# Beyond App Markets: Demystifying Underground Mobile App Distribution Via Telegram

YANHUI GUO, Beijing University of Posts and Telecommunications, China
DONG WANG, Beijing University of Posts and Telecommunications, China
LIU WANG, Beijing University of Posts and Telecommunications, China
YONGSHENG FANG, Beijing University of Posts and Telecommunications, China
CHAO WANG[†], Huazhong University of Science and Technology, China
MINGHUI YANG, OPPO, China
TIANMING LIU[*†], Monash University, Australia and Huazhong University of Science and Technology, China
HAOYU WANG[*†], Huazhong University of Science and Technology, China

The thriving mobile app ecosystem encompasses a wide range of functionalities. However, within this ecosystem, a subset of apps provides illicit services such as gambling and pornography to pursue economic gains, collectively referred to as "underground economy apps". While previous studies have examined these apps' characteristics and identification methods, investigations into their distribution via platforms beyond app markets (like Telegram) remain scarce, which has emerged as a crucial channel for underground activities and cybercrime due to the robust encryption and user anonymity.

This study provides the first comprehensive exploration of the underground mobile app ecosystem on Telegram. Overcoming the complexities of the Telegram environment, we build a novel dataset and analyze the prevalence, promotional strategies, and characteristics of these apps. Our findings reveal the significant prevalence of these apps on Telegram, with the total sum of subscription user numbers across channels promoting these apps equivalent to 1% of Telegram's user base[1]. We find these apps primarily cater to gambling and pornography services. We uncover sophisticated promotional strategies involving complex networks of apps, websites, users, and channels, and identify significant gaps in Telegram's content moderation capabilities. Our analysis also exposes the misuse of iOS features for app distribution and the prevalence of malicious behaviors in these apps. This research not only enhances our understanding of the underground app ecosystem but also provides valuable insights for developing effective regulatory measures and protecting

[*]Corresponding authors: Haoyu Wang (haoyuwang@hust.edu.cn) and Tianming Liu (Tianming.Liu@monash.edu).
[†]The full name of the authors' affiliation with Huazhong University of Science and Technology is: Hubei Key Laboratory of Distributed System Security, Hubei Engineering Research Center on Big Data Security, School of Cyber Science and Engineering, Huazhong University of Science and Technology.
[1]This percentage is calculated using total subscription user numbers without accounting for potential user overlaps, please refer to §4.2 for a detailed explanation and discussion of implications.

Authors' Contact Information: Yanhui Guo, Beijing University of Posts and Telecommunications, Beijing, China; Dong Wang, Beijing University of Posts and Telecommunications, Beijing, China; Liu Wang, Beijing University of Posts and Telecommunications, Beijing, China; Yongsheng Fang, Beijing University of Posts and Telecommunications, Beijing, China; Chao Wang, Huazhong University of Science and Technology, Wuhan, Hubei, China; Minghui Yang, OPPO, Shenzhen, Guangdong, China; Tianming Liu, Monash University, Melbourne, Victoria, Australia and Huazhong University of Science and Technology, Wuhan, Hubei, China; Haoyu Wang, Huazhong University of Science and Technology, Wuhan, Hubei, China.


ACM 2476-1249/2024/12-ART33
https://doi.org/10.1145/3700432

users from potential risks associated with these covert operations. Our findings provide implications for platform regulators, app market operators, law enforcement agencies, and cybersecurity professionals in combating the proliferation of underground apps on encrypted messaging platforms.

CCS Concepts: • **Security and privacy** → *Mobile and wireless security*.

Additional Key Words and Phrases: Underground Mobile Apps, Telegram, App Distribution, Cybercrime

**ACM Reference Format:**
Yanhui Guo, Dong Wang, Liu Wang, Yongsheng Fang, Chao Wang, Minghui Yang, Tianming Liu, and Haoyu Wang. 2024. Beyond App Markets: Demystifying Underground Mobile App Distribution Via Telegram. *Proc. ACM Meas. Anal. Comput. Syst.* 8, 3, Article 33 (December 2024), 25 pages. https://doi.org/10.1145/3700432

## 1 Introduction

Mobile apps have developed into an extensive and complex ecosystem, covering various fields ranging from entertainment to education. However, within this process, a portion of apps serve the underground economy, driven by the pursuit of significant economic gains, giving rise to so-called "underground economy apps" [44]. These apps provide non-compliant services in sensitive areas such as gambling and pornography, violating or skirting legal regulations, posing significant challenges to digital security and law enforcement, thereby attracting wide attention from academia and industry. Current research on underground apps has covered the promotional websites of these apps [43, 46, 47], the characteristics of underground apps [45, 46, 48–50], and methods for identifying these apps [44, 67].

Telegram is a widely popular instant messaging platform, whose encryption features and commitment to user anonymity have made it a preferred channel for the dissemination of underground economy and cybercrime content [5, 28]. While investigations into Telegram's underground activities, including conspiracy channels [51], identification of clone and fake channels [53], and analysis of misinformation distribution [54], have emerged, studies examining the underground mobile app ecosystem within Telegram are still missing, which is crucial for developing targeted regulatory strategies and protecting users from potential risks.

This research is dedicated to exploring the ecosystem of underground mobile apps within the Telegram environment (**TUApps** for short). We aim to provide comprehensive insights into the TUApps ecosystem, including the entities involved, supporting networks, operational methods, and their impact on users and the digital marketplace. To achieve these objectives, our research is structured around three key research questions (§3.1): What is the prevalence and accessibility of underground mobile apps on Telegram? What are the strategies for promoting these apps on Telegram? And, what are the development and behavioral characteristics of these apps?

The first challenge in this research was to obtain a set of underground apps from Telegram. Telegram has a large and widely distributed user base, making the filtering and identification of underground activities inefficient and resource-intensive. Our breakthrough came upon finding that Telegram hosts bots equipped with search capabilities, allowing us to efficiently trawl through channels using specified keywords. To enhance our dataset's breadth and depth, we expanded our channel list by incorporating popular channels and employing a snowballing technique (§3.2.1). We then extracted website URLs promoting underground apps from the message histories of these channels (§3.2.2), and retrieved the apps from these promotional websites with a semi-automated crawling method (§3.2.3). This process established a robust foundation for our analysis of the underground app ecosystem on Telegram. Subsequently, we conducted an in-depth exploration of the TUApps ecosystem, guided by our three research questions (§4,5,6). Our findings underscore the urgent need for enhanced regulation of underground apps on Telegram and reveal several critical implications. Our research also illuminates potential avenues for mitigation, offering valuable

insights not only to platform regulators but also to a broader range of stakeholders including app market operators and law enforcement agencies (§7.1).

In summary, the contributions of this paper are as follows:

- We conducted a comprehensive study on the ecosystem of underground apps on Telegram, which is the first of its kind. We contributed a novel dataset[2] comprising over 200 million messages, 10,000 apps downloaded, and 1,000 unique apps. Our research investigated multiple aspects of these apps, including their prevalence, promotional strategies, and characteristics.
- We uncovered the significant presence of TUApps on Telegram, nominally reaching 1% of the platform's user base. These apps primarily cater to gambling and pornography services. In terms of availability, they are distributed mainly through APK, IPA, and Web Clip files rather than being available on official app stores. Notably, we identified the misuse of iOS features like TestFlight, Enterprise Signing, and Web Clips for TUApp distribution, revealing vulnerabilities in the iOS ecosystem.
- We unveiled sophisticated promotional strategies of TUApps, detailing a complex promotional framework comprising interlinked relationships among apps, promotional websites, users, and channels. Our analysis revealed that promotional websites use diverse hosting strategies while maintaining structural URL similarities, offering potential detection avenues. We identified significant gaps in Telegram's ability to detect potentially harmful content, with only 0.28% of promotional channels flagged as scams. We also revealed the presence of dedicated app promotion teams active on Telegram.
- We analyzed the development and behavioral characteristics of TUApps, revealing that a quarter of them are malicious, posing substantial security risks. We found their preference for cross-platform development frameworks such as Flutter for ease of development. In terms of payment methods, TUApps frequently utilize fourth-party payment platforms to evade regulatory scrutiny.

## 2 Background

### 2.1 Telegram

Telegram is a cross-platform encrypted instant messaging service with over 800 million users [7] that originated in 2013, and is known for its security, speed, and rich features. It allows users to communicate one-on-one through text, images, videos, and files. Additionally, Telegram provides channel and group functionalities, enabling users to interact within broader social circles.

With a maximum capacity of 200,000 members, groups support interactive communication among multiple users, whereas channels are designed for broadcast communication from administrators to an unlimited audience. Telegram allows these channels and groups to be set as public (i.e., can be found with Telegram search, and every user can access historical messages and join the group) or private (i.e., only members can see the posts, and users need an invite link to join it). Despite the differentiation between channels and groups, this distinction is inconsequential for our research. Throughout this research, we employ the term "channel" to encompass both channels and groups.

Due to the anonymity and the convenience of Telegram communication, it has become a natural platform for the dissemination of underground economy and cybercrime, such as money laundering [6], financial fraud [4], revenge porn [2], and the trade of personal information [9]. It has been pointed out by many news media and reports [4, 5, 8, 28] that Telegram has evolved into a "cybercrime ecosystem" resembling dark web forums, attracting criminals such as cyber thieves

[2]Our code and dataset are available at https://github.com/security-pride/TUApps. Kindly refer to §A for more details on availability.

and hackers. To boost platform security and user trust, Telegram has introduced the "verified" and the "scam" marks to channels and accounts to alert users about potential fraudulent activities.

## 2.2 Bots in Telegram

Telegram offers the Bot API [32] to developers, enabling them to create bots with diverse functionalities tailored to specific requirements. These functionalities encompass information retrieval, automation, and productivity tools, among others. One notable type of bot is the "search bot", which utilizes user-provided keywords to conduct channel searches and curate a selection of channels matching those keywords. This empowers users to discover and explore channels that closely align with their interests and preferences.

