T sites and none of the BTS-O sites.

4.3. Website Record Features

We found that the age of TSS domain names tends to be significantly "younger" than that of legitimate sites, though still significantly "older" than other types of malicious domains (e.g., active phishing and malware domains typically have a recent registration date). Using *Whois* lookups for domain registrations, we found that 80% of TSS websites are about 2-5 years old (in terms of registration date), while only 6.7% of BTS-T websites are this young.

We also observed that many TSS websites undergo stages of evolution, whereby at the beginning the website may not include any phone number. During this initial phase, the developer typically aims to build a network of *backlinks* to help popularize the website and inflate its search ranking. Then, after some time (often, a few months), as the website becomes highly ranked in search engine results and the developer is able to sell her services to a call center, the website is weaponized by adding a TSS phone number that will be used to forward prospective victims' calls to a TSS call center.

At the same time, it should be noted that TSS websites operate with impunity for several years, before being taken down or blocked (or falling out of business for other reasons). For instance, the Google Safe Browsing (GSB) blocklist only included a single domain out of the 806 ground-truth TSS websites in our TSS-TA dataset. Interestingly, that one domain that was blocked by GSB with label as "Social Engineering," thus showing that Google is in fact interested in labeling TSS scams, although

GSB currently seems to have very minimal coverage. This further indicates that existing mitigations are mostly ineffective and points to the need for new defenses, which we propose in Section 5.

Perhaps even more interestingly, we also found that TSS websites periodically change the advertised phone number, presumably if/when a toll-free number is blocked by the owner (i.e., a telephone company) due to abuse complaints. Specifically, we found that 45% of TSS sites changed phone numbers during their lifetime, while only 1.5% of BTS−T sites did. Also, 33.4% of the TSS sites we analyzed changed their phone numbers at least twice during their lifetime.

Another characteristic we observed is that TSS webmasters may advertise the same phone number (used to sell victim calls to TSS call centers) on multiple TSS websites they control, which are often related to different TSS scam topics/brands. To measure this, we counted the number of sites that share a phone number with one or more other sites, both for TSS and benign tech support websites. Overall, while more than 65% of the TSS sites share a phone number with at least one other TSS site, only 0.58% of BTS−T sites did so. Also, we found several website clusters containing 6 or more TSS sites (with different domains and scam topics) all sharing the same number. An example of three TSS websites with different topics sharing the same scam phone number is shown in Fig. 4. For TSS websites, sharing the same (usually toll-free) number makes sense, since operating a phone number has a non-negligible cost and TSS webmasters can diversify their scam portfolio and attract more potential victims with a single number. As for different BTS−T sites sharing the same phone number, we only found 7 groups of 2 legitimate domains (i.e., 14 domains out of 2400) that shared one number. For instance, we found cases in which two domains belonging to the same legitimate technical support company with operations into two different locations hosted two slightly different versions of the company website (advertising services at the two different locations), which used the same contact number.

Finally, we also observed that most TSS websites (81.6%) tend to advertise toll-free phone numbers whereas only few (9.5%) legitimate sites (BTS−T) use such numbers. Rather, legitimate third-party support sites tend to use local numbers that match the area code of the region they are serving. We presume that this is motivated by the higher cost of maintaining a toll-free number vs. local numbers. On the other hand, for TSS sites having a toll-free number lends an air of legitimacy to the website which is imperative when impersonating popular tech brands and could thus be a big factor in enticing future victims.

5. Mitigating TSS in Search Results

In Section 4, we have analyzed the main characteristics of TSS websites. In particular, we have focused on key topic-agnostic features that play a critical role in inflating the search rankings and advertising the TSS phone number used to monetize the scam, besides other characteristics that are typical of TSS websites but rare in legitimate technical support sites. We now describe how those insights can be translated into statistical features to be used in the context of a supervised machine learning classifier, and then present a new detection system that is able to detect TSS websites among organic web search results.

5.1. TSS Detection Features

Backlink Features. Given a web page WP, we collect backlink data using Moz[10], as explained in Section 4.1. Specifically, we collect information from each backlink web page BP_i that links to WP. We refer to the domain name under which page BP_i is hosted as BD_i. Similarly, we refer to a TSS domain under which WP is hosted as WD. From the Moz data, we measure several statistical features, which we list below:

- *Number of backlinks*: Number $BL(WP)$ of distinct backlink pages (i.e., number of 'BP_i's) and number of related distinct domain names (i.e., number of 'BD_i's) pointing to WP. *Intuition*: TSS webpages inflate their ranking by abusing other pages on multiple sites to create numberous backlinks.
- *Link distribution of backlink pages*: For each backlink page, BP_i, pointing to WP, we count the number $OL(BP_i)$ of outgoing links (i.e., URLs pointing to a domain different from BD_i). Then, we compute the median and 95th percentile of the set of values $\{OL(BP_i)\}_{i=1,\ldots,BL(WP)}$, and use them as additional features. We also compute a similar feature for the number $OD(BP_i)$ of distinct domains extracted from all outgoing links found across the WP's backlink pages. *Intuition*: TSS webmasters typically abuse unmoderated third-party web pages that allow users to create content containing many outgoing links.
- *"Follow" vs. "nofollow" links*: Absolute number of `nofollow` backlinks and ratio of `nofollow` vs. all backlinks towards WP. *Intuition*: TSS webmasters try to find third-party pages that fail to automatically include the `nofollow` attribute to web links embedded in user-created content.

Feature robustness discussion: Notice that TSS websites need to be pointed to by a large set of "artificial" backlinks, as part of a BHSEO strategy towards inflating search result rankings. Thus, while it may be possible for TSS webmasters to attempt to manipulate the backlink features, this would ultimately affect their ability to inflate the ranking of their websites.

Phone Number Prominence Features. As discussed in Section 4.2, TSS websites tend to advertise their phone numbers with a much higher prominence than benign websites. To capture this characteristic, we extract phone numbers from a web page as described in Section 4.2. Given a web search result link L, we use our web crawler (Section 3) to visit up to 100 different pages under L's website. Let $WP_i(L)$ represent a web page in this set of crawled pages. For each page, we measure these features:

- *Phone number occurrences*: We measure the number of times a phone number appears in the HTML of the visited pages. In addition, using OCR we separately measure the number of visual occurrences of phone numbers within the browser view port and the full rendered page. We also measure these occurrences in the `title` and `head` HTML tags, since these are often

abused by the scammers to display phone numbers to potential victims even before they click on a search result link (see Figure 10). *Intuition*: Because TSS phone numbers are critical for monetizing scams, TSS websites tend to advertise them on multiple pages and more than once per page.

- *Area- and location-weighted number prominence*: We aim to capture the visual prominence of phone numbers by computing the fraction of the area of a web page viewport that is covered by the bounding box drawn around phone numbers found on that page. Furthermore, we use a set of heuristics to calculate whether the page displays a phone number towards the middle of the visible portion of the page. To this end, we extract the location of the phone number bounding box, and the vertical middle point VB of the box. Then, we compute the distance of VB for the top, VB_t, and bottom, VB_b of the page. Finally, we compute the location prominence feature as $\min(VB_t, VB_b)/H$, where H is the height of the page viewport in pixels (our crawler uses an instrumented browser with a common desktop screen resolution). *Intuition*: To more easily attract a potential victim's attention, TSS phone numbers are typically visually large and often placed towards the middle of the web page.

These features are measured for each page $WP_i(L)$, and then averaged across all pages we crawled on a site.

Feature robustness discussion: The phone number advertised on TSS websites is key to monetizing the scams. This is the reason that drives TSS webmasters to promote their phone numbers so aggressively. While it is possible for scammers to tinker with the way phone numbers are displayed, this has the risk to diminish their ability to successfully lure more victims, and in particular elderly people (their primary victims), to placing a call.

Website Record Features. To capture the observations from Section 4.3 on how TSS websites change in time, we measure the following features.

