Demystifying Illegal Mobile Gambling Apps

Yuhao Gao
Beijing University of Posts and Telecommunications
Beijing, China

Haoyu Wang*
Beijing University of Posts and Telecommunications
Beijing, China

Li Li
Monash University
Melbourne, Australia

Xiapu Luo
The Hong Kong Polytechnic University
Hong Kong, China

Guoai Xu*
Beijing University of Posts and Telecommunications
Beijing, China

Xuanzhe Liu
Peking University
Beijing, China

ABSTRACT

Mobile gambling app, as a new type of online gambling service emerging in the mobile era, has become one of the most popular and lucrative underground businesses in the mobile app ecosystem. Since its born, mobile gambling app has received strict regulations from both government authorities and app markets. However, to the best of our knowledge, mobile gambling apps have not been investigated by our research community. In this paper, we take the first step to fill the void. Specifically, we first perform a 5-month dataset collection process to harvest illegal gambling apps in China, where mobile gambling apps are outlawed. We have collected 3,366 unique gambling apps with 5,344 different versions. We then characterize the gambling apps from various perspectives including app distribution channels, network infrastructure, malicious behaviors, abused third-party and payment services. Our work has revealed a number of covert distribution channels, the unique characteristics of gambling apps, and the abused fourth-party payment services. At last, we further propose a "guilt-by-association" expansion method to identify new suspicious gambling services, which help us further identify over 140K suspicious gambling domains and over 57K gambling app candidates. Our study demonstrates the urgency for detecting and regulating illegal gambling apps.

CCS CONCEPTS

• **Security and privacy → Software and application security**.

ACM Reference Format:
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 (WWW '21), April 19–23, 2021, Ljubljana, Slovenia.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3442381.3449932

1 INTRODUCTION

Mobile app ecosystem has seen explosive growth in recent years [43]. A large number of studies in our community have been focused on analyzing various kinds of issues (e.g., security [12], privacy [26], and fraudulent behaviors [8, 18]) in mobile apps, towards building a better mobile app ecosystem [40, 48].

Mobile gambling apps, derived from online gambling services, have become one of the most popular and lucrative underground businesses in the mobile app ecosystem. Due to the special nature of gambling apps, most countries and regions around the world have strict legal regulations on online gambling activities. Thus, gambling apps are restricted in most mobile app markets. Currently, in Google Play, gambling apps are only permitted in the UK, Ireland, and France [35], with a series of strict conditions, e.g., the developer must have a valid gambling license for each country in which the app is distributed. This, however, is unacceptable to many operators of illegal online gambling activities. For greater benefits, these illegal gambling operators often use various methods to evade supervision and even violate the law to obtain economic benefits. For these reasons, mobile gambling apps are very different from traditional consumer-oriented mobile apps in app markets like Google Play and iOS app store.

Although illegal gambling apps have become the primary source of revenue for organized crime groups, our research community still lacks the understanding of the illegal gambling app ecosystem, especially when considering that operators have employed various kinds of techniques to evade governments' regulation. It is unknown to us the status quo of illegal gambling apps. *Whether the illegal gambling apps are prevalent in the wild? How do they spread and reach to mobile users? What are their main characteristics? How can we identify the campaigns behind them?*

This Work. In this paper, we make the first systematic study of the illegal gambling app ecosystem. Specifically, we focus on illegal gambling apps that target Chinese mobile users, as online gambling has been under strict legislative control and mobile gambling apps outlawed in China. The first challenge in this study is to harvest a set of illegal gambling apps. As illegal gambling apps are usually distributed in covert channels beyond app markets (i.e., we did not find any gambling app on the official Google Play store and some popular Chinese alternative markets by keywords searching), it is non-trivial for us to collect them. However, we observe that the illegal gambling apps have a strong correlation with illegal online gambling websites, i.e., the websites usually provide download links of the corresponding gambling apps. To this end, we first make efforts to analyze over 40 million domains by cooperating with a major Internet Service Provider (ISP) in China, and take advantage of a machine-learning based approach to identify illegal

*Haoyu Wang and Guoai Xu are co-corresponding authors.