Based on our preliminary investigation, we discovered that search bots are frequently utilized by underground actors for advertising and promotion purposes. The working mechanism of these search bots revolves around keywords, which are sold by the bot owners for profit. When users search for a keyword that has been purchased, the buyer's content is strategically displayed at a higher rank or in a pinned position, ensuring maximum visibility.

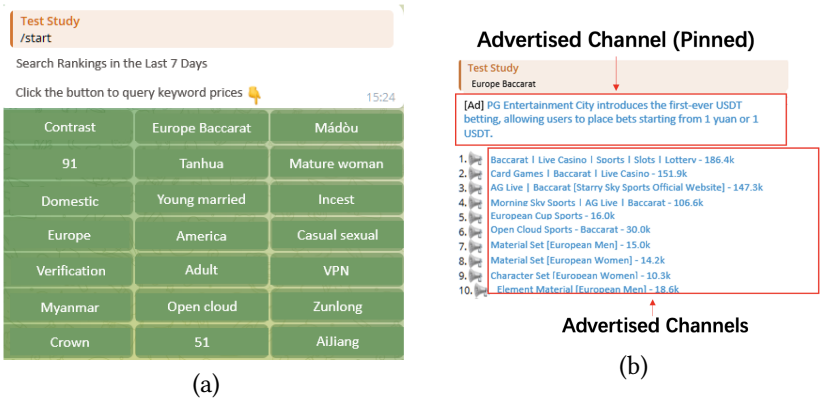



Fig. 1. Keywords List in Search Bot and Search Results for "Europe Baccarat"

A prime example of such a search bot is "Search Groups Chinese Search" [31], which boasts a user base of over 64,000 Telegram users. As illustrated in Figure 1(a), the bot offers a wide array of keywords for sale, with the majority pertaining to the underground economy. When a user searches for one of these keywords to find related channels, the search bot returns results that prioritize the buyer's promotional content. For instance, as depicted in Figure 1(b), a search for gambling content related to "Europe Baccarat[3]" yields results where the pinned item and the top 10 search results all feature advertised channels associated with "Europe Baccarat" gambling activities. The prices of these ad slots ranged from $660 to $750.

## 2.3 App Installation Package

Android and iOS, the two dominant mobile operating systems, employ distinct app installation package formats and distribution mechanisms.

On the Android platform, apps are packaged in the APK (Android Package Kit) format. The open nature of the Android ecosystem allows for app distribution through multiple channels, including the official Google Play Store, third-party marketplaces, and direct APK file downloads. While

[3]A popular card game in casinos.

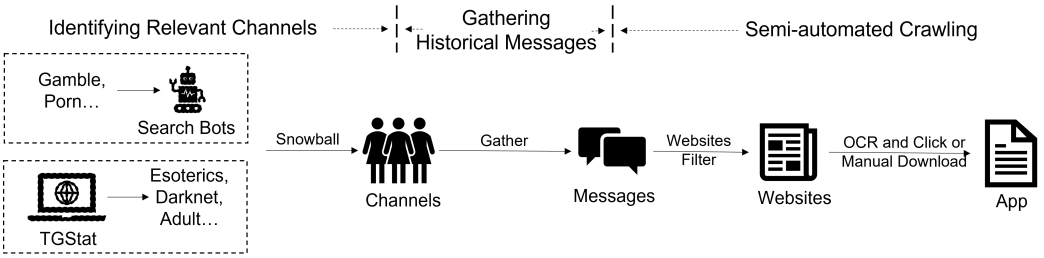



Fig. 2. Dataset Collection

Android systems by default restrict the installation of APKs from unknown sources, users can easily modify this setting. This flexibility inadvertently facilitates underground app developers in guiding users to bypass security restrictions.

Conversely, the iOS platform uses the IPA (iOS App Store Package) format for app installation. iOS enforces stricter controls, generally prohibiting the installation of IPA files from unauthorized sources. Ordinary users are typically limited to downloading and installing apps exclusively through the official Apple App Store. However, iOS does provide alternative distribution methods, ostensibly for specific purposes such as app testing. These include TestFlight, Enterprise Signing, and Web Clip. Our research reveals that these mechanisms are being exploited to distribute underground apps, a topic we will explore in greater detail in §4.4.

## 3 Study Design

### 3.1 Research Questions

Our study is driven by the following research questions (RQs):

- **RQ1: What is the prevalence and accessibility of underground mobile apps on Telegram?** This question aims to explore the types of underground economy these apps serve, their distribution scope on Telegram channels, their availability (or lack thereof) on official app stores, and the various distribution methods employed to make these apps accessible to users, with particular attention to the unique restrictions on the iOS platform.
- **RQ2: What are the strategies for promoting underground mobile apps on Telegram?** This question examines the intricate promotional networks that facilitate the spread of underground mobile apps within Telegram. Focusing on the roles of users, channels, and websites, we aim to unravel how these entities collaborate to amplify the visibility and accessibility of these apps.
- **RQ3: What are the characteristics of the underground mobile apps promoted on Telegram?** With these apps typically engaging in illicit activities or distributing prohibited content, and opting for Telegram to sidestep detection and scrutiny, it becomes imperative to investigate their attributes. This includes an examination of their development and behavior characteristics, to acquire a thorough understanding of how these underground apps function.

### 3.2 Dataset Collection

To answer our research questions, we collect underground apps promoted within public Telegram channels that could be associated with underground activities. Figure 2 shows our data collection process, which consists of three steps: (1) identifying relevant channels in Telegram, (2) obtaining historical messages from channels, and (3) semi-automated crawling of apps.

*3.2.1 Identifying Relevant Channels.* To sample relevant channels, we leverage Telegram's "search" function and the snowballing-based technique to retrieve a set of channels related to underground activities. Our approach was threefold:

(1) As previously mentioned, search bots are often employed to promote underground content. We hence leverage search bots to kick-start our channel acquisition process. To identify effective search bots, we refer to the chart of top-100 bots on TelegramChannels [33], a popular third-party channel and bot ranking platform. We supplemented this selection with 8 additional bots discovered through our daily interactions on the platform. As introduced in §2.2, these bots offer keywords related to underground activities for sale. We thus curated a collection of nearly 300 keywords associated with underground activities in both English and Chinese from the purchasable keywords listed in these search bots[4]. These keywords were employed in our interactions with these bots to curate a contextually relevant collection of channels. This yields an initial list of 15,796 channels.

(2) Considering that underground apps may also be promoted through regular channels, we enhance our channel collecting strategy. Specifically, we rely on the rankings from a third-party website, TGStat [30], that provides extensive statistics and analytics for Telegram channels and groups[5]. TGStat categorizes Telegram channels into over 40 categories, including regular categories such as News and Tech, as well as categories related to underground economy such as Darknet and Adult. We extract the top 100 channels with the most subscribers from each category listed on TGStat. This yields another list of 4,700 popular channels covering various categories.

(3) With the above two lists of channels as 'seed channels', we next incrementally expand our channel list with potentially relevant public channels whose messages have been forwarded to existing seed channels. Such a snowballing approach involves crawling the historical messages of these seed channels (as detailed in §3.2.2), parsing the message content, and extracting channel URLs from which messages had been forwarded. This effort yielded an extended list of 71,332 public Telegram channels for our research.

*3.2.2 Gathering Historical Messages from Channels.* Following the compilation of a list of potentially relevant channels, we next crawled historical messages from these channels. The historical messages were collected using Telegram's open API and Telethon Python library [34]. In principle, all messages ever posted in a public channel are available. However, considering the time overhead of the crawling effort and the timeliness of the messages (web links contained in earlier messages are likely to be dead), we limited our data scraping to a maximum of 100,000 messages from each channel. In total, we collected over 200 million messages. This extensive message dataset allows us to explore the dissemination of underground apps within the Telegram ecosystem.

*3.2.3 Semi-automated Crawling of Apps.* Our goal is to find underground mobile apps circulating in messages on the Telegram platform. Typically, these apps are promoted through web pages or download links. We therefore extract website URLs from the messages using regular expression matching. Due to the substantial volume of URLs (over 8 million) and the presence of irrelevant websites, we implement an additional filter to identify key URLs. Specifically, we filter by the prevalence of the promoted URLs, i.e., the number of channels promoting them, focusing on URLs that are mentioned in more than 20 different channels. As a result, we obtain a total of 21,147 website URLs being promoted in over 20 channels. These selected URLs account for 65.4% of the entire promotion frequency of the 8 million URLs, offering a concentrated view of the TUApps landscape while ensuring the process remains streamlined and effective.

[4]The bot and keyword lists are available on our website.

[5]TGStat is also frequently utilized by previous research on Telegram, such as [52, 53, 63]. We use a different website for bot ranking because TGStat provides no such service.

We then develop a crawler to retrieve the mobile apps linked to each of the selected website URLs (if available). The scraper utilized an open-source OCR toolkit PaddleOCR [25], and the open-source webdriver Selenium [29], which allows one to control a browser programmatically. For each website, if accessible, we initiated the process by capturing a screenshot of the webpage. Subsequently, we employ PaddleOCR to recognize the text within the screenshot. If any of the identified text matches keywords associated with app downloads such as 'Download', 'Android', or 'iOS', we determine the text's location and use Selenium to automatically trigger clicks on these locations to initiate app downloads. To ensure the accuracy of the collection, we further performed a manual examination of the screenshots from the accessed websites. For downloaded apps, we check the screenshots to verify their relevance to underground economy. For cases where websites failed to download automatically, we examined their screenshots to identify any contained apps, and if present, we manually downloaded the respective packages to ensure the inclusion of relevant apps that might have evaded the automated process.