- *Website registration*: Using *Whois*, we extract the date when the domain name of a page WP was registered. *Intuition*: TSS websites tend to be relatively "younger" compared legitimate technical support sites.
- *Phone number changes*: Using archive.org, we check how many times the phone number advertised by a page WP changed, in time. Furthermore, we compute the maximum number of days between two phone number changes throughout the history of the website. *Intuition*: In Section 4.3, we discussed how many TSS websites undergoes changes in time. For instance, the advertised TSS phone number may change when a previous number is blocked by the owner telephone company due to abuse complaints. It is to be noted here that completely eradicating a TSS site domain as soon as its first associated phone number is blocklisted is not in the site operator's best interest as popularizing a TSS site on search engines in a labor and time-intensive effort that requires employing elaborate Blackhat SEO techniques.

Feature robustness discussion: As discussed previously, TSS scammers may get their phone numbers blocked due to eventual abuse complaints from victims who realized the scam as well as scam baiters. These complaints may thus force a churn in the phone numbers being advertised in the TSS websites. Similarly, although TSS websites currently operate for fairly long periods of time (up to a few years), URL reputation and blocklist services may eventually catch up, driving a (slow) churn in the domains used to host the scam campaigns.

5.2. TASR System Overview

To detect TSS websites among web search results, we propose TASR (Topic-Agnostic Scam Recognizer), a *topic-agnostic* TSS detection system. Our system aims to be used by a search engine (e.g., Google) or as a browser extension to detect and filter out (or at the very least de-rank) search result links that are labeled as *TSS* with high confidence.

Given a list of organic search results $S = \{s_1, \ldots, s_n\}$ (i.e., non-ad web search result links), TASR computes the features described above (Section 5.1) for each link s_i. More precisely, given a web page w_i pointed to by link s_i, TASR translates w_i into a feature vector and uses a RandomForest classifier to assign a label, either *TSS* or *benign* (i.e., *not TSS*), to w_i, and thus in turn to link s_i. Then, search links labeled as TSS can be demoted or pruned from the search results.

6. Evaluation

In this section, we evaluate TASR's ability to detect TSS websites among organic web search results and compare its performance to previous work.

6.1. Experimental Setup

Datasets: Using our data collection systems and ground truth labeling, we gathered the following main datasets (see Section 3), which we use for evaluating TASR:

- `TSS-TA`: 806 ground-truth TSS websites collected with the topic-agnostic system described in Section 3.1. In all the following experiments, we use (part of) this dataset for training.
- `BTS`: 2400 legitimate technical support websites collected as described in Section 3.3. We randomly split this datasets into 1,000 samples used for training and the remaining 1400 for testing.
- `ALL-TB`, `TSS-TB`: During our topic-based data collection described in Section 3.2, we collected 6,486 distinct second-level domains among which we could identify 2,833 domains with a valid phone number. We refer to these 2,833 domains as `ALL-TB`. Of these, we were able to confirm 456 ground-truth TSS websites, which we refer to as `TSS-TB`. These two datasets are exclusively for testing purposes (i.e., we never use them for training) as they represent the types of websites encountered by real users through their web searches.

Experiments: We evaluate TASR in multiple settings:

Cross-validation: In this experiment, we rely on the `TSS-TA` dataset and the `BTS` dataset. We sample 500 TSS sites from `TSS-TA` and 1,000 sites from `BTS` and use these for performing 10-fold cross-validation.

Train-test: We also perform train-test experiments by using the above 500 TSS and 1000 benign sites for training.

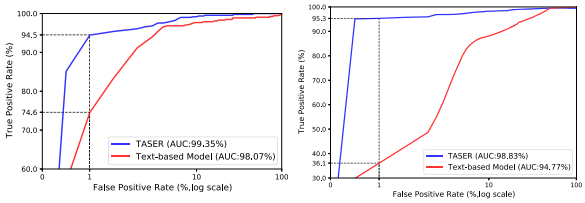
Then, we use the remaining 306 sites from TSS-TA as the positive test set, and the remaining 1400 benign sites from BTS as the negative test set.

Topic generalization: To test TASR's ability to adapt to newer topics, we first manually analyzed the above 500 TSS sites and attributed each of them with one or more of 21 different scam topics (see Table 10 for the full list). For each TSS topic, we conducted a *leave-one-topic-out* test as follows. Assume that there are N_T TSS sites that belong to a particular topic T. We exclude these N_T sites from the 500 TSS sites to create a *positive* test dataset. The remaining $(500 - N_T)$ form the *positive* training dataset. Then, starting from 1000 benign websites drawn from the BTS dataset (the same 1000 sites used in the *train-test* experiment), we randomly select $2 \cdot N_T$ sites and set them aside as the *negative* test dataset, whereas the remaining $(1000 - 2 \cdot N_T)$ benign site form the *negative* training dataset. For each scam topic, we train on the combined *positive* and *negative* training datasets and test on the combined *positive* and *negative* test sets. We repeat this experiment three times for each topic and average the results.

Deployment: In addition, we evaluate TASR's ability to work in a realistic deployment scenario. For this, we used all samples in TSS-TA and BTS datasets for training. We then deploy the trained classifier over the search results obtained by our topic-based search-engine crawler (see Section 3.2). As mentioned earlier, these search results are included in the ALL-TB, therefore we use the ALL-TB as test dataset. As part of this experiment, we measure the coverage of TASR with respect to the ground truth dataset TSS-TB and also evaluate possible false positives by performing manual analysis of sites flagged as being TSS-related by our classifier.

Comparison with previous work: For each of the experiments described above, we also compare directly with the topic-based detection system proposed in [45], which we refer to as *text-based model*, since its features are mostly based on the text contained in web pages. As the source code for this classifier is unavailable (we contacted the authors directly to ask for the code), we built a Naive Bayes (NB) classifier as described in [45]. Note that the model proposed in [45] works only on a single web page as opposed to an entire web site as in TASR. While many TSS websites in our dataset do have significant textual content on the home page, there were a few sites which only have links to interior pages in the home page. Thus, in order to allow the text-based model proposed in [45] to observe more input data (and achieve better performance), we select text content from three random pages within the same website and append it to the home page's text content, before feeding it to the text-based model. For each experiment, to enable a one-to-one comparison we train/test the model using the same exact set of ground-truth datasets used to train/test our TASR system.

Feature analysis: Given the three feature groups discussed in Section 5.1 (backlinks, phone numbers, and website hitory features), which are used by TASR, we evaluate the relative contribution of each feature group to TASR's performance. For this, we conducted cross-validation experiments with TSS-CA data in two modes: (1) a *leave-one-fg-out* mode (where *fg* stands for feature group), in which we leave one feature group at a time out of train and test sets; and (2) a *keep-one-fg-in* mode in which we only train and test using one feature group each time.



(a) Cross-Validation RoC (b) Topic generalization RoC

Figure 11: RoC curves for TASR and text-based classifier

6.2. Results

We now present the results obtained for each of the experiments outlined in Section 6.1.

Cross-validation: The cross-validation results are summarized by the ROC curve in Fig. 11a. We can see a high performance of TASR, with an AUC of 0.9935. Setting TASR's detection threshold so as to have less than 1% false positives yields a true-positive rate of 94.5%. In comparison, the text-based model we reproduced from [45] had a much lower performance, with a true positive rate of 74.6% at the same false positive rate of 1%.

Train-test: In this setting the classifier is trained on 500 TSS samples and 1000 benign samples, as explained in Section 6.1. To compute the true and false positives, we fixed the detection threshold to the value we previously found in the *cross-validation* experiment, which aims to tune TASR to generate no more than 1% false positives (according to the *cross-validation* ROC analysis). In this setting, TASR was able to detect 92.5% of TSS sites at a false positive rate of only 0.4%. Similarly, we repeated the same experiment (with the same training and test samples) with the text-based model and tuned the model as before to generate no more than 1% of false positives. In this setting, the text-based model was only able to detect 82.0% of TSS samples at a false positive rate of 1.1%.