WWW '21, April 19–23, 2021, Ljubljana, Slovenia

ACM ISBN 978-1-4503-8312-7/21/04.
https://doi.org/10.1145/3442381.3449932

online gambling websites. After that, we use a semi-automated approach to download illegal gambling apps (see **Section 3.2**). Our dataset collection lasts five months, and eventually, we are able to collect 3,366 unique gambling apps, with in total 5,344 different versions. We further characterize the illegal gambling apps from their distribution channels (see **Section 4**), network infrastructure (see **Section 5.1**), malicious behaviors (see **Section 5.2**), third-party libraries (see **Section 5.3**) and payment services (see **Section 5.4**). At last, we investigate the relations among the illegal gambling apps, and identify the illegal campaigns that create and operate the gambling apps (see **Section 6**). The results give a first impression on the landscape of the illegal gambling apps, revealing some unexpected and interesting observations:

- **Illegal gambling apps are prevalent in the wild.** Although gambling apps are outlawed in China, we still observe a large number of illegal gambling apps target Chinese users. By investigating the domain-app relations, we identify over 3,000 unique gambling apps with over 5,000 different versions.
- **Covert app distribution channels are favored by gambling apps to evade supervision.** Gambling apps are mainly distributed using covert channels beyond app markets. Besides gambling websites, we have identified a number of specific distribution channels for releasing gambling apps.
- **The network infrastructure analysis suggests the ineffectiveness of gambling app supervision in China.** Using automated testing, we have identified over 11K server domains that are highly related to the functionalities of gambling apps. Surprisingly, roughly half of the gambling servers are located in mainland China. Illegal gambling apps have the tendency to hide their operators' identities, i.e., almost all the registrant information is private and CNAMEs (Canonical Name Record)[1] are favored by them.
- **A number of third-party services are abused by illegal gambling apps.** Push notification services are abused to distribute gambling related contents, and self-upgrade services are abused to distribute new versions of gambling apps. Besides, we have identified that fourth-party payment services are widely used in gambling apps, e.g., to hide the identities of money recipients.
- **The illegal gambling apps are operated in groups.** Using code-level and signature-level clustering analysis, we have identified 193 gambling campaigns behind the gambling apps. Then, we further propose a "guilt-by-association" expansion method to identify new suspicious gambling services and apps, which enables us to identify over 140K new suspicious gambling servers and thousands of gambling apps.

Our results motivate the need for more research efforts to illuminate the widely unexplored illegal gambling app ecosystem. We hope our efforts can bring awareness of regulators, practitioners, and fellow researchers about this problem and subsequently attract more advanced studies towards better characterizing illegal mobile gambling apps. We have released our dataset to the research community to boost further research on mobile gambling apps at:

https://mobile-app-research.github.io/

[1] A CNAME record is a type of resource record in the DNS that maps one domain name (an alias) to another (the canonical name) [52].

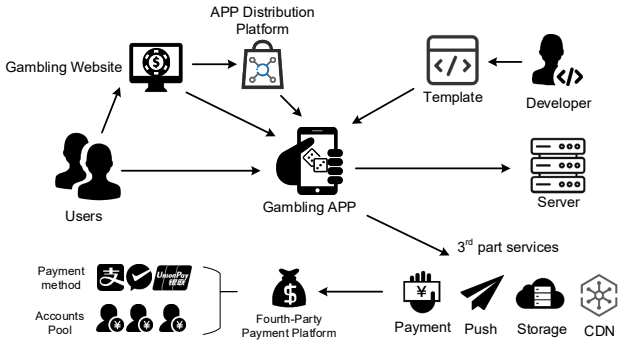


Figure 1: An Overview of the Gambling App Ecosystem.

2 PRELIMINARY STUDY OF ILLEGAL GAMBLING APPS

We first conduct a preliminary study (by searching gambling apps online and manually looking into their working processes) towards understanding the major players and the working protocol of illegal gambling apps. Based on our manual observation, we summarize the working process of illegal gambling apps in Figure 1. To evade detection and regulation, ***gambling apps*** usually employ covert channels for app distribution, beyond the traditional app markets. The ***illegal gambling websites*** usually host corresponding gambling apps operated by the same company, making it easier for users to search and download illegal gambling apps. ***Gambling app developers*** usually create a number of gambling apps that are connected to different web servers, as the illegal gambling servers would be blocked from time to time. Besides, a number of ***third-party services***, including ***payment services*** would be exploited by illegal gambling apps for easing the creation of apps.