### 3.3 Dataset Overview

Our dataset collection process lasted for 6 months from September 2023 to February 2024. The dataset overview is presented in Table 1. In total, we collected 71,332 potentially relevant public channels, from which we obtained over 200 million historical messages and extracted a total of 8,283,875 website URLs from the messages. Of these, 21,147 URLs were promoted by more than 20 channels, which we visited in turn to retrieve their linked mobile apps. As a result, 5,692 websites successfully executed downloads, from which we collected 5,692 Android installation files and 4,968 iOS installation files[6], corresponding to 557 unique Android apps and 517 unique iOS apps after de-duplication. These apps were derived from 2,333,830 messages promoted in 6,137 channels, posted by 7,957 Telegram users.

Table 1. Overview of the Dataset

| Item | Count |
|---|---|
| Public Channels | 71,332 |
| Messages | Over 200 million |
| Total Websites | 8,283,875 |
| Android Apps Downloaded | 5,692 |
| iOS Apps Downloaded | 4,968 |
| Unique Android Apps | 557 |
| Unique iOS Apps | 517 |

## 4 Prevalence and Accessibility of TUApps

This section examines the prevalence and accessibility of TUApps. We explore the types of underground economy these apps serve, analyze their distribution scope across Telegram channels, investigate their presence (or absence) on public app stores, and detail the various forms employed for their distribution.

### 4.1 Classification of TUApps

To understand the types of underground economy associated with the apps in our dataset, we conducted a manual classification[7] of the 557 apps.

[6]Typically, Android and iOS apps are released simultaneously, meaning that on a website, users can download both the Android and iOS versions. However, some websites only provide the Android version.

[7]Two authors independently performed the classification and then convened to compare and discuss the results.

As shown in Table 2, the apps were categorized into five distinct groups: pornography, gambling, pirated software, virtual private network (VPN), and cryptocurrency trading platforms. Notably, pornography and gambling apps constituted the overwhelming majority of the dataset, accounting for over 95% of the total apps.

Table 2. Classification of TUApps

| Types of Underground Economy | # Count | % Percent |
|---|---|---|
| Pornography | 314 | 56.4 |
| Gambling | 228 | 40.9 |
| Pirated Software | 10 | 1.8 |
| Virtual Private Network (VPN) | 3 | 0.5 |
| Cryptocurrency Trading Platform | 2 | 0.3 |

### 4.2 Distribution Scope of TUApps

To assess the prevalence of TUApps on Telegram, we analyzed their distribution scope, focusing on the number of channels promoting each app and the total promotional messages across these channels. Our analysis revealed a significant presence, with each app, on average, being promoted 4,190 times across 37.17 channels. Examining the 6,137 channels involved in TUApp promotion, we found an average of 1,409 members per channel, with a total subscription base of 8,647,033 members across all channels. **This substantial reach, before accounting for potential overlaps, constitutes 1.08% of Telegram's reported 800 million users** as of 2023 [7]. The most widely distributed app in our dataset, a gambling app called "Zunlong Kai Shi", was advertised across 216 distinct channels to 217,512 users, with an astonishing 357,951 promotional messages in total. It is important to note that, due to ethical concerns, our data collection methodology recorded only channel membership numbers without identifying individual users. This approach likely leads to an overestimation of unique individuals exposed to TUApps, as users may be members of multiple channels, suggesting that the actual number of unique individuals is lower than the total sum of 8,647,033 members. However, it is crucial to recognize that our data collection process, as detailed in §3.2, captures only a partial snapshot of the underground app's presence in the Telegram ecosystem. Our method, which included crawling relevant channels and messages, and sampling URLs occurring in more than 20 different channels, while effective in offering a concentrated view of the TUApps landscape, cannot encompass the entire scope of underground app activity on Telegram. Consequently, while user overlap may inflate the total subscription numbers, the true reach of underground apps on Telegram may still be underestimated.

These findings underscore the intensive and wide-ranging promotional efforts undertaken for TUApps on Telegram, while also highlighting their prevalence on the platform. The broad distribution of promotional activities across multiple channels suggests that app developers and promoters employ diverse strategies to maximize their reach to target audiences. This approach is likely driven by the intense competition within the TUApps market on Telegram. We further investigate their promotional strategies in §5.

### 4.3 Public Availability of TUApps

We investigated the public availability of the collected apps in our study by searching for their presence in mainstream app markets.

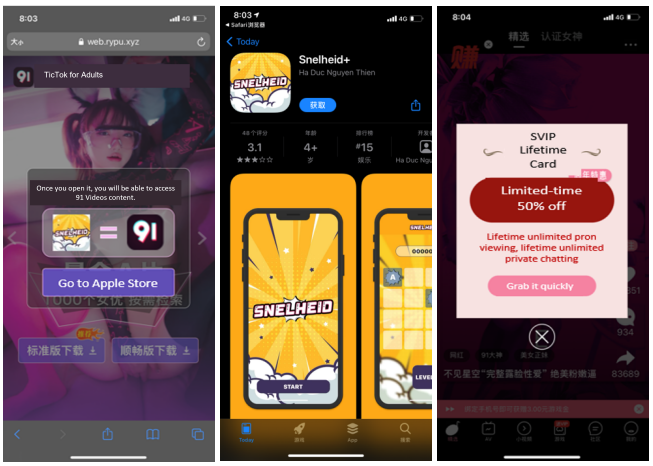
For Android apps, we first used Androguard [10] to extract the package name of each app in our dataset. We then searched for these package names in the official Android marketplace, Google

Play [17], and popular third-party app markets such as Xiaomi [38] and Huawei [18] App Store. None of the package names from our dataset were found to be indexed in these app markets.

For iOS apps, we extracted each app's name and unique bundle identifier (bundleId) from the *Info.plist* file's `CFBundleDisplayName` and `CFBundleIdentifier` fields in IPA files, and the `PayloadDisplayName` field in iOS Web Clips (elaborated in §4.4). We then systematically searched for the extracted app names using the iTunes Search API [13]. For each app, we retrieved the bundleId from the meta-information in its search results for accurate app matching.

Interestingly, we found 12 matches among the iOS apps. Upon closer examination, we discovered that these 12 apps were promoted on Telegram with official Apple App Store links, and they were underground apps disguised as legitimate apps on the App Store. A most recent research [67] also referred to such apps as "mask apps". Figure 3(a) shows an example of one such app being promoted on a website within Telegram. The app, which leads users to the App Store for downloads, is actually a pornographic app masquerading as a regular app. Instead of downloading the app under its original name "TicTok for Adults", users are directed to download a game app named "Snelheid+", as shown in Figure 3(b). The app's icon and description on the App Store appear innocuous. However, upon downloading and opening the app, it is revealed to be filled with pornographic content, as depicted in Figure 3(c). It is worth noting that by the time of writing, all 12 apps were no longer available on the Apple App Store, likely due to Apple's app vetting process.

Our findings reveal that despite Apple's stringent app vetting procedures, underground actors still find ways to promote their apps using the official App Store. However, such cases are rare on iOS (12 out of 517), and we found no similar instances on Android. This disparity can be attributed to the ease with which Android APKs can be promoted and installed outside of official app markets.




(a) (b) (c)

Fig. 3. Underground App in AppStore

### 4.4 Distribution Forms of TUApps

In terms of distribution forms, Android TUApps are packaged as APK files, while iOS TUApps are distributed as IPA files and Web Clips [14]. Due to the iOS restriction on direct IPA file installation, the IPA files are distributed through alternative methods such as Enterprise Signing [12] and TestFlight invitations [35]. As shown in Table 3, of the 517 unique iOS apps in our collection, 497

were available as Web Clips, while the remaining 20 IPA files were distributed using Enterprise Signing (16) and TestFlight invitations (4). We elaborate on these three distribution forms below:

Table 3. Distribution Forms of iOS TUApps

| Forms | # Unique App Count | # Samples Count | % Samples Percent |
|---|---|---|---|
| TestFlight Invitation | 4 | 316 | 6.4 |
| Enterprise Signing | 16 | 805 | 16.2 |
| Web Clip Installation | 497 | 3,847 | 77.4 |

**TestFlight Invitation.** Our dataset included 316 samples (6.4% of 4,968 downloaded iOS samples) distributed via TestFlight. This Apple-provided platform for app testing allows developers to invite up to 10,000 external testers through public links, with apps remaining valid for 90 days. TestFlight's relatively lenient review standards, compared to the App Store, facilitate the distribution of certain underground apps. Users only need to install TestFlight and follow the invitation link to access these apps. Some distributors also share App Store accounts to circumvent tester limitations.
**Enterprise Signing.** We identified 805 samples (16.2%) distributed through Enterprise Signing. This method allows companies to deploy proprietary apps directly to iOS devices in a controlled and customized manner, bypassing the App Store and Apple's standard review process. A key feature of Enterprise Signing is the absence of restrictions on expiration dates or the number of users, making it an attractive option for disseminating underground apps. However, it is important to note that while both TestFlight and Enterprise Signing undergo less stringent vetting compared to App Store submissions, they remain within Apple's regulatory purview. Apple has been known to ban accounts misusing these methods for illegitimate purposes [3]. This likely contributes to the lower prevalence of these distribution methods in our dataset compared to Web Clips.
**Web Clip Installation.** The majority of iOS TUApp samples in our dataset (3,847, 77.4%) were distributed as Web Clips. This iOS feature enables users to add customized shortcut icons to their device's home screen, providing easy access to specific web pages or settings [14]. The process of setting up Web Clips is significantly less complex than guiding users through TestFlight invitations or Enterprise Signing procedures. Despite Web Clips offering only a shortcut to a specific web page rather than a full app experience, most underground actors prefer this method over distributing IPA files on iOS platforms. This preference stems from Web Clips' lack of regulation, as they fall outside Apple's app review process, their affordability due to not requiring a costly personal or enterprise Apple developer account, and their user-friendly nature, demanding minimal effort from users during installation. These advantages have made Web Clips the favored route for disseminating a significant volume of iOS TUApps. This finding aligns with previous research on the prevalence of Web Clips in the underground economy [49], underscoring the method's effectiveness in circumventing traditional app distribution channels.