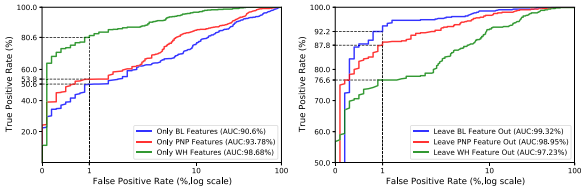
Topic generalization The performance difference compared to previous work becomes even more evident in the *topic generalization* results summarized in the ROC curve in Figure 11b. While in this setting TASR is able to achieve a detection rate of 95.3% at 1% false positives, the text-based model saw a steep fall in performance with a detection rate of only 36.1%. This shows that the text-based model struggles to generalize to new topics, which can be described by significantly different text, compared to TSS topics seen during training. On the other hand, TASR is designed to be topic-agnostic and to leverage the inherent features that capture how TSS scams are promoted on the Web and search engine results, rather than focus on the specific content and scam topics embedded in the TSS websites.

Deployment: When testing TASR on all web search results (ALL-TB dataset), as described in Section 6.1, our system flagged 717 websites out of 2,883 as TSS. Upon analyzing these results, we observed that 434 websites

from the search results that were labeled as TSS were already included in our TSS-TB ground truth dataset. Namely, in this setting TASR detected 95% (434/456) of all TSS websites included in TSS-TB, indicating a high "coverage" on a never-before-seen TSS dataset. To investigate the remaining 283 search results labeled as TSS by TASR, for which we did not have previous ground truth, we performed manual analysis following the same methodology described in Section 3. While it is difficult to establish a definitive label on these sites, due to lack of hard ground truth evidence, we leveraged a set of indicators to establish a *likely* ground truth label. First, we verified that these 283 sites have no intersection with the benign sites in the BTS datasets. Next, we measured the intersection of these 283 sites with top sites ranked by Tranco [43]. We found only 1 site, heimdalsecurity.com, in the top 100,000 Tranco sites and 25 sites in the top 100k to 1 million sites.

By manually inspecting the 26 sites in the top 1M raking, we found that 20 of them are highly suspicious because they include several visual features common to TSS websites and also included fake or non-existent business mailing/street addresses, a feature that we found to be often used in TSS sites. However, the remaining 6 were likely benign tech-related sites, which thus we consider as false positives. To get a better idea of the false positives generated by TASR, we also took a random sample of 50 websites among the "unpopular" 258 sites labeled as TSS and conducted a similar manual inspection. Using our domain knowledge, we observed that 48 of these 50 sites appeared to be highly suspicious and likely TSS sites, while 2 other sites were technology-related sites that seemed to be false positives. Overall, we manually investigated 76 of the previously unlabeled 283 sites that have been labeled as TSS by TASR and found 68 to be (likely) TSS sites whereas 8 were (likely) not.

Next, we also evaluated the extent of potential false negatives. For this, we chose a random sample of 50 sites that were labeled as benign (non-TSS) by TASR and manually inspected them as above. We concluded that 49 of these websites are likely benign while only 1 site (123hpcom.tech) appears to be a TSS site targeting HP printers. Manual inspection revealed that this site is not "weaponized" with a TSS phone number and rather seemed to rely on a live chat-based mechanism to lure in victims. In fact, we interacted via chat with one of the website's agents, and we asked whether they could give us a phone number to call for support. However, we were told that they first needed to learn about our technical issues via the chat system, and did not provide us with a phone number. Because most TSS operations rely on phone numbers to monetize the scams through a call center, some chat-based scams may not be easily detected. Also, it should be noted that chat-based support systems are less likely to attract older victims, who may be more comfortable with speaking to a technician over the phone than via chat. Furthermore, high-pressure sales tactics and social engineering over the phone are more effective then via chat, since phone communications are more direct and can more easily play on one's emotions, whereas chat windows can be easily closed with a mouse click. This likely explains why during this study we did not see a significant number of such TSS being promoted.



(a) *Keep-one-fg-in* mode (b) *Leave-one-fg-out* mode

Figure 12: RoC curves showing the impact of the 3 main feature groups used by TASR

Feature analysis: The results for the feature analysis experiments are shown in Figure 12. The *keep-one-fg-in* experiment results show that all three feature groups make an important contribution towards detection, with an AUC $>$ 0.9 for each of the three feature groups. At the same time, the *leave-one-fg-out* graph shows that the TASR is not dependent on a single critical feature group, although web site record features do appear to be the most important group overall.

7. Discussion

Armed with the knowledge of TASR's architecture, a TSS webmaster may attempt to alter her TSS websites so that they diverge from the characteristics of TSS sites leveraged by our system, to evade detection. However, by doing so the TSS webmaster risks decreasing the efficacy as well as the reach of her scams. For instance, as described in Section 4, TSS sites are often able to inflate their rankings to appear prominently in search results, even ahead of official brand websites (e.g., HP's official website for printers), thus attracting a large number of potential victims. If the scammers stop or reduce the use of BHSEO techniques such as "artificial" backlinks, this may cause a significant decline in the search rankings and thus decrease their profits. Similarly, TSS websites may be redesigned to display phone numbers less prominently. However, such attempts may hamper access to the phone number, which may drive a decrease in phone calls especially among older adults (their main target population) due to visual acuity issues [13]. Also, the attackers can attempt to avoid historical phone number churn by simply spinning up a new website with a new phone number each time a TSS number gets blocked. However, this would force the TSS webmaster to rebuild the backlink network, and make the website more easily detectable using features such as the age of the site's domain.

Another issue to consider is that some of the features (e.g., the *Website Record* features) used by TASR dependent on third-party resources, which might not have the ideal required visibility. For example, to compute the phone number churn features, we relied on data from archive.org which might have less than desirable granularity with respect to the time-intervals of crawling. However, we envision that TASR could be deployed by web search engines (e.g., Google, Bing, etc.) whose core functionality depends on frequent crawling of candidate sites, thus resulting in much more fine-grained snapshots. This may potentially lead to even better performance than what we reported in our evaluation.

Abuse Disclosure. Our work revealed systematic abuse of multiple web platforms by TSS scammers. Importantly, we found large scale abuse of social media platforms like Facebook and WhatsApp (Section 2), which we have disclosed to Meta. In addition, we have found continuing abuse of Google's web search engine, which we reported to Google along with a list of targeted search terms. This was accepted as a valid bug report.

When analyzing TSS-related backlinks, we found that besides abusing reputable third-party sites that host user-generated content (see Section 4.1), scammers also frequently abuse blog hosting platforms such as `wordpress.com` and `blogspot.com` to build dedicated blogs that link back to their TSS sites. While this approach may not have a large impact in inflating search rankings [51], we still reported this wide abuse to the platform owners. We also disclosed our findings about systematic abuse to Pinterest (see Section 3), which has become an unwitting participant in promoting TSS.

We also disclosed the information we found on WhatsApp and Facebook to law enforcement agencies in India, as they may be useful for initiating call center take-down operations [19], [21]. Beside call centers, law enforcement agencies could also use other TSS ecosystem components we tracked, such as social media posts from money launderers, to dismantle scam operations by targeting the underground economy components, as suggested in prior works in other domains [36].

Ethical considerations. Our work involved automated crawling of web search results and candidate TSS websites. Web crawling has been used extensively in web security research (e.g., in [40], [45], [38], just to cite a few related works), and thus we believe that our data collection approach does not raise any significant ethical questions. As discussed in Section 3, we intentionally avoid an analysis of Google's search ads, since this would require ad clicks that could cause financial repercussions to benign advertisers and may thus present ethical issues. Further we note that we did not interact with the TSS sites or forums, but only passively collected data from them. In two ad-hoc instances (Section 6.2 and Appendix B), we interacted with TSS scammers to corroborate some of our findings. Our IRB has reviewed the process used for these interactions and granted approval.