Next, we introduce the major components in the ecosystem, which will further be characterized in the following sections.

2.1 Gambling Websites and Gambling Apps

Gambling websites have gained tremendous popularity since the born of online gambling twenty years ago [54]. When it comes to the mobile app era, online gambling has been evolved into a new form, i.e., *mobile gambling app*, which provides great user experience on smartphones. *In general, an illegal campaign can operate both gambling websites and gambling apps.* As gambling apps are strictly regulated by the government, most legitimate app markets do not provide distribution services for gambling apps. Subsequently, gambling app developers have to find new channels to distribute their apps. As discovered by previous works [56], illegal gambling websites may exploit blackhat SEO (Search Engine Optimization) for promoting their websites, aiming to attract more players. Under this campaign, existing gambling websites could also provide download links to the corresponding gambling apps so as to reach potential mobile users. These *website-app* relationships could then be leveraged to identify mobile gambling apps.

2.2 App Distribution Channels

In this paper, we call the channels (e.g., websites) that directly provide the download service of gambling apps as the *app distribution*

channels. Note that, although we mentioned that most gambling websites would provide download links of gambling apps for promotion, many of them do not provide the download links directly. Usually, users would be redirected to a specially designed site to provide such app download services. The distribution channels will be studied in Section 4.

2.3 App Server/Network Infrastructure

Due to the strict restriction of online gambling, gambling apps usually utilize several methods to protect their services from being banned. We found that their servers are not remain unchanged. Instead, they usually register a number of server addresses, and the available ones will be selected during app initialization. Moreover, to protect the real server addresses, a large number of CNAMEs are used. The network infrastructure of illegal gambling apps will be detailed in Section 5.1.

2.4 Third-Party Services

Third-party services have become an indispensable part of app development [31, 41]. Illegal gambling apps also take advantage of third-party services to implement their functionalities. For example, content distribution networks (CDNs), network storage services, push notification services, mobile advertising SDKs, and other kinds of third-party services have been found to be abused by illegal gambling apps. We will further characterize the abused third-party services in Section 5.3.

2.5 Payment Services

The sole purpose of illegal gambling apps is to make a profit. However, the explicit money flow can provide clues for governments and police authorities to trace and even arrest illegal gambling app operators. Therefore, unlike other apps, *payment anonymity* is vital to illegal gambling apps. Unlike payment methods such as via credit cards and online banking in western countries, gambling app operators in China prefer to abuse third-party online payment channels like *Alipay* and *WeChat pay*. Moreover, to further hide their payment behaviors, the so-called *fourth-party payment services* that provide thousands of virtual merchants are "innovatively" used by illegal gambling apps, which will be presented in Section 5.4.

2.6 Creators and Illicit Campaigns

Previous work [20, 61] suggested that malicious developers usually release malware in the form of app repackaging or code reuse. Illegal gambling apps come with no difference. In general, an illicit gambling campaign can release a number of similar gambling apps (based on the same template) to different release channels, in order to gain more potential users. Once the gambling apps are banned, the creators can soon release similar apps as the replacements. We will study the illicit gambling campaigns in Section 6.

3 STUDY DESIGN

3.1 Research Questions

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

RQ1 **To what extent are gambling apps distributed in the wild, and how are they penetrated?** Considering that mobile gambling apps are restricted by both app markets and the government, it is thus interesting to investigate the distribution channels of them.

RQ2 **What are the characteristics of gambling apps?** Considering that illegal gambling services are prohibited in China and usually involve tremendous illicit profits, it is interesting to investigate their characteristics, including their deployed network infrastructures, leveraged third-party services, and adopted online payment channels. It can help better understand how gambling apps operate.

RQ3 **Can we infer the underground campaigns behind illegal gambling apps so as to uncover more of such apps?** Identifying illegal gambling services is a long-term need. It is thus necessary to investigate the illegal campaigns that create and operate the gambling apps, and identify new gambling services if possible.

3.2 Dataset Collection

Our dataset collection contains two steps: 1) identifying illegal online gambling websites, and 2) collecting illegal gambling apps.