The diversification of these distribution strategies demonstrates that despite iOS's stringent security measures, underground actors continue to discover and exploit loopholes for content distribution. This ongoing cat-and-mouse game underscores the pressing need for more robust regulatory measures and enhanced vetting processes to mitigate abuse of these distribution channels.

**Answer to RQ1:** Our analysis reveals that TUApps predominantly cater to pornography and gambling services. These apps exhibit a significant presence on Telegram, with each app being distributed across an average of 37.17 channels in 4,190 messages, and collectively reaching 1% of Telegram's user base at face value. While official app stores generally do not list these apps, we observed rare instances of their temporary presence on the Apple App Store, which were swiftly

removed upon detection. On iOS, the distribution landscape is diverse: Web Clips emerge as the primary distribution method, supplemented by instances of TestFlight and Enterprise Signing abuse for IPA distribution.

## 5 TUApps Promotion in Telegram

This section examines the multifaceted promotional ecosystem of underground apps on Telegram through three key aspects: First, we analyze the promotional strategies employed, revealing the techniques used to maximize app visibility and user engagement. Next, we investigate the existence and structure of organized promotion teams, shedding light on the collaborative efforts behind app distribution. Finally, we examine the characteristics of promotional websites, including their longevity and infrastructure, to understand their role in supporting TUApp distribution.

### 5.1 Promotion Strategy

Our research unravels the promotional networks of TUApps by tracing their origins, which involve three key entities: *websites* that host app files, Telegram *users* who promote these websites through messages, and *channels* that provide promotional space for these users. Figure 4 illustrates this intricate promotional model of TUApps using the case of a pornography app "91 Production", demonstrating how the app is distributed across multiple websites, each promoted by different users within various channels. Our analysis of this model reveals several noteworthy observations.

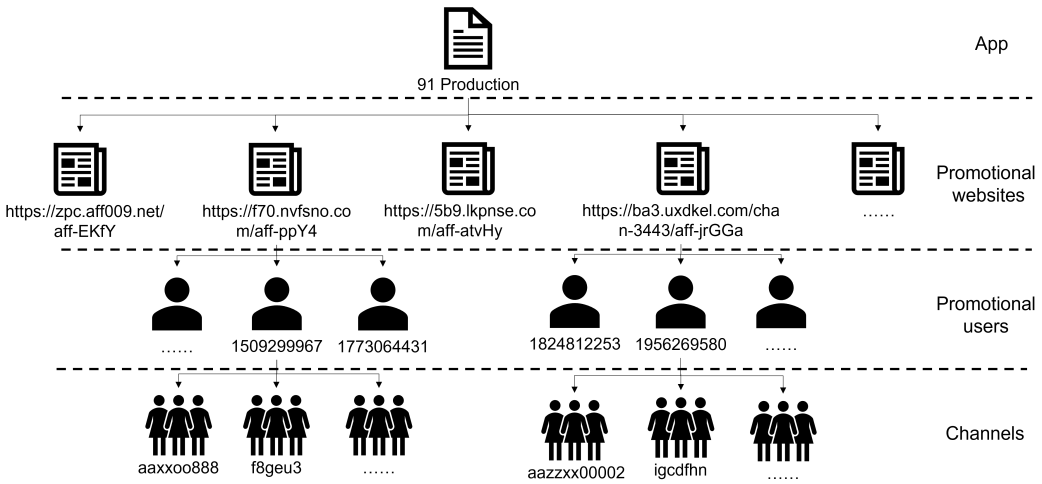



Fig. 4. Promotion Model of TUApps

*5.1.1 Promotional Websites.* We identified two significant patterns regarding websites:

- **Diverse Hosting**: Promotional websites for the same TUApp tend to utilize different hosts. On average, only 11.3% of websites promoting the same TUApp share identical secondary hosts. This diversity suggests a deliberate strategy of constantly generating new hosts for TUApp promotion, potentially as a means to evade regulatory detection and action.
- **URL Structural Similarities**: Despite the host diversity, we observed striking similarities in URL structures and formats used to promote the same app. For instance, as shown in Figure 4, websites promoting "91 Production" often incorporated the common string "aff-" at the end of

their URLs. Additionally, these URLs frequently exhibited similar segmentation patterns when split by special characters (such as "/", "-", and ".").

To validate our observations on URL structural similarities, we conducted a detailed analysis of URLs promoting each TUApp in our dataset. Our methodology involved segmenting the URLs based on common delimiters (e.g., /, -, ?, &) and excluding common segments such as top-level domains (e.g., com, net) and protocol prefixes (http, https, www). We then analyzed the remaining segments, regardless of their length. For each app, we calculated two metrics: the proportion of URLs sharing at least one unique segment with another URL promoting the same app; and the proportion of URLs having the same number of segments as the most common segment count for that app. Averaging these metrics across all apps in our dataset, our analysis revealed that: 91.46% of URLs promoting the same app contained at least one identical segment with another URL for that app. 72.74% of URLs maintained a consistent number of segments within each app's promotional URL set. Further analysis of segment lengths showed that segments between 3 and 6 characters long constitute the majority, accounting for 80.67% of all segments and 80.75% of identical segments across URLs. These structural similarities in URL composition, despite varying hosts, present a potential avenue for detection and mitigation strategies against TUApp promotion.

*5.1.2 Promotional Users.* Our analysis of the 7,957 users involved in TUApp promotion revealed a sophisticated and diversified approach to app marketing. These users demonstrate a high degree of versatility, rarely limiting themselves to promoting a single website or app. Instead, they engage in extensive promotion, advertising a wide range of sites associated with multiple apps across numerous channels. On average, each promoter advocates for 5.7 distinct TUApps, utilizes 74.32 unique URLs in their promotional efforts, and operates across 13.72 different channels.

Furthermore, we identified three primary promotional strategies employed by these users:

- Top-of-Chat Advertisements: Channel administrators leverage Telegram's pinning feature to continuously display TUApp ads at the top of the chat interface to to maximize visibility.
- Greeting Advertisements for Newcomers: Automated welcome messages containing ads for TUApps, set up by channel administrators, are sent to new members upon joining.
- Regular Advertisement Messages: Promotional users periodically post ads within channels to maintain consistent visibility of TUApps.

*5.1.3 Promotional Channels.* We conducted a comprehensive analysis of 6,137 channels involved in disseminating TUApps. Our investigation revealed that, on average, each channel hosted 8.29 promotional users, collectively promoting 7.4 apps through 293.31 promotional messages.

We sought to determine whether these channels were regular channels unrelated to the underground economy or specifically dedicated to serving underground content. To investigate this, we employed a stratified sampling approach based on promotional message frequency. Using quantile-based classification, we categorized channels into three groups: low density (below the first quartile, Q1), medium density (between Q1 and the third quartile, Q3), and high density (above Q3). We then randomly selected 50 channels from each category for manual inspection. Among them, high-density channels (more than 23.42 promotional messages per day) and medium-density channels (7.69 to 23.42) were all dedicated to serving underground content. Even among low-density channels (less than 7.69), only 4 out of 50 were normal channels unrelated to underground activities.

These underground-focused channels typically revolved around specific themes such as pornography or gambling. They provided a range of services including app download links, user guides, and enticing offers promising quick money-making opportunities or free access to restricted content. Some channels also incentivized user growth by offering rewards to existing users for successfully inviting new members. Notably, despite Telegram's efforts to alert users through "scam" channel

tags, we observed a significant gap in the platform's ability to identify and flag potentially harmful channels. Out of the 6,137 promotional channels analyzed, only 17 (0.28% of the total) were marked as scams. While not all promotional channels are necessarily scams, this low percentage suggests a considerable challenge in Telegram's capabilities to identify potentially harmful channels.

### 5.2 Promotion Team Identification

We next investigate the existence of organized promotion teams among the 7,957 promotional users. Our hypothesis posited that users promoting the same website are likely members of the same team. To test this hypothesis and identify potential teams, we utilized the Louvain algorithm [41], a widely recognized method for community detection in large networks. The Louvain algorithm is renowned for its effectiveness in social network analysis and community detection within computer science research as exemplified in [58, 59, 66]. We applied this algorithm to our dataset by constructing an undirected graph where nodes represent users and edges represent shared promotional websites. The weight of each edge was determined by the number of promotional sites common to the connected users. Our constructed graph comprised 7,957 nodes (users) interconnected by 49,607 edges (shared promotional websites). The Louvain algorithm partitioned this graph into 681 distinct communities, potentially representing different promotion teams. To evaluate the quality of this community division, we employed the modularity metric, which ranges from -0.5 to 1, with higher values indicating stronger community structures. Our network exhibited a modularity of 0.955, suggesting a robust and well-defined community structure.

Our analysis of the community structure revealed a highly skewed distribution of users across clusters. The majority of communities (81%, 552 out of 681) contained between 3 to 10 users, with a median size of 6 users. The smallest community had 3 users, while the largest contained 348 users. Notably, only 8 communities exceeded 100 users. This distribution indicates that the promotion network is characterized by numerous small teams, with a few exceptionally large communities.

Following our community detection, we examined the promotional patterns within the top 20 identified communities. This analysis revealed two distinct categories of promotional teams: those focused on promoting a single TUApp and those engaged in promoting multiple TUApps. Notably, among the top 20 identified teams, 17 were involved in promoting multiple apps. This prevalence of multi-app promotional teams strongly suggests the existence of specialized underground app promotion services operating independently from app development teams on Telegram.