8. Related Work

Prior work on TSS has focused primarily on exposing the mechanics of the scams by documenting the interactions that scammers have with victims [33], [34], [40], [44]. Furthermore, these works have focused mainly on *aggressive* TSS, which are early versions of TSS sites that were mainly distributed via malicious ad networks with lax content policies [50]. On the other hand, we focused on studying the TSS ecosystem and on developing a defense against modern and insidious variety of TSS sites that instead use a number of technology-related topics as bait and abuse search engines to lure victims.

Thomas et al. [46] surveyed various underground economies supporting multiple criminal business models such as spam services and malware-driven scareware. However, the underground economies behind telephony scams have not been covered in such prior studies. At same time, the telephony channel is critical to TSS scams we study. In this paper, we exposed supporting components such as toll-free number providers and web masters that sell calls that play a pivotal role in the survival of these scams. Also, our findings *reinforce* the main idea proposed in [46] that identifying the economic components of a cybercrime often leads to identification of brittle dependencies which can be leveraged for defense. For instance, following our approach, future law enforcement official can pursue more studies to curb activities of TSS money launderers that are predominantly monetizing gift cards thus stifling the entire crime.

There have also been some recent works that focus on combating telephony scams via different techniques that work directly at the telephony channel level. Phone number blocklists have been shown to be effective in combating scams and can be populated by telephony honeypots [32] as well as social media platform-based crawlers [31]. However, blocklist-based solutions primarily focus on blocking incoming calls from known spam numbers and suffer from coverage issues. To improve phone blocklists, researchers have begun to work towards client-based telephony scam solutions [37], [12]. TASR is complementary to these defense solutions as it can be deployed at the web search engine level to prevent the exposure of scam numbers to potential victims.

Specific blackhat SEO techniques such as malicious redirections [38], stealthy defacements [53] and wildcard DNS entries [25] have been studied. Our TSS site analysis revelaed that operators often use a more labor-intensive process by crafting several "reputation-leaking" backlinks. It is also important to note that black hat SEO identifying features form only one of the three feature groups that we propose for detection of modern TSS sites.

The closest related work to ours is by Srinivasan et al. [45], in which the authors developed a text-based TSS classifier as a tool to measure the prevalence of aggressive as well as search-based modern TSS scams. With our TSS ecosystem study, which to our knowledge has not been done before, we show that this modern variety of search-based TSS scams are proliferating with an increasing range of targeted topics, making such text-based solutions ineffective. Therefore, we developed TASR as a new *topic-agnostic* defense against TSS that could be deployed by web search engines. We also performed a thorough evaluation in which we compared TASR directly with the classifier proposed in [45] via multiple experiments.

9. Conclusion

In this paper, we studied how to detect links to malicious TSS websites that appear in web search results. We first studied the TSS ecosystem, with particular focus on how modern TSS campaigns are operated and promoted on the web. Then, we capitalized on our findings by proposing a novel detection system named TASR that can be used to differentiate TSS websites from legitimate technical support websites by leveraging features that are indispensable traits of TSS web pages. Our experimental results showed that TASR can detect 94.5% of the TSS links in web search results at a false positive rate of less than 1%, and that it significantly outperfors previous work in this area.

References

[1] Abusive experiences - Web Tools Help — support.google.com. https://support.google.com/webtools/answer/7347327?hl=en&ref_topic=7071812. [Accessed 20-Dec-2021].

[2] Average office rental in Delhi-NCR market declines 9— hindustantimes.com. https://www.hindustantimes.com/delhi-news/average-office-rental-in-delhi-ncr-market-declines-9-in-h1-2020/story-xUXkAwJOfSvXhScNblNQyM.html. [Accessed 20-Jan-2022].

[3] Chromium bugs: Allow window close during javascript alerts. https://bugs.chromium.org/p/chromium/issues/detail?id=673166, Last access on 2020-12-20, 2016.

[4] Chromium bugs: Cannot close entire chrome software or tab during a popup. https://bugs.chromium.org/p/chromium/issues/detail?id=714459, Last access on 2020-12-20, 2017.

[5] Chromium bugs: Web page hijacks chrome and won't allow normal quit. https://bugs.chromium.org/p/chromium/issues/detail?id=682855, Last access on 2020-12-20, 2017.

[6] Chromium bugs: Browser hang in m67 with excessive download attempts. https://bugs.chromium.org/p/chromium/issues/detail?id=860045, Last access on 2020-12-20, 2018.

[7] Chromium bugs: Browser hang with execessive download attempts. https://bugs.chromium.org/p/chromium/issues/detail?id=809775#c19, Last access on 2020-12-20, 2018.

[8] Chromium bugs: history.pushstate abuse hangs chrome. https://bugs.chromium.org/p/chromium/issues/detail?id=1038223, Last access on 2020-12-20, 2019.

[9] https://www.bbb.org, Last access on 2020-12-20, 2020.

[10] https://analytics.moz.com/pro/link-explorer/home, 2020.

[11] Chromium bugs: Malvertisers exploit infinite pushstate loop to freeze chrome. https://bugs.chromium.org/p/chromium/issues/detail?id=1113285, Last access on 2020-12-20, 2020.

[12] Vijay A Balasubramaniyan, Aamir Poonawalla, Mustaque Ahamad, Michael T Hunter, and Patrick Traynor. Pindr0p: Using single-ended audio features to determine call provenance. In *Proceedings of the 17th ACM conference on Computer and communications security*, pages 109–120, 2010.

[13] Michael Bernard, Chia Hui Liao, and Melissa Mills. The effects of font type and size on the legibility and reading time of online text by older adults. In *CHI'01 extended abstracts on Human factors in computing systems*, pages 175–176, 2001.

[14] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(Jan):993–1022, 2003.

[15] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Computer networks and ISDN systems*, 30(1-7):107–117, 1998.

[16] Jim Browning. The Refund Scam — youtube.com. https://www.youtube.com/watch?v=X4PllvUowaQ. [Accessed 20-Jan-2022].

[17] Jim Browning. Tech support scams. https://www.youtube.com/c/JimBrowning, Last access on 2021-01-22, 2022.

[18] Samantha Chatman. Cash app fake contact number scam steals thousands of dollars from users. https://abc11.com/cash-app-contact-number-card-bank/8134234/, Last access on 2021-5-01, 2020.

[19] Catalin Cimpanu. After microsoft complaints, indian police arrest tech support scammers at 26 call centers. https://www.zdnet.com/article/after-microsoft-complaints-indian-police-arrest-tech-support-scammers-at-26-call-centers/, 2018. [Accessed 10-Dec-2021].

[20] Thomas Claburn. The register: Google nuked tech support ads to kill off scammers. ok. it also blew away legit repair shops. not ok at all. https://www.theregister.com/2019/07/16/google_ban_on_tech_support_ads/, Last access on 2020-12-20, 2019.

[21] Sarah Coble. Delhi police bust call center scammers. https://www.infosecurity-magazine.com/news/delhi-police-bust-call-center/, 2021. [Accessed 10-Dec-2021].

[22] Federal Trade Commission. Ftc consumer information: Gift card scams. https://www.consumer.ftc.gov/articles/gift-card-scams, Last access on 2021-05-18, 2021.

[23] Federal Trade Commission. Protecting older consumers. https://www.ftc.gov/system/files/documents/reports/protecting-older-consumers-2020-2021-report-federal-trade-commission/protecting-older-consumers-report-508.pdf, Last access on 2022-01-04, 2021.

[24] Houk Consulting. Don't fall for tech support refund scams! https://www.houkconsulting.com/2020/06/tech-support-refund-scams/, Last access on 2021-05-18, 2021.