3.2.1 Identifying illegal gambling websites. Based on our manual exploration, we observe that illegal gambling apps have a strong correlation with illegal online gambling websites, i.e., the websites usually provide download links of the corresponding gambling apps. Hence, we propose to identify gambling websites first and then to harvest the correlated gambling apps. In order to obtain as many gambling websites as possible, we cooperate with a major ISP in China to obtain the DNS request data of all the users in a major city from August 2019 to January 2020, which contains over 40 million unique domains in total. Note that the dataset collection has no ethical issues, as we did not obtain any raw data related to personal information from the ISP. The dataset collection process lasts five months. We use the following method to retrieve web content from identified gambling websites every day.

To identify gambling domains, we first filtered irrelevant domains using the ICP license data and Alexa top 100,000 sites [3]. Then, we crawl the web contents of the remaining domains. To speed up the detection process, we have manually summarized a list of gambling keywords (over 100) in both Chinese and English. We only keep the domains that have embedded at least one keyword in their web content. After that, we borrow the idea of an existing method [56] to identify illegal gambling websites. Specifically, each web page is parsed to extract all the shown texts. After removing stop words, we represent each web page as a feature vector containing a number of keywords. A SVM classifier is then used to identify illegal gambling websites. This approach has been demonstrated to be effective in the previous study, with an accuracy of over 99.99%. With this approach, we could identify 6,105 online gambling websites, for which we have manually confirmed that all of them are indeed gambling websites.

3.2.2 Collecting illegal gambling apps. Then, we further analyze these websites to download their corresponding gambling apps. We adopt a semi-automated approach here. In general, the gambling

Table 1: Overview of the gambling app dataset.

#Item	Count
Illegal Gambling sites	6,105
Distribution (download) channels	1,415
Gambling apps	3,366
Gambling apks (with different versions)	5,344
Developer certificates	1,136

websites provide a number of ways to download apps, including QR code (i.e., downloading apps through scanning the QR Code), and indirect download links, etc. Thus, we first analyze the contents to identify and click any potential download links automatically. For those websites that we cannot download apps automatically, we further visit them and download apps manually, if any. Two authors of this paper have spent a significant amount of time to download mobile apps on the identified gambling websites. We then followed a semi-automated process (download, launch, and explore the apps) to select true gambling apps. All the downloaded apps are indeed gambling apps, as we have recorded the screenshots of the app runtime UIs, which can be used for manual confirmation. We observed that most of the gambling apps contain multiple types of games including sports gambling, casino, poker, and lottery, etc, which will be discussed in Section 7. Note that we only focus on Android apps, without considering iOS apps, although a few websites provide ways to installing iOS gambling apps.

3.3 Dataset Overview

Table 1 presents an overview of our dataset. Note that, as our dataset collection process lasts five months, we have identified gambling apps with multiple versions from different gambling websites[2]. At last, we have downloaded over 3,366 unique illegal gambling apps (with 5,344 different versions) from the 6,105 gambling websites. These apps are signed by 1,136 different developer signatures. These apps are downloaded from 1,415 unique websites that provide the download services, which will be detailed in Section 4.

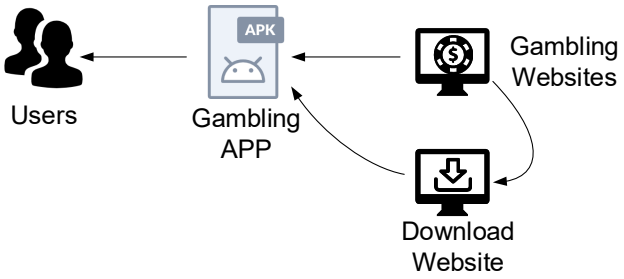


Figure 2: An overview of gambling app distribution.

4 DISTRIBUTING GAMBLING APPS

In this section, we characterize the distribution of gambling apps by analyzing the associated gambling websites and download links.

[2]Here, we consider the apps with the same package name and same developer signature as the same app with different versions.