To further investigate whether apps promoted by the same team were developed by the same entity, we conducted an app clustering analysis based on their similarity. We employed FSquaDRA2 [68] to calculate similarity scores between each pair of apps, forming clusters based on these scores. While initially designed as an app repackaging detection tool, FSquaDRA2 has proven versatile in broader app similarity analysis and clustering tasks. Its effectiveness in these areas has been demonstrated in several recent studies [46, 50, 55]. The tool's ability to assess app similarity makes it well-suited for our purpose of identifying apps with shared development origins or significant structural similarities. Following established research practices, we set a clustering threshold of 0.8, grouping apps with similarity scores exceeding this value. Additionally, we clustered apps with identical digital signatures, as this strongly indicates a common development origin. This analysis resulted in the distribution of 557 Android apps into 41 clusters, with 32 clusters comprising multiple apps. Table 4 presents an overview of the five largest clusters, detailing the number of apps per cluster, signature quantities, and observed patterns in package and MainActivity naming.

We then mapped the promotion teams to these app clusters, revealing a many-to-many relationship between promotion teams and development teams. Among the top 20 promotion teams, 7 exclusively promoted apps from a single development team, while the remaining 13 promoted apps from multiple development teams. This finding not only confirms the existence of organized

Table 4. Top 5 App Clusters Information

| App Count | Signature Count | Package Name | MainActivity Name |
|---|---|---|---|
| 27 | 3 | live.*.* | com.tbone.*.MainActivity |
| 15 | 2 | tv.*.* | com.example.*.MainActivity |
| 13 | 1 | net.*.* | com.tast.*.MainActivity |
| 13 | 2 | vip.*.* | com.awjq.*.MainActivity |
| 8 | 1 | gov.*.* | com.pron.*.MainActivity |

app promotion teams operating independently of development teams but also aligns with our observations from search bot interactions, where promotion teams proactively reached out and offered their services to Telegram users.

These results highlight the complex ecosystem of underground app promotion on Telegram, suggesting a sophisticated and adaptable underground economy capable of efficiently distributing and promoting illicit apps across the platform.

### 5.3 Promotional Website Analysis

To gain insights into the operational characteristics of websites promoting TUApps, we conducted an analysis focusing on their survival time and infrastructure details.

*5.3.1 Website Operational Duration.* We assessed the operational lifespan of promotional websites using Python's requests library [27] to perform daily status checks. Websites returning a 200 HTTP status code were considered active, while any other response indicated inactivity. Our analysis of website longevity revealed several key findings:

- Overall, promotional websites for underground apps exhibited short lifespans, with an average operational duration of just 24 days, indicating their transient nature.
- Websites promoting pornography apps demonstrated notably short lifespans, with an average operational duration of 18.5 days. Nearly 90% of these sites ceased operations within a month, suggesting a high-turnover, high-risk environment.
- Gambling app websites demonstrated relatively greater resilience, maintaining an average lifespan of 33.2 days. Some outliers in this category showed exceptional longevity, remaining active for up to six months, potentially indicating a more stable and lucrative market.

The generally short lifespan of these promotional websites underscores the challenges faced by both operators in maintaining their online presence and by authorities in tracking and regulating these rapidly shifting platforms.

*5.3.2 Infrastructure and Registration Analysis.* We further investigated the network infrastructure and registration details of the promotional websites. We employed the *ipinfo.io* [19] service to identify the Autonomous System Numbers (ASNs) corresponding to the websites' IP addresses. Domain registrar information was obtained through *whois* [37] searches. Key findings include:

- As shown in Table 5, a significant number of ASNs were predominantly hosted in the United States, China, and Malaysia.
- Table 6 illustrates the domain registrar landscape, with *GoDaddy* and *Name.com* collectively managing more domains than the combined total of the next eight competitors.

This concentration of hosting and registration services suggests potential focal points for future regulatory or investigative efforts in combating the distribution of underground apps.

Table 5. Top 10 ASNs of Promotion Websites

| ASN | ASN Description | Count | Country |
|---|---|---|---|
| AS13335 | Cloudflare, Inc. | 898 | US |
| AS55720 | Gigabit Hosting Sdn Bhd | 501 | MY |
| AS16509 | Amazon.com, Inc. | 494 | US |
| AS36351 | SoftLayer Technologies Inc. | 458 | US |
| AS45090 | Shenzhen Tencent Computer Systems Company Limited | 253 | CN |
| AS140227 | Hong Kong Communications International Co., Limited | 215 | CN |
| AS134548 | DXTL Tseung Kwan O Service | 211 | CN |
| AS37963 | Hangzhou Alibaba Advertising Co.,Ltd. | 205 | CN |
| AS4847 | China Networks Inter-Exchange | 104 | CN |
| AS45062 | Netease-Network | 104 | CN |

Table 6. Top 10 Registrars of Promotion Websites

| Registrar | Count |
|---|---|
| GoDaddy.com, LLC | 1118 |
| Name.com, Inc. | 745 |
| Gname.com Pte. Ltd. | 321 |
| eName Technology Co.,Ltd. | 182 |
| Alibaba Cloud Computing (Beijing) Co., Ltd. | 152 |
| GANDI SAS | 151 |
| Alibaba Cloud Computing Co., Ltd. (Wanwang) | 84 |
| Alibaba Cloud Computing Ltd. d/b/a HiChina | 76 |
| Go Daddy, LLC | 65 |
| NameSilo, LL | 58 |

**Answer to RQ2:** The promotion of underground apps on Telegram reveals a sophisticated ecosystem characterized by intricate relationships among apps, websites, users, and channels. Our analysis uncovered that promotional websites employ diverse hosting strategies while maintaining structural similarities in URLs, potentially offering avenues for detection and mitigation. The vast majority of channels involved in app promotion are dedicated to underground content, with only 0.28% flagged as scams by Telegram, indicating significant gaps in platform-level detection. We identified organized promotion teams, some specializing in single apps while others manage multiple apps, showcasing a mature and adaptable underground economy. Promotional websites for TUApps demonstrate remarkably short operational periods of merely 24 days on average. The network infrastructure supporting these promotional activities is primarily concentrated in the United States and China. The complex promotional landscape underscores the challenges in regulating and mitigating the distribution of underground apps on encrypted messaging platforms.

## 6 Characterizing TUApps

This section provides an in-depth analysis of the characteristics of TUApps, focusing on their development features and behavioral patterns.

### 6.1 App Development Characteristics

We investigate the key development characteristics of TUApps, exploring three critical aspects: development frameworks, third-party library integration, and app signatures. This analysis provides insights into the technical choices and strategies adopted by underground app developers.

*6.1.1 Development Framework Analysis.* To understand the technological foundation of TUApps, we examined the development frameworks used in their creation. Following a methodology similar to [48], we analyzed 35 popular mobile app development frameworks, focusing on their distinctive features such as dynamic link libraries (.so) and header files (.h). We then performed matching within each app to identify the frameworks used. Our analysis revealed a significant preference for cross-platform development frameworks among TUApp developers. The four most commonly used frameworks were: Flutter (39.0%), React Native (6.6%), Cordova (5.6%), and Unity3D (3.9%). Notably, Flutter [15] emerged as the dominant choice, likely due to its ease of use and rapid development capabilities. This preference for Flutter in TUApps (39.0%) stands in stark contrast to its usage in regular apps, which according to AppBrain [11], is only 4.88%. The prevalence of cross-platform frameworks in TUApp development presents unique challenges for app analysis. Traditional app analysis tools, including static analysis frameworks like FlowDroid [40] and dynamic analysis tools such as DroidBot [56], are primarily designed for native apps and offer limited support for cross-platform apps. This discrepancy highlights a critical need for advanced analytical tools capable of effectively examining cross-platform apps, particularly in the context of identifying and analyzing underground apps.

*6.1.2 Third-Party Library Analysis.* We utilized LibRadar [57] to analyze and catalog the embedded third-party resources in each app. Our investigation revealed that TUApps incorporate, on average, approximately 14.3 external libraries. While many of these libraries, such as OKHttp and Fastjson, are commonly used in regular app development, we observed some notable anomalies. Surprisingly, the most frequently incorporated library was ZXing [39] (71.1%, used in 396/557 apps), a tool for generating and decoding QR codes. This prevalence suggests a significant reliance on QR code functionality within TUApps, potentially for sharing app links or facilitating transactions. Additionally, we found a high frequency of payment-related libraries, with AliPay being particularly prominent (64.1%). This frequent integration of payment capabilities indicates that many TUApps are designed with monetization as a key feature, likely facilitating direct financial transactions within the apps.

*6.1.3 Signature Analysis.* We used Androguard [10] and plistlib [26] to extract the signatures of Android and iOS apps. A closer inspection of the certifying authorities revealed stark distinctions between the iOS and Android ecosystems, as shown in Table 7. For iOS apps, the certifying entities turned out to be corporations that could be identified through search engines, suggesting the possible misappropriation of signature validation. Conversely, the credentials identifying Android apps' certifying authorities were often ambiguous, and app developers seemed to deliberately obscure their identities.

### 6.2 App Behavior Characteristics

This section investigates the behavioral patterns of TUApps to identify potential security concerns. Our analysis encompasses app network traffic, maliciousness assessment, and payment method examination.

*6.2.1 Traffic Analysis.* To analyze app network traffic, we conducted automated testing of all apps while capturing their network communications. We employed BurpSuite [16], a widely-used

Table 7. Top 5 Signatures Authorities for iOS and Android

| iOS | Android |
|---|---|
| Nan Chang Wo Ai Wo Jia Technology CO.Ltd | dd |
| Efka - Iiektronnikons Ethnikos Foreas Koinokis Asfalisis | gjlc |
| AVIATION INFORMATION AND TELECOMMUNICTIONS JSC | momo13 |
| Baretch (International) Information Network Ltd | AW |
| EARLY MAKERS GROUP | baichaocan2022 |

cybersecurity tool, for intercepting and analyzing network traffic. Our testing methodology varied by platform. For Android apps, we installed apps on a Pixel 3 device via ADB and used FastBot [42] for automated testing. For iOS apps, We used ideviceinstaller to install IPA files to an iPhone 8 and employed macaca [24] for automated testing. For web clips, we extracted and tested URLs directly in the Safari browser. Each app test ran for five minutes, during which we captured not only network traffic, but also UI screenshots after each step of automated testing for the subsequent maliciousness analysis.