[25] Kun Du, Hao Yang, Zhou Li, Haixin Duan, and Kehuan Zhang. The {Ever-Changing} labyrinth: A {Large-Scale} analysis of wildcard {DNS} powered blackhat {SEO}. In *25th USENIX Security Symposium (USENIX Security 16)*, pages 245–262, 2016.

[26] Facebook. Facebook community standards: Fraud and deception. https://www.facebook.com/communitystandards/fraud_deception, Last access on 2021-05-01, 2021.

[27] Emma Flethcer. Older adults hardest hit by tech support scams. https://www.ftc.gov/news-events/blogs/data-spotlight/2019/03/older-adults-hardest-hit-tech-support-scams, Last access on 2021-04-20, 2019.

[28] Github. Dsjas - dave smith johnson and son. https://github.com/DSJAS/DSJAS, Last access on 2021-5-01, 2021.

[29] Google. Qualify your outbound links to Google. https://developers.google.com/search/docs/advanced/guidelines/qualify-outbound-links. [Accessed 25-Jan-2022].

[30] David Graff. Google ads and commerce blog: Restricting ads in third-party tech support services. https://www.blog.google/products/ads/restricting-ads-third-party-tech-support-services/, Last access on 2020-12-20, 2018.

[31] Payas Gupta, Roberto Perdisci, and Mustaque Ahamad. Towards measuring the role of phone numbers in twitter-advertised spam. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, pages 285–296, 2018.

[32] Payas Gupta, Bharat Srinivasan, Vijay Balasubramaniyan, and Mustaque Ahamad. Phoneypot: Data-driven understanding of telephony threats. In *NDSS*, 2015.

[33] David Harley, Martijn Grooten, Steven Burn, and Craig Johnston. My pc has 32,539 errors: how telephone support scams really work. *Virus Bulletin*, 2012.

[34] MALWAREBYTES LABS. Psa: Tech support scams pop-ups on the rise. https://blog.malwarebytes.com/threat-analysis/2014/11/psa-tech-support-scams-pop-ups-on-the-rise/, Last access on 2021-10-10.

[35] Matthias A Lee. pytesseract 0.3.7. https://pypi.org/project/pytesseract/, Last access on 2021-3-20, 2019.

[36] Kirill Levchenko, Andreas Pitsillidis, Neha Chachra, Brandon Enright, Márk Félegyházi, Chris Grier, Tristan Halvorson, Chris Kanich, Christian Kreibich, He Liu, Damon McCoy, Nicholas Weaver, Vern Paxson, Geoffrey M. Voelker, and Stefan Savage. Click trajectories: End-to-end analysis of the spam value chain. In *32nd IEEE Symposium on Security and Privacy, S&P 2011, 22-25 May 2011, Berkeley, California, USA*.

[37] Huichen Li, Xiaojun Xu, Chang Liu, Teng Ren, Kun Wu, Xuezhi Cao, Weinan Zhang, Yong Yu, and Dawn Song. A machine learning approach to prevent malicious calls over telephony networks. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 53–69. IEEE, 2018.

[38] Long Lu, Roberto Perdisci, and Wenke Lee. Surf: Detecting and measuring search poisoning. In *Proceedings of the 18th ACM Conference on Computer and Communications Security*, CCS '11, page 467–476, New York, NY, USA, 2011. Association for Computing Machinery.

[39] Mary L McHugh. Interrater reliability: the kappa statistic. *Biochemia medica*, 22(3):276–282, 2012.

[40] Najmeh Miramirkhani, Oleksii Starov, and Nick Nikiforakis. Dial one for scam: A large-scale analysis of technical support scams. In *24th Annual Network and Distributed System Security Symposium, NDSS 2017, San Diego, California, USA, February 26 - March 1, 2017*. The Internet Society, 2017.

[41] Cristina Miranda. Ftc consumer information: Scammers demand gift cards. https://www.consumer.ftc.gov/blog/2018/10/scammers-demand-gift-cards, Last access on 2021-05-18, 2018.

[42] Payscale. Average software developer salary in india. https://www.payscale.com/research/IN/Job=Software_Developer/Salary, Last access on 2021-5-24, 2021.

[43] Victor Le Pochat, Tom Van Goethem, Samaneh Tajalizadehkhoob, Maciej Korczyński, and Wouter Joosen. Tranco: A research-oriented top sites ranking hardened against manipulation. *arXiv preprint arXiv:1806.01156*, 2018.

[44] Sampsa Rauti and Ville Leppänen. "you have a potential hacker's infection": A study on technical support scams. In *2017 IEEE International Conference on Computer and Information Technology (CIT)*, pages 197–203. IEEE, 2017.

[45] Bharat Srinivasan, Athanasios Kountouras, Najmeh Miramirkhani, Monjur Alam, Nick Nikiforakis, Manos Antonakakis, and Mustaque Ahamad. Exposing search and advertisement abuse tactics and infrastructure of technical support scammers. In Pierre-Antoine Champin, Fabien Gandon, Mounia Lalmas, and Panagiotis G. Ipeirotis, editors, *Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018*, pages 319–328. ACM, 2018.

[46] Kurt Thomas, Danny Yuxing Huang, David Y. Wang, Elie Bursztein, Chris Grier, Tom Holt, Christopher Kruegel, Damon McCoy, Stefan Savage, and Giovanni Vigna. Framing dependencies introduced by underground commoditization. In *14th Annual Workshop on the Economics of Information Security, WEIS 2015, Delft, The Netherlands, 22-23 June, 2015*, 2015.

[47] tollfreenumber.org. The big list of toll-free phone number providers. https://tollfreenumber.org/toll-free-phone-number-providers/, Last access on 2021-05-18, 2021.

[48] Susan Tompor. Facing a summer delay? don't fall for fake travel sites. https://www.freep.com/story/money/personal-finance/susan-tompor/2019/08/05/fake-travel-sites-scam-consumers/1920164001/, Last access on 2021-5-01, 2019.

[49] Huahong Tu, Adam Doupé, Ziming Zhao, and Gail-Joon Ahn. Users really do answer telephone scams. In Nadia Heninger and Patrick Traynor, editors, *28th USENIX Security Symposium, USENIX Security 2019, Santa Clara, CA, USA, August 14-16, 2019*, pages 1327–1340. USENIX Association, 2019.

[50] Phani Vadrevu and Roberto Perdisci. What you see is NOT what you get: Discovering and tracking social engineering attack campaigns. In *Proceedings of the Internet Measurement Conference, IMC 2019, Amsterdam, The Netherlands, October 21-23, 2019*, pages 308–321. ACM, 2019.

[51] Tom Van Goethem, Najmeh Miramirkhani, Wouter Joosen, and Nick Nikiforakis. Purchased fame: Exploring the ecosystem of private blog networks. In *Proceedings of the 2019 ACM Asia Conference on Computer and Communications Security*, pages 366–378, 2019.

[52] Liz Walsh. Microsoft advertising blog: User safety policy revision (global). https://about.ads.microsoft.com/en-gb/blog/post/may-2016/user-safety-policy-revision-(global), Last access on 2020-12-20, 2016.

[53] Ronghai Yang, Xianbo Wang, Cheng Chi, Dawei Wang, Jiawei He, Siming Pang, and Wing Cheong Lau. Scalable detection of promotional website defacements in black hat {SEO} campaigns. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 3703–3720, 2021.

Appendix A. Facebook and WhatsApp Groups Analysis

Table 4 shows the details of data we collected from 10 different Facebook groups for the analysis in this paper. Along with the names and creation dates of the groups, the table also shows the average number of posts that are made in the group per day (as estimated by Facebook). The table also lists the distinct number of posts and their associated authors for the data that we collected. Interestingly the table shows that the many of the groups have existed more than 4 years with one particular active group being created in March 29th, 2015. This high age of these groups indicates that Facebook is not actively trying to curtail these groups despite the malicious nature of the posts being made in this groups as we will see in the rest of this section. It is to be noted that this is happening despite Facebook's policy to remove content that encourages or coordinates scams and money laundering activities [26]. The table also shows the large size of the groups with three of them having more than ten thousand members. Overall, the average size of these Facebook groups is about 7000 members.

Further, Table 4 shows that the number of posts and the number of members making posts involved in each of the Facebook groups during our 1 month crawl is uniformly high across all groups. The average values are about 2204 posts with an average of 525 authors per group. Table 5 shows a similar breakdown for the WhatApp groups. These groups are smaller compared to Facebook groups with each one having an average of 3331 posts from an average of 281 authors even though we extracted them from a much larger time frame of 9 months. This is likely due to the fact that WhatsApp has a limit of about 256 users in any group where as Facebook groups have no such limits. Further, Facebook's support for nested comments underneath each posts allows for much richer and complex group interaction mechanisms which seemed to be utilized well by the scammers as can be seen by the number of posts that received comments in Table 1.

All Facebook posts have unique user IDs which allowed us to track users across different groups. Similarly, all WhatsApp posts have a phone number by which we can track users making the posts. We used these to measure the total number of unique message posters in these groups. In total, we saw 2229 unique post authors in Facebook during the 1 month period and 3158 post authors in WhatsApp during the 9 month period. Please note that these numbers are smaller than the sums of number of message posters in Table 4 (5247) and Table 5 (3654), thus indicating that there is a significant number of users who are in multiple groups. Our analysis shows that about 20% of total users are members of atleast 3 TSS scam Facebook groups.

Qualitative Analysis. We applied an open coding approach on sample posts (about 2%) from Facebook and Whatsapp to derive an understanding of the various components that exist in the TSS ecosystem and their interplay as shown in Figure 1 and discussed in Section 2.2. During this analysis, we observed that posts from TSS Website developers often mention about "call prices" and the usage of "Google Adwords". On the other hand, posts advertising money laundering services often mention "card blocking" which appeared to be the TSS underground market term for money laundering services. The generated codes from this process allowed us to automatically label all posts and measure frequencies of these posts (Tables 1 and 2) as well as conduct more fine-grained measurements such as call prices in posts made by TSS webmasters (Table 7). The entire codebook used for labeling TSS posts is provided in Listing B.3.

Inter-rater Reliability Metrics. In order to verify the reliability of the labeling process above, we made use of two human labellers. The first labeller, L_1 is a co-author who was not directly involved in the generation of the codebook. Another labeller, L_2, is a non-author who was completely shielded from the codebook. Instead, this latter labeller was only provided an overview of the findings of our TSS ecosystem study with the help of Figure 1 in a 30 minutes session. This process enabled us to confirm the veracity of the labeling without any bias. Both the labellers were given a set of 100 posts randomly sampled from the entire Facebook dataset. Each labeler independently completed the labeling process by reading each post carefully and considering if it fits any (one or more) of the five categories we considered in Tables 1 and 2. If a post did not fit any of the five categories, it was given an "Other" label by them. We then compared these labels with the labels that were auto-generated by our

ID	Creation date	# Members	# Posts /day	# Posts	# Authors
F1	March 3, 2016	12.8 K	7	2126	692
F2	March 29, 2015	2.2 K	35	2399	421
F3	Aug 1, 2015	11.3 K	65	1982	573
F4	Dec 25, 2015	6.1 K	49	2351	473
F5	Dec 27, 2016	9.1 K	72	1833	596
F6	Dec 4, 2018	5.1 K	65	2219	560
F7	Feb 20, 2019	2.9 K	49	2433	463
F8	Jan 29, 2017	5.5 K	27	2110	424
F9	April 26, 2016	11.6 K	117	2045	617
F10	July 28, 2016	3.3 K	36	2545	428
Total	-	69.9 K	22043	5247	

TABLE 4: Breakdown of data collected from TSS Facebook groups

ID	Creation date	# Posts	# Authors
W1	Sept 8, 2019	4636	260
W2	Oct 15, 2020	550	76
W3	Sept 15, 2020	1071	114
W4	Aug 22, 2019	3116	383
W5	Feb 26, 2018	300	45
W6	Apr 24, 2019	3092	260
W7	Jan 14, 2018	423	47
W8	Aug 31, 2019	5550	395
W9	Sept 8, 2019	3925	445
W10	Sept 4, 2019	10274	633
W11	Nov 12, 2019	2130	206
W12	Nov 6, 2019	5268	505
W13	Nov 5, 2017	2972	285
Total	-	43307	3654

TABLE 5: Breakdown of data collected from TSS WhatsApp groups

codebook based on the preliminary sample analysis. The Cohen's Kappa score as well as a simple label agreement fraction between L_1, L_2 and the label vector generated by our codebook (denoted by C) are presented in Table 6. The table shows the Cohen's Kappa score to be close to a value around 0.8. which indicates a "strong" agreement [39] that is also corraborated by the high fraction of agreements between all pairs.

Metric	C vs. L_1	# C vs. L_2	# L_1 vs. L_2
Cohen's Kappa	0.80	0.77	0.87
Agreement	0.85	0.83	0.90

TABLE 6: Inter-rater reliability metrics for labeling TSS posts

Along with the codebook, a comprehensive list of TSS underground terminology we learned during the qualitative analysis process is presented in Appendix B.2 for future researchers.

Brand name	min	median	max
Amazon	$2.52	$4.90	$8.05
iOS	$3.15	$4.20	$9.10
PayPal	$5.11	$6.37	$7.70
Cash App	$7.70	$10.50	$11.90
Norton	$2.52	$4.90	$8.05
Roku	$5.46	$6.02	$10.50
EBay	$5.88	$8.40	$8.40
Windows	$5.60	$5.60	$5.60
Delta	$13.30	$13.30	$13.30

TABLE 7: Price (in USD) of TSS calls advertised by TSS web developers

Repeat Analysis. In 2022, we revisited these Facebook and WhatsApp groups and noticed that all of them are still active. We inspected 100 recent Facebook posts manually in one of the groups and found posts pertaining to all the categories covered in Table 1 showing that the ecosystem is still very active. Further, despite our small sample size, we noticed new brands such as Hulu and Fubo being targeted by the scammers which we did not notice in our earlier data. Moreover, we also saw hiring ads for call centers in a new location (Uttarkhand, India) which we have not seen in our prior data. This shows that the TSS ecosystem is still active and continuing to adapt.

Appendix B. TSS Cost Analysis and Return of Investment

Notice that what follows is only a back-of-the-envelope calculation, based on what we learned from TSS social media posts and information published in previous work, and other public information. We first give some necessary background quantitative data and then follow it up with Return of Investment calculations.

Background. TSS web developer often mention the "quality" of the calls (see example in Figure 3). This refers to the likelihood of the calls being from true gullible victims and not from "scam baiters" (e.g., from the scammer.info community) who frequently call the scammers pretending to be victims and waste the time and resources of the TSS scammers. Often, the TSS web developers also advertise metrics such as "Average Handling Time" (AHT) to denote how long the calls they are selling last on average which also gives the call centers an idea of the "gullibility" of the victims that call this campaign. The average advertised AHT value we came across in these services was about 23 minutes. By comparison, [40] showed that the average time taken by a TSS scammer to arrive to the point where they sell the scam services to a victim is 17 minutes.

We saw that toll-free number providers advertised a median price of $28 USD per month for unlimited incoming calling minutes for each phone number. As for sales of potential victim data, we found a single victim's data is being advertised for a median value of as much as ≃$1.40 USD.

We also measured the remuneration provided to the agents for working for the call centers. Interestingly, job description for TSS agents resembled those of a regular job requiring five-day work weeks. The advertised salaries have a median value between $700 and $900 USD per month. These can be deemed to be quite lucrative, as the average salary of a software developer in India is about $570 USD per month [42]. Moreover, the jobs provide additional benefits such as providing transportation service to and from the work place. Importantly, most of the positions advertise a "commission" to the agents on the scammed money: the agents are allowed to take a 30% cut. This likely keeps the agents motivated in scamming the victims. Finally, we found that money laundry services

hired by call centers typically charge as much as 35% to 55% of the money scammed from the victims.