Our examination of the network traffic associated with underground apps revealed patterns in domain usage and server affiliations. Table 8 illustrates the most frequently utilized top-level domains (TLDs) in this traffic. While traditional generic top-level domains (gTLDs) such as .com and .org dominate, we observed a notable trend towards newer gTLDs like .vip and .app, as well as country code top-level domains (ccTLDs) such as .me. The preference for these alternative TLDs can be attributed to their cost-effectiveness and relatively relaxed management policies, making them particularly attractive for underground app operations.

Table 8. Top-level Domain Distribution

| Top-level Domain | Category | # count | % Percent |
|---|---|---|---|
| .com | gTLD | 5,472 | 62.40 |
| .me | ccTLD | 704 | 8.02 |
| .vip | New gTLD | 576 | 6.56 |
| .org | gTLD | 544 | 6.20 |
| .app | New gTLD | 480 | 5.47 |
| .top | New gTLD | 288 | 3.64 |
| .shop | New gTLD | 160 | 1.82 |
| .xyz | New gTLD | 160 | 1.82 |

Extending our analysis beyond TLDs, we examined server affiliations within the network traffic, focusing on ASNs and domain registrars. We employed methodologies similar to those used in analyzing promotional websites (§5.3.2). This investigation yielded results similar to those observed in promotional websites: ASNs were predominantly hosted in the United States and China, with GoDaddy emerging as the most prevalent domain registrar. These parallel findings underscore the interconnected nature of the underground app ecosystem, where similar infrastructure choices are made across app operation and promotion. This consistency may offer insights for identifying and potentially mitigating the spread of underground apps.

*6.2.2 Maliciousness Analysis.* Our maliciousness analysis began by scanning each TUApp using VirusTotal [36], a leading online platform that aggregates antivirus results from over 60 security vendors. Given a sample, VirusTotal aggregates the detection labels from various antivirus engines and reports how many of them classify the sample as “malicious”. Generally, this number serves

as an indicator to assess the level of a sample's maliciousness [65], i.e., the larger the number, the greater the maliciousness. As a result, 142 apps (representing 25.5% of the dataset) were flagged as malicious by VirusTotal engines. The number of engines identifying these apps as malicious ranged from 4 to 26, with an average of 15.04 engines. Given that many researchers consider detection by more than 4 engines sufficient to deem a sample malicious [60, 62, 69], this finding suggests that TUApps exhibit a significant level of maliciousness.

To further classify these malicious apps, we employed AVClass2 [61], a widely-used malware family labeling tool. The apps were categorized into various families. Table 9 presents the top 5 malware families identified, detailing their prevalence and typical behaviors. 'Jiagu' was the most prevalent, identified in 32 apps (22.5%), and is known for packing app code to evade detection. This was followed by 'Piom', a Trojan that spies on user behavior, and 'Softpulse', which displays numerous unwanted ads. This family distribution highlights the diverse malicious behaviors present in the TUApps dataset.

Table 9. Top 5 Malware Families

| Famliy Name | # Count | % Percent | Famliy Description |
|---|---|---|---|
| jiagu | 32 | 22.5 | Packs app code to evade detection |
| piom | 31 | 21.8 | Trojan that spies on user behavior [1, 23] |
| softpulse | 25 | 17.6 | Display numerous unwanted ads [20] |
| boogr | 19 | 13.4 | Downloads other malicious files [21] |
| mobtes | 17 | 11.9 | Performs stealthy destruction actions [22] |

Digging deeper into their behavior, our manual inspection of the screenshots captured during automated testing revealed two key findings:

- Deceptive Practices: Many apps employed misleading icons and names, mimicking popular legitimate apps to lure users and disguise their true functionality. Table 10 illustrates examples of such deceptive naming practices.
- Superuser Requests: We identified 47 pornography apps requested superuser rights at startup, with 24 apps checking for root status. These apps would terminate if superuser access was denied on rooted devices. Notably, this behavior was absent in gambling apps.

Table 10. Underground Apps Mimicking Popular Apps

| Underground App | Mimicked App |
|---|---|
| DiDi | DiDi |
| Pilipili | Bilibili |
| JinDong | JingDong |
| Kuaishou Express | Kuaishou |
| Adult Bilibili | Bilibili |

These findings underscore the potential security risks associated with TUApps and highlight the importance of robust security measures in identifying and mitigating threats within the underground app ecosystem.

*6.2.3 Payment Behavior.* To analyze payment behaviors in TUApps, we conducted a manual examination of a random sample of 100 apps. Among these, 83 offered functionalities for purchasing memberships or recharging accounts. We simulated the order placement process in these 83 apps, halting just before the actual payment step. Our investigation uncovered a sophisticated

payment system employed by 69 of the examined apps. These apps redirected users to external web browsers to complete transactions through fourth-party payment platforms [46]. These fourth-party platforms serve as intermediaries, routing payment requests from the apps to well-known third-party services such as WeChat Pay or Alipay. Table 11 provides examples of five fourth-party platforms we identified during our analysis.

Table 11. Examples of Fourth-party Payment Services Identified

| Platform | Alipay | WeChat Pay |
|---|---|---|
| http://ngxkdeja.xingyuwl.xyz | Yes | No |
| http://www.weepay.top | No | Yes |
| https://wapcashier.ysepay.com | Yes | No |
| http://pj888.005c.com | Yes | No |
| http://xin.xinma.work | No | Yes |

A key feature of this payment system is its use of dynamic virtual accounts for each transaction. This practice effectively obscures the ultimate destination of funds, creating a maze of financial transfers that is challenging to trace. This payment mechanism offers a high degree of anonymity through its multi-layered forwarding and complex account utilization. Such structures are also susceptible to exploitation for money laundering activities [64]. Potential money launderers could leverage these intricate payment processes to convert illicit funds into seemingly legitimate transactions. This complexity adds another layer of challenge to regulatory and law enforcement efforts in monitoring and controlling the financial aspects of the underground app ecosystem.

**Answer to RQ3:** Our analysis of TUApps reveals distinctive characteristics in both app development and behavior. On the development front, TUApps predominantly favor Flutter for ease of development, heavily integrate QR code and payment libraries, and exhibit significant differences in app signatures between Android and iOS platforms. Behaviorally, TUApps employ a mix of traditional gTLDs and new TLDs, with their network infrastructure primarily concentrated in the United States and China. Notably, one-fourth of the analyzed TUApps are classified as malware, displaying a high degree of maliciousness. Furthermore, these apps frequently utilize fourth-party payment platforms, likely as a strategy to evade regulatory scrutiny. These findings underscore the sophisticated nature of TUApps, combining advanced development techniques with potentially malicious behaviors and evasive financial practices.

## 7 Discussion

### 7.1 Implications and Potential Mitigations

Our comprehensive study of the underground mobile app ecosystem on Telegram reveals several critical implications and potential avenues for mitigation. These insights are valuable for platform regulators, app market operators, law enforcement agencies, and cybersecurity professionals.

Firstly, our analysis demonstrates that despite Telegram's efforts to warn users through scam tagging, current measures have failed to effectively regulate the presence of potentially harmful content on the platform. The encrypted communication and anonymity offered by Telegram, while valuable features, inadvertently provide a haven for illicit activities. This situation is particularly concerning given our finding that a quarter of TUApps exhibit maliciousness. To address this, we propose leveraging our findings to enhance Telegram's detection capabilities. By developing more sophisticated content analysis tools capable of identifying patterns in promotional messages, channel behaviors, and URL structures associated with underground apps, Telegram could significantly

improve its ability to detect potentially harmful content. This could be coupled with an in-app warning system that alerts users before they interact with suspicious content or click on external links, thereby reducing the risk of harm and enhancing overall platform safety and user trust.

Secondly, although iOS's restrictions on sideloading IPA files serve as a deterrent to the distribution of underground apps, the exploitation of iOS system features—such as TestFlight, Enterprise Signing, and Web Clips—for the distribution of TUApps demands attention from mobile system platforms. It is crucial for these platforms to further enhance regulatory measures to address these vulnerabilities or loopholes to prevent the misuse of their system features for distributing underground apps. Moreover, the discovery of underground apps temporarily available on the official Apple App Store underscores the importance of continually refining app review processes.

Thirdly, the prevalence of fourth-party payment processors in underground apps raises concerns about potential money laundering activities. The complex web of transactions involving multiple layers of intermediaries can obscure the origin and destination of funds, making it challenging for authorities to trace illicit financial flows. We suggest collaborating with financial institutions to develop more advanced transaction monitoring systems capable of identifying suspicious patterns associated with these layered payment structures.

Our findings not only shed light on the sophisticated nature of this ecosystem but also provide concrete pathways for enhancing detection, prevention, and regulatory measures. By implementing these recommendations, stakeholders can leverage the insights from our research to develop more effective strategies for combating the proliferation of underground apps.

### 7.2 Limitation

Our research encountered several limitations, primarily in data collection and the scope of our analysis. We employed multiple methods to source channels promoting underground apps, including bot-based searches, collection of top regular channels, and snowballing techniques. However, we observed significant disparities in the effectiveness of search bots between English and Chinese channels, with Chinese search bots yielding thousands of relevant channels in response to keyword queries, while English bots returned only dozens. Despite our intention to maintain a language-agnostic approach in data collection, utilizing both English and Chinese keywords (as detailed in § 3.2.1), our dataset predominantly captured underground apps and promotional activities targeting Chinese-speaking users. This skew likely reflects China's stringent regulatory environment, particularly concerning pornography and gambling, which may drive such activities to seek refuge in discreet platforms like Telegram. In contrast, regions with more relaxed regulations might allow similar content to be directly accessible via websites or social media, potentially reducing the demand for dedicated apps.

While our study offers valuable insights into the Chinese-language underground app ecosystem within Telegram, this predominance may limit the generalizability of our findings to other linguistic or geographic contexts. Nonetheless, we believe that the patterns and dynamics observed can serve as a foundation for understanding the challenges associated with underground app distribution on encrypted messaging platforms, especially in environments with strict content regulations. Researchers interested in underground apps targeting users of other languages could apply our approach, leveraging language-specific search bots and keywords to identify relevant channels and uncover underground app ecosystems within their linguistic contexts.

Furthermore, the reliance on public channels for data collection may overlook underground apps promoted through private channels and one-on-one communications, potentially underestimating the scale and diversity of the underground economy. Additionally, the dynamic nature of the underground app market, with rapidly emerging new apps and promotion strategies, may limit the long-term applicability of the findings. Our focus on Telegram means that the findings may not be

directly applicable to other platforms, such as Discord [28], which warrant separate investigation in future work.

## 8 Related Work

### 8.1 Telegram's Underground Ecosystem

Previous studies on Telegram's underground ecosystem primarily focuses on the conversation channels and their thematic content. Morgia et al. [53] delved into Telegram's dark side by examining channels marked as fake or scam, exploring the nature and operations of clones and conspiracy movements. Further extending their research [52], they amassed a collection of more than 100,000 Telegram channels to study the spread of illicit content, such as pornography, fraudulent credit card activities, violent content, hacking, and white supremacist ideologies. They also studied the dissemination of misinformation through Telegram channels, alongside proposing detection techniques for these deceptive channels [54]. Vincenzo et al. [51] undertook a thorough analysis of conspiracy theories circulating on Telegram, focusing on the identification of channels promoting such theories and their revenue-generating strategies. In contrast to previous research that focused on channels and thematic content, our study concentrates on mobile underground apps in Telegram.

### 8.2 Analysis of Underground Apps

Existing research on underground apps primarily revolves around their acquisition, identification, and analysis within the underground app ecosystem.

Regarding the acquisition of underground apps, the strategies generally involve either cooperation with authoritative institutions or analyzing promotional websites for these apps. Chen et al. [44], Hu et al. [49] and Hong et al. [48] succeeded in obtaining underground app samples through collaborating with authoritative institutions. Han et al. [47] and Gao et al. [46] acquired gambling-related domain names from authoritative institutions and crawled corresponding websites to obtain gambling apps. Through traffic analysis, Chen et al. [43] identified underground online portals and crawled underground apps from them.

To identify underground apps, Chen et al. [44] employed the similarity of UI transition graphs to detect underground apps. Zhao et al. [67] identified mask apps—malicious apps disguised as legitimate apps—on the Apple App Store, leveraging features including discrepancies between app descriptions and user reviews, app recommendation relationships, and code similarities.

Regarding the ecosystem analysis of underground apps, Han et al. [47] explored the ecosystem of illegal Android gambling apps by examining the relationship between gambling apps and advertising campaigns on upper-level pages. From a different angle, Gao et al. [46] focused on the ecosystem of illegal mobile gambling apps, analyzing their distribution methods, key characteristics, and identification methods. Hu et al. [49] studied the usage of underground Web Clips on the iOS platform from the perspectives of creators, distributors, and operators. Hong et al. [48] analyzed the chain of app gambling scams, starting from fraud alerts, and examined the development frameworks, permissions, compatibility, and network infrastructure of fraudulent apps. Chen et al. [45] revealed the structure of the underground app ecosystem and the operation of app builders by studying four participating entities – developers, proxies, operators, and harvesters – along with their workflows. Hu et al. [50] conducted an in-depth study of the ecosystem of fraudulent dating apps, analyzing the producers, publishers, and distribution networks, as well as identifying such apps. Our work complements these studies by specifically focusing on the underground app ecosystem in Telegram.

## 9 Conclusion

This study presents the first comprehensive analysis of the underground mobile app ecosystem on Telegram, illuminating a previously understudied facet of digital underground economies. By constructing and analyzing a novel dataset, we reveal a sophisticated network of underground apps nominally reaching 1% of Telegram's user base. Our research uncovered several critical findings, including the misuse of iOS features for app distribution, significant gaps in Telegram's content moderation capabilities, and the alarming prevalence of malicious behaviors in these underground apps. The widespread use of fourth-party payment platforms further complicates regulatory oversight. Our work underscores the urgent need for enhanced regulatory measures and detection mechanisms on encrypted platforms. Our findings provide valuable insights for platform regulators, app market operators, law enforcement agencies, and cybersecurity professionals in their efforts to combat underground app proliferation and protect users from associated risks.

## Acknowledgment

We sincerely thank our shepherd Prof. Andrea Marin (Università Ca' Foscari di Venezia) and all the anonymous reviewers for their valuable suggestions and comments to improve this paper. This work was supported by the Key R&D Program of Hubei Province (2023BAB017, 2023BAB079), the National NSF of China (grants No.62072046, 62302181), the Xiaomi Young Talents Program, the HUSTCSE-FiberHome Joint Research Center for Network Security, and the Beijing University of Posts and Telecommunications 2023 Education and Teaching Reform Project (2023YB37).

## References

[1] 2019. Cyber Security Asean. https://cybersecurityasean.com/news-press-releases/july-2019%E2%80%99s-most-wanted-malware-vulnerability-opendreambox-200-webadmin-plugin.
[2] 2020. Telegram's massive revenge porn problem has made these women's lives hell. https://mashable.com/article/nudes-revenge-porn-crime-telegram.
[3] 2021. A Threat Analysis of Sideloading. https://www.apple.com/privacy/docs/Building_a_Trusted_Ecosystem_for_Millions_of_Apps_A_Threat_Analysis_of_Sideloading.pdf.
[4] 2022. Telegram: A Cybercriminal Hotspot - Compromised Financial Accounts. https://cybersixgill.com/news/articles/telegram-a-cybercriminal-hotspot-compromised-financial-accounts.
[5] 2023. Is Telegram turning into a hub for cybercrime activities? https://10guards.com/en/articles/is-telegram-turning-into-a-hub-for-cybercrime-activities/.
[6] 2023. Money mules: Scam syndicates use Telegram to recruit young people for bank and Singpass accounts. https://www.straitstimes.com/singapore/courts-crime/money-mules-scam-syndicates-use-telegram-to-recruit-young-people-for-bank-and-singpass-accounts.
[7] 2023. Stories and 10 Years of Telegram. https://telegram.org/blog/stories.
[8] 2023. Telegram and OSINT Investigations: An Essential Platform in 2023. https://flare.io/learn/resources/blog/telegram-investigation/.
[9] 2023. Top Industries Significantly Impacted by Illicit Telegram Networks. https://thehackernews.com/2023/08/top-industries-significantly-impacted.html.
[10] 2024. androguard: Reverse engineering and pentesting for Android applications. https://github.com/androguard/androguard.
[11] 2024. AppBrain. https://www.appbrain.com/stats/libraries/development-tools.
[12] 2024. Apple, Apple Developer Enterprise Program. https://developer.apple.com/programs/enterprise.
[13] 2024. Apple, iTunes search API. https://affiliate.itunes.apple.com/resources/documentation/itunes-store-web-service-search-api/.
[14] 2024. Apple, Web Clips MDM payload settings for Apple devices. https://support.apple.com/en-mn/guide/deployment/depbc7c7808/1/web/1.0.
[15] 2024. Build for any screen. https://flutter.dev.
[16] 2024. Burp Suite: Application Security Testing Software. https://portswigger.net/burp.
[17] 2024. Google, Google play store. https://play.google.com/store/apps.
[18] 2024. Huawei, Huawei app store. https://consumer.huawei.com/cn/support/appgallery/.