ROI Calculation. Consider a small call center that employs 3 agents[3], each being paid a salary of $700 per month as per above findings. We assume the costs for setting up a minimalistic physical call center office space, including office and equipment rental, utilities such as electricity and internet, etc., to be about $600 per month[4]. Let us also assume each call to last about 23 minutes, as we found earlier. This allows each agent to take approximately 20 calls in 8 hours, thus allowing the call center to take about 1200 calls each month. Assuming an average cost of about $7 USD per each acquired TSS call (see Table 7), we estimate the cost of routing the calls to a TSS call center to be approximately $8400 USD thus bringing the total monthly cost for operating a TSS call center to about $11,100 USD. Assuming the money laundering cut to be around 40% and agent commission to be 30%, the center would need to make about 106 successful calls (i.e., when the callee is actually scammed into paying money) in a month in order for it to break even, which translates into an 8.8% phone scam success rate. However, this would not be sufficient to yield a profit to the center's owner. If the call center owner wants to make a profit of at least as much as their three employees (the agents) put together, then they will need to make about 185 successful calls per month, requiring a success rate of as much as 15.4% and yielding a profit of $8,325 USD per month for the owner of the call center.

Considering recent research on telephony scams which showed that about 1.17% of recipients of telephony scam calls get actually scammed by the calls [49], the success rates required for profitable TSS scams is very high. This difference may be explained by noticing that, unlike in [49], TSS calls involve the potential victims themselves making the calls, thus making the phone-based portion of the scam more targeted. However, a 10-fold in increase in success rate is not easily achieved by call centers. We also confirmed this anecdotally, by posing as a TSS call center owner and interacting with fellow TSS scammer. The scammer, a TSS call center operator himself, advised that TSS operations present a large investment risk, as they require substantial capital. The scammer advised that it is very important to hire good agents who are very well trained and experienced in the scam topics/brands being targeted, which typically translates into a much higher success rate (i.e., more victims paying the scammer).

# Appearances	TSS (61,999)	BTS-T (75,810)	BTS-O (1,638)
0	26.9%	49%	83.9%
1	23.5%	33%	12.8%
2	17.6%	9.9%	1.7%
3-5	20.4%	6.9%	1.6%
6-10	8.9%	1.1%	0
10+	2.7%	0.1%	0

TABLE 8: Appearances of phone numbers on web pages

3. TSS call sale posts often explicitly cite the minimum number of agents that need to work in a call center with 3 being the most popular requirement.

4. We considered a 500 square foot office in the New Delhi area which typically costs about 78 Indian Rupees (= $1 USD) per square foot [2] and added a conservative figure of $100 for addtional expenses such as equipment rental and utilities such as electricity and internet.

	TSS	BTS-T
2000 and before	28 (2.1%)	450 (18.8%)
2001-2005	40 (3.1%)	606 (25.3%)
2006-2010	39 (3%)	629 (26.2%)
2011-2015	154 (11.8%)	552 (23%)
2016-2020	1,042 (80%)	163 (6.7%)

TABLE 9: Distribution of Domain Name Registration Year for Ground Truth Websites

B.1. Screenshots of various posts on TSS groups and TSS sites

Figures 13, 14, 15 and 16 show some examples of different types of posts related to TSS operations that we gathered from TSS groups on Facebook.

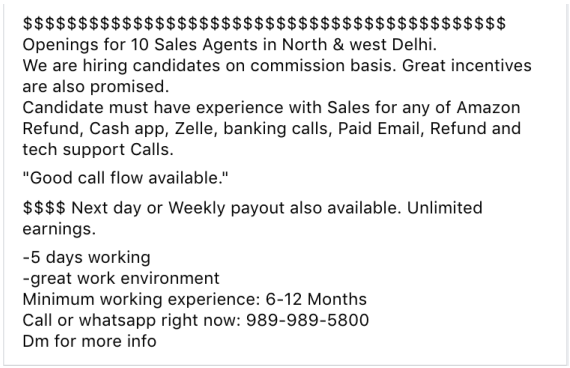


Figure 13: A Facebook post advertising TSS agent job in Delhi, India. Note the explicit requirement for experience in prior scam calls targeting various brands.

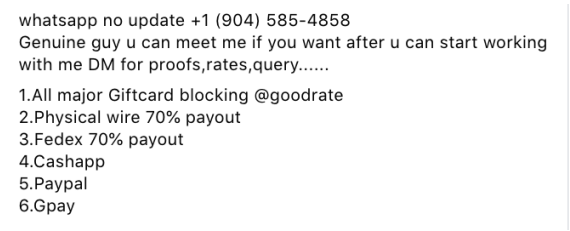


Figure 14: A Facebook Post advertising money laundering services for TSS call centers.

B.2. Glossary of Underground TSS Terminology

The criminals in Technical Support Scam ecosystem have evolved to use their own lingo over the past few years. This makes it difficult for a security analyst to fully understand many of these posts although the majority of conversations are still in English. For this, we are providing this glossary of underground TSS terminology in order to help future analysts studying TSS.

1) **Age filter**: When TSS ecosystem participants advertise sale of calls with age filters they are typically refering to the fact that the call has been generated with an advertisement that was targeted specifically at older age populations.
2) **AHT**: Stands for Average Handling Time (usually in minutes). This is a metric used in call centers to refer

Sequence	Topic Label	Example Brand or Product	# Related TSS Websites
1	Printer & Scanners	Hp/Brother/Dell/Canon	206
2	Streaming devices / TV	Roku/Firestick/Comcast/ Direct TV/ Charter	51
3	Digital streaming	Netflix/ Hulu	35
4	Airlines	Delta	22
5	Email	Yahoo / Outlook/ Aol / Gmail	193
6	Routers & Modems & Network	TP-Link / Linksys / Belkin	75
7	Desktop Apps	All web browsers / Adobe acrobat reader / Turbotax	67
8	Cash Transfer	Paypal / Chime	46
9	Financial Accounting	Quickbooks/ ADP/ Sage/ Quicken/ Reckon/ Xero	50
10	Antivirus	Norton/ AVG/ McAfee/ Malwarebyte	115
11	OS Support	Windows/ Android/ MacOS	62
12	Personal Computers	Dell/ HP/ Mac/ IBM PC/ Acer/ Sony	106
13	Personal Virtual Assistant	Alexa/ Echo/ Google Home	10
14	Smart - IoT Devices	Arlo/ Honeywell/ Ring doorbells / Nest thermostat	6
15	Socila Media Apps	Twitter/ Facebook/ Instagram	57
16	Handheld Devices	Kindle/ iPad/ iPhone/ LG/ Garmin	33
17	VOIP Devices	Magicjack	4
18	Game Support	Pogo / Xbox	18
19	WordPress	WordPress	3
20	E-commerce	Amazon/ Ebay	18
21	All-Support	All-Support	16

TABLE 10: Breakdown of TSS Website Topics in Our Collection

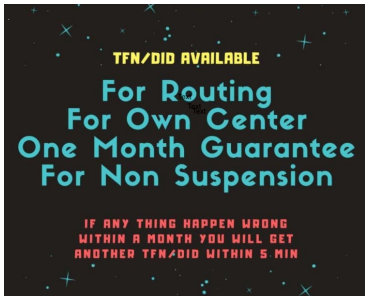


Figure 15: A Facebook post advertising toll-free number services to TSS call centers. Note how the ad guarantees "bullet-proof" Toll Free numbers.