[19] 2024. IPinfo, Official Python Library for IPinfo API (IP geolocation and other types of IP data). https://github.com/ipinfo/python.
[20] 2024. Kaspersky Threats — AdWare.Win32.SoftPulse.gokp. https://threats.kaspersky.com/en/threat/AdWare.Win32.SoftPulse.gokp/.
[21] 2024. Kaspersky Threats — Boogr. https://threats.kaspersky.com/en/threat/Trojan.AndroidOS.Boogr/.
[22] 2024. Kaspersky Threats — Mobtes. https://threats.kaspersky.com/en/threat/Trojan.AndroidOS.Mobtes/.
[23] 2024. Kaspersky Threats — Trojan.AndroidOS.Piom.bbdw. https://threats.kaspersky.com/en/threat/Trojan.AndroidOS.Piom.bbdw/.
[24] 2024. macaca: Automation solution for multi-platform. https://github.com/alibaba/macaca.
[25] 2024. PaddleOCR: Awesome multilingual OCR toolkits based on PaddlePaddle. https://github.com/PaddlePaddle/PaddleOCR.
[26] 2024. plistlib: Generate and parse Apple .plist files. https://docs.python.org/3/library/plistlib.html.
[27] 2024. requests, A simple, yet elegant, HTTP library. https://pypi.org/project/requests/.
[28] 2024. The Rise of Cybercrime on Telegram and Discord and the Need for Continuous Monitoring. https://www.cloudsek.com/blog/the-rise-of-cybercrime-on-telegram-and-discord-and-the-need-for-continuous-monitoring.
[29] 2024. Selenium automates browsers. That's it! https://www.selenium.dev/.
[30] 2024. Telegram channels and groups catalog | TGStat. https://tgstat.com/.
[31] 2024. Telegram Search Engine Send keywords to search for groups and channels! This advertising space is available for sponsorship. https://t.me/TGbaiduCN.
[32] 2024. Telegram, Telegram API. https://core.telegram.org.
[33] 2024. TelegramChannels: Discover The Best Telegram Channels. https://telegramchannels.me/.
[34] 2024. Telethon: Pure Python 3 MTProto API Telegram client library, for bots too! https://github.com/LonamiWebs/Telethon.
[35] 2024. TestFlight: Beta Testing made simple with TestFlight. https://developer.apple.com/cn/testflight/.
[36] 2024. VirusTotal: Analyse suspicious files, domains, IPs and URLs to detect malware and other breaches, automatically share them with the security community. https://www.virustotal.com/.
[37] 2024. whois. https://github.com/richardpenman/whois.
[38] 2024. Xiaomi, Xiaomi app store. https://app.mi.com/.
[39] 2024. ZXing ("Zebra Crossing") barcode scanning library for Java, Android. https://github.com/zxing/zxing.
[40] Steven Arzt, Siegfried Rasthofer, Christian Fritz, Eric Bodden, Alexandre Bartel, Jacques Klein, Yves Le Traon, Damien Octeau, and Patrick McDaniel. 2014. Flowdroid: Precise context, flow, field, object-sensitive and lifecycle-aware taint analysis for android apps. *ACM sigplan notices* 49, 6 (2014), 259–269.
[41] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment* 2008, 10 (2008), P10008.
[42] Tianqin Cai, Zhao Zhang, and Ping Yang. 2020. Fastbot: A Multi-Agent Model-Based Test Generation System Beijing Bytedance Network Technology Co., Ltd.. In *Proceedings of the IEEE/ACM 1st International Conference on Automation of Software Test*. 93–96.
[43] Pei Chen, Gang Hong, Mengying Wu, Jinsong Chen, Haixin Duan, and Min Yang. 2024. An underground industry application collection method based on flow analysis. *Journal of Software* 35, 8 (2024), 0–0.
[44] Zhuo Chen, Jie Liu, Yubo Hu, Lei Wu, Yajin Zhou, Yiling He, Xianhao Liao, Ke Wang, Jinku Li, and Zhan Qin. 2023. Deuedroid: Detecting underground economy apps based on utg similarity. In *Proceedings of the 32nd ACM SIGSOFT International Symposium on Software Testing and Analysis*. 223–235.
[45] Zhuo Chen, Lei Wu, Jing Cheng, Yubo Hu, Yajin Zhou, Zhushou Tang, Yexuan Chen, Jinku Li, and Kui Ren. 2021. Lifting The Grey Curtain: A First Look at the Ecosystem of CULPRITWARE. *arXiv preprint arXiv:2106.05756* (2021).
[46] Yuhao Gao, Haoyu Wang, Li Li, Xiapu Luo, Guoai Xu, and Xuanzhe Liu. 2021. Demystifying illegal mobile gambling apps. In *Proceedings of the Web Conference 2021*. 1447–1458.
[47] Yadi Han, Shanshan Wang, Yiwen Li, Xueyang Cao, Limei Huang, and Zhenxiang Chen. 2023. Measurement of Illegal Android Gambling App Ecosystem From Joint Promotion Perspective. In *2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 1–11.
[48] Geng Hong, Zhemin Yang, Sen Yang, Xiaojing Liaoy, Xiaolin Du, Min Yang, and Haixin Duan. 2022. Analyzing ground-truth data of mobile gambling scams. In *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2176–2193.
[49] Qinyu Hu, Songyang Wu, Wenqi Sun, Zhushou Tang, Chaofan Chen, Zhiguo Ding, and Xiaomei Zhang. 2022. Measurement of the Usage of Web Clips in Underground Economy. *arXiv preprint arXiv:2209.03319* (2022).
[50] Yangyu Hu, Haoyu Wang, Yajin Zhou, Yao Guo, Li Li, Bingxuan Luo, and Fangren Xu. 2019. Dating with scambots: Understanding the ecosystem of fraudulent dating applications. *IEEE Transactions on Dependable and Secure Computing* 18, 3 (2019), 1033–1050.

[51] Vincenzo Imperati, Massimo La Morgia, Alessandro Mei, Alberto Maria Mongardini, and Francesco Sassi. 2023. The Conspiracy Money Machine: Uncovering Telegram's Conspiracy Channels and their Profit Model. *arXiv preprint arXiv:2310.15977* (2023).

[52] Massimo La Morgia, Alessandro Mei, and Alberto Maria Mongardini. 2023. TGDataset: a Collection of Over One Hundred Thousand Telegram Channels. *arXiv preprint arXiv:2303.05345* (2023).

[53] Massimo La Morgia, Alessandro Mei, Alberto Maria Mongardini, and Jie Wu. 2021. Uncovering the dark side of Telegram: Fakes, clones, scams, and conspiracy movements. *arXiv preprint arXiv:2111.13530* (2021).

[54] Massimo La Morgia, Alessandro Mei, Alberto Maria Mongardini, and Jie Wu. 2023. It'sa Trap! Detection and Analysis of Fake Channels on Telegram. In *2023 IEEE International Conference on Web Services (ICWS)*. IEEE, 97–104.

[55] Li Li, Tegawendé F Bissyandé, Hao-Yu Wang, and Jacques Klein. 2019. On identifying and explaining similarities in android apps. *Journal of Computer Science and Technology* 34 (2019), 437–455.

[56] Yuanchun Li, Ziyue Yang, Yao Guo, and Xiangqun Chen. 2017. Droidbot: a lightweight ui-guided test input generator for android. In *2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C)*. IEEE, 23–26.

[57] Ziang Ma, Haoyu Wang, Yao Guo, and Xiangqun Chen. 2016. Libradar: Fast and accurate detection of third-party libraries in android apps. In *Proceedings of the 38th international conference on software engineering companion*. 653–656.

[58] Volodymyr Miz, Joëlle Hanna, Nicolas Aspert, Benjamin Ricaud, and Pierre Vandergheynst. 2020. What is trending on wikipedia? capturing trends and language biases across wikipedia editions. In *Companion proceedings of the Web conference 2020*. 794–801.

[59] Behnaz Moradi-Jamei, Brandon L Kramer, J Bayoán Santiago Calderón, and Gizem Korkmaz. 2021. Community formation and detection on GitHub collaboration networks. In *Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*. 244–251.

[60] Feargus Pendlebury, Fabio Pierazzi, Roberto Jordaney, Johannes Kinder, and Lorenzo Cavallaro. 2019. {TESSERACT}: Eliminating experimental bias in malware classification across space and time. In *28th {USENIX} Security Symposium ({USENIX} Security 19)*. 729–746.

[61] Silvia Sebastián and Juan Caballero. 2020. Avclass2: Massive malware tag extraction from av labels. In *Proceedings of the 36th Annual Computer Security Applications Conference*. 42–53.

[62] Yun Shen, Pierre-Antoine Vervier, and Gianluca Stringhini. 2022. A large-scale temporal measurement of android malicious apps: Persistence, migration, and lessons learned. In *31st USENIX Security Symposium (USENIX Security 22)*. 1167–1184.

[63] Kseniia Tikhomirova and Ilya Makarov. 2021. Community detection based on the nodes role in a network: The telegram platform case. In *Analysis of Images, Social Networks and Texts: 9th International Conference, AIST 2020, Skolkovo, Moscow, Russia, October 15–16, 2020, Revised Selected Papers 9*. Springer, 294–302.

[64] Milind Tiwari, Jamie Ferrill, and Douglas MC Allan. 2024. Trade-based money laundering: a systematic literature review. *Journal of Accounting Literature* (2024).

[65] Jingjing Wang, Liu Wang, Feng Dong, and Haoyu Wang. 2023. Re-measuring the label dynamics of online anti-malware engines from millions of samples. In *Proceedings of the 2023 ACM on Internet Measurement Conference*. 253–267.

[66] Bryan White, Aniket Mahanti, and Kalpdrum Passi. 2022. Characterizing the OpenSea NFT marketplace. In *Companion Proceedings of the Web Conference 2022*. 488–496.

[67] Yijun Zhao, Lingjing Yu, Yong Sun, Qingyun Liu, and Bo Luo. 2024. No Source Code? No Problem! Demystifying and Detecting Mask Apps in iOS. In *Proceedings of the 32nd IEEE/ACM International Conference on Program Comprehension*. 358–369.

[68] Yury Zhauniarovich, Olga Gadyatskaya, Bruno Crispo, Francesco La Spina, and Ermanno Moser. 2014. FSquaDRA: Fast detection of repackaged applications. In *Data and Applications Security and Privacy XXVIII: 28th Annual IFIP WG 11.3 Working Conference, DBSec 2014, Vienna, Austria, July 14-16, 2014. Proceedings 28*. Springer, 130–145.

[69] Shuofei Zhu, Jianjun Shi, Limin Yang, Boqin Qin, Ziyi Zhang, Linhai Song, and Gang Wang. 2020. Measuring and modeling the label dynamics of online {Anti-Malware} engines. In *29th USENIX Security Symposium (USENIX Security 20)*. 2361–2378.

# A Appendix

## Ethics and Availability

Our research involved the collection and analysis of a large-scale dataset from public Telegram channels. While these channels are publicly accessible, we acknowledge that the users participating in these channels may not have explicitly consented to the collection and use of their data for research purposes. To address these concerns, we have taken measures to ensure the responsible handling of the collected data:

- We have implemented secure data storage practices. The data is stored on encrypted drives with restricted access limited to authorized researchers directly involved in this study.
- In reporting our findings, we have presented the results in an aggregated and anonymized manner, avoiding the inclusion of personally identifiable information.

Regarding availability, our code and dataset are available at https://github.com/security-pride/TUApps. Specifically:

- The underground apps discovered in our research will be made publicly available, as they are essential for understanding and combating illicit activities within the mobile app ecosystem.
- To protect user privacy, the dataset containing user messages will be available privately upon request from researchers who agree to adhere to strict ethical guidelines and maintain data confidentiality.
- The code used for data collection and analysis will also be available privately upon request to prevent potential misuse and ensure that future research follows appropriate ethical standards.

We believe that this approach balances the need for transparency and reproducibility in research with the importance of protecting user privacy and preventing unethical data collection practices.

Received August 2024; revised September 2024; accepted October 2024