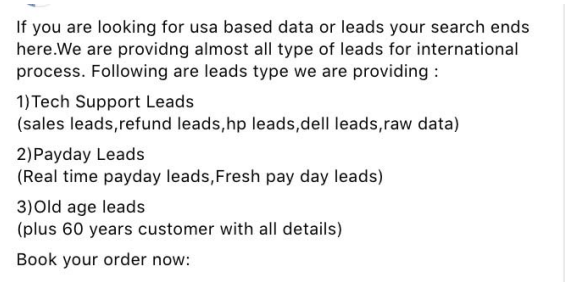


Figure 16: A Facebook post advertising sale of victim data

to the average length of a call. In the TSS ecosystem, this metric is announced in call sale advertisements in order to indicate the potential vulnerability of the callers. A high AHT (such as 20 minutes or more) is preferred by the call centers as it indicates a high degree of vulnerable callers who can be engaged in long conversations.

3) **Blasting data**: "Blasting" refers to the process of cold calling or sending targeted phishing emails. This is an alternative mechanism to the web-based victimization method we studied in this paper. The data usually includes name and location as well as e-mail or phone number which can then be used for making targeted calls or e-mails.
4) **Blocking**: "Blocking" is the process of laundering money from victims in the form of gift cards or money transfer services such as Cash App (see Fig. 14).
5) **BSOD**: Stands for Blue Screen of Death. These refer to the aggressive TSS ads that typically show the Windows crash screen as studied in [40] and [50].
6) **Email calls**: Some TSS scammers also generate calls by sending targeted phishing emails to potential victims and then sell them to the call centers. These are referred to as Email calls.
7) **F2F**: Stands for "Face to Face". Although scammers get introduced via platforms such as Facebook and WhatsApp, some of them prefer to do their first transaction in person in order to establish trust. This term is used for such an in-person meeting request.
8) **GC**: Stands for Gift Card. Gift cards are one of the predominant means by which TSS scammers steal money from victims.
9) **IVR data**: This is the same as "Blasting data" above. However, this term is specifically used in reference to cold calls only. The buyers of this data use it to generate robo calls to victims via Interactive Voice Response (IVR) technology and then connect the responsive victims to live TSS agents.
10) **Popup calls**: Popup calls refer to calls generated from victims via TSS ads distributed via lax advertising networks as studied in [50]. In general, these are aggressive TSS attacks in which the popup claims that the system/phone has crashed or has been infected with a virus.
11) **PPC calls**: Stands for Pay Per Click. These refer to TSS attacks calls which are generated via paid search engine advertising from networks such as Google Ads.
12) **Quality**: The quality of calls generally refers to the potential likelihood of the callers to be victimized into buying the fake products offered by the scammers. TSS call sellers generally tend to advertise "high quality" of their calls and quote high AHT figures and age filters in support of this.

13) **Refund calls/data**: An alternative scam strategy in which TSS scammers target a prior victim with the pretext of returning their money but rather steal more from them by making some changes to HTML source code on their logged in bank accounts [24], [16].
14) **"Taking remote"**: The process of taking control of a victim's computer in order to simulate a non-existent system issue and convince the victim to pay for support services.
15) **TFN**: Stand for Toll Free Number. These are often used by TSS scammers in their websites in order to lend an air of legitimacy to the phone number and increase the likelihood of victims calling.
16) **"@44"**: The @ sign followed by a number is usually indicated in "blocking" service ads to indicate the rate at which a US dollar is converted to Indian Rupees after factoring in the cut taken by the money launderer.

B.3. Codebook for TSS post analysis

```

# Post is tagged as being related to "hiring of TSS agents" (2)
# if this method does not return False.
def agents(msg):
    looking = ('looking' in msg or 'hiring' in msg or
               'required' in msg or 'need' in msg)
    agent = 'agent' in msg or 'fresher' in msg

    if agent and looking:
        return 'need agents'
    return False


# Post is tagged as being related to "money launderers" (3)
# advertising services if this method does not return False.
def money_laundering(msg):
    available = 'available' in msg
    payment = 'payment' in msg or 'payout' in msg
    blocking = 'blocking' in msg

    giftcard = ('gift card' in msg or 'giftcard' in msg or
           'gifts card' in msg)
    if giftcard and (blocking or available or payment):
        return 'giftcard'

    gateway = ('gatewy' in msg or 'gateway' in msg or
               'gatway' in msg)
    payment_form = (gateway or payment or 'loader' in msg or
                    'merchant' in msg or 'link' in msg)
    two_d = ('2d' in msg or '2-d' in msg or '2 d' in msg or
             '3d' in msg or '3-d' in msg or '3 d' in msg or
             'e check' in msg)
    if two_d and payment_form:
        return '2d or 3d gateway'

    bank = 'bank account' in msg
    if bank:
        return 'bank account'

    seller = ('sell' in msg or 'buy' in msg or
              'trader' in msg or 'flash' in msg)
    bitcoin = 'btc' in msg
    if bitcoin and seller:
        return 'bitcoin'

    loader = 'loader' in msg
    card = ('mastercard' in msg or 'americanexpress' in msg or
            'amex' in msg)
    if card and loader and available:
        return 'card'

    different_payments = ('chime' in msg or 'paypal' in msg or
                          'zelle' in msg or 'g-pay' in msg or
                          'apple' in msg)
    if different_payments and payment:
        return 'other payment forms'
    if blocking and (payment or available):
        return 'blocking available'

    return False


# Post is tagged as being from "TSS web masters" (4)
# advertising call sales if this method does not return False.
def calls(msg):
    available = ('avail' in msg or 'active' in msg or
                 'running' in msg or 'book' in msg or
                 'live' in msg or ' cc ' in msg or
                 'order more' in msg or 'direct center' in msg)
    call = 'call' in msg or 'cll' in msg
    call_or_available = call and available

    google_ads = (('adword' in msg or 'ppc' in msg) and
                  ('campaign' in msg or 'account' in msg or
                   call or available))
    if google_ads:
        return 'google ads'

    if 'amazon' in msg and call_or_available:
        return 'amazon'
    if 'ios' in msg and call_or_available:
        return 'ios'
    if 'paypal' in msg and call_or_available:
        return 'paypal'
    if 'chime' in msg and call_or_available:
        return 'chime'
    if (('cashapp' in msg or 'cash app' in msg) and
        call_or_available):
        return 'cashapp'
    if ('quickbook' in msg) and call_or_available:
        return 'quickbook'
    if 'printer' in msg and call_or_available:
        return 'printer'
    if 'popup' in msg and call_or_available:
        return 'popup'
    if 'ebay' in msg and call_or_available:
        return 'ebay'
    if 'delta' in msg and call_or_available:
        return 'delta'
    if 'bsod' in msg and call_or_available:
        return 'bsod'
    if 'refund' in msg and call_or_available:
        return 'refund'
```

```
    if (' cc details ' in msg or 'indian cc' in
    msg or
        ' cc data ' in msg):
        return 'query for call center'
    if 'website traffic available' in msg or (
    call and available):
        return 'calls available'
    return False


# Post is tagged as being from "Toll-free number
     providers" (5)
# advertising services if this method does not
    return False.
def tfn(msg):
    available = ('avail' in msg or 'providing'
    in msg or
                 'get unlimited call' in msg)
    toll_free_number = (('tfn' in msg or 'toll
    free' in msg or
                         'toll-free' in msg) and
    available)
    did = ' did ' in msg and available

    if toll_free_number or did:
        return True
    return False

# Post is tagged as being related to "victim
    data sales" (6)
# advertising call sales if this method does not
     return False.
def victim_data(msg):
    data = 'data avail' in msg
    available = 'avail' in msg
    email = 'email lead' in msg and available
    refund = 'refund lead' in msg and available
    ivr = 'ivr lead' in msg and available
    fresh = 'fresh lead' in msg and available

    if data or email or refund or ivr or fresh:
        return True
    return False
```

Listing 1: Python version of the complete codebook produced by our qualitative analysis of TSS posts for categorization. Note that the same post can be "tagged" with multiple category labels.