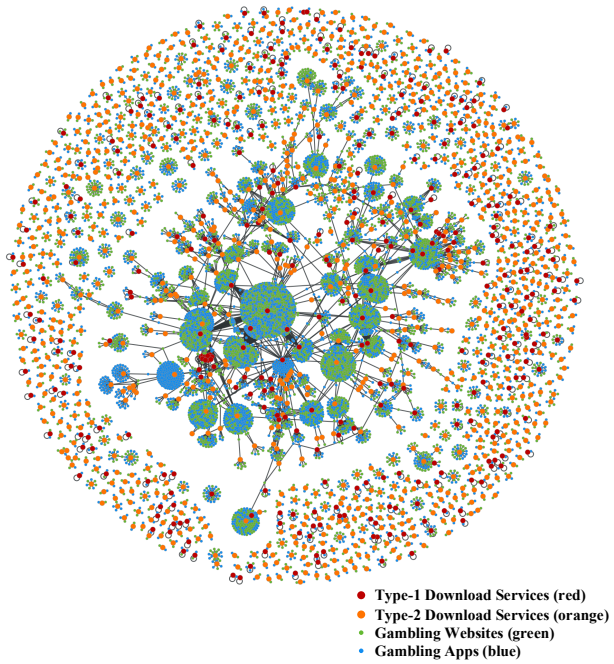


Figure 3: The relationship among gambling apps, gambling websites and app distribution channels.

4.1 Overview of Distribution Channels

Different from ordinary apps that are distributed via app markets, gambling apps usually exploit hidden distribution channels to evade supervision. Indeed, we have attempted to search all the identified 3,366 apps on Androzoo [25], a large Android app corpus containing over 10 million apps crawled from Google Play and a number of third-party app markets. As expected, none of them was found.

As shown in Figure 2, gambling websites usually redirect the download requests to some hidden download websites. During our exploration, these gambling apps were downloaded from 1,415 distribution sites (counted by the unique domain name). Considering that we extracted the download links from over 6,000 gambling websites, it is obvious that some distribution websites provide download services for more than one gambling app.

4.2 Relation Analysis

We further analyze the relationship among gambling websites, gambling apps, and their distribution channels. We notice that the distribution channels can be generally classified into two types. For the first type, 345 download services have identical domain names with the gambling websites (**Type-1**). In this case, it is explicit that the gambling websites and download services are operated by the same people. For the second type, most of the download services have different domain names with gambling websites (**Type-2**). For example, as shown in Table 2, some distribution sites have been linked by hundreds of gambling websites, providing download services for all of them.

Table 2: Top 10 distribution services of gambling apps.

Domain	Websites	Apks	Unique Apks	Developers
yb9.me	954	839	650	8
asd2159.ucc-bundle.broadcast-large.com	223	124	112	7
yt9.me	200	191	182	1
yk7.me	150	125	104	3
app3.ppbk9.com	136	91	60	2
www.tginapp.com	128	100	97	6
www.tbk-app.com	122	104	102	50
app.gg88668.com	109	90	26	1
wrdphone.xuliehaowang.com	96	86	13	7
220.248.178.250:8081	90	121	48	24

Relation Graph. We create the relation graph to characterize their relationships. As shown in Figure 3, we use nodes with different colors to represent gambling apps (blue), gambling websites (green), **type-1** download services (red), and **type-2** download services (orange). There are two kinds of edges in the graph. 1) The edge between a gambling website and a type-2 download service indicates that the gambling website uses the download service to distribute gambling apps. 2) The edge between a download service (**type-1** or **type-2**) and a gambling app indicates that the gambling app was downloaded from the service. It is interesting to observe that, although many download services are used for distributing specific gambling apps (the peripheral nodes in Figure 3), some nodes are quite dense (in the center of Figure 3), suggesting that they are highly related.

Table 2 further shows the top-10 distribution channels of gambling apps in our dataset. Surprisingly, the top 10 domains have provided download services for 1,394 (26%) apps. For example, the most representative one, namely `yb9.me`, involves 954 gambling websites and among which we could collect 839 APKs, accounting for 650 distinct APKs. Furthermore, we have noticed clues that some distribution services might be operated by gambling campaigns. Indeed, some of such services only provide download links for apps signed by some specific signatures. For example, although we have collected 650 unique apps from `yb9.me`, they were signed by only eight unique developer signatures. As another example, the domain `yt9.me` provided download services for 182 apps signed by the same developer signature.

Abusing third-party app distribution channels. To further understand these websites that provide download services, we take advantage of a headless browser to crawl the homepages of these download websites (if available) for further manual verification. By manually analyzing these results, we found that some download services (52 in total) are common services that can be used by any apps, including many benign apps. They are not directly related to gambling websites in business, which are, however, abused by gambling apps as the distribution channels. For example, `xmwvip.vip` is a typical developer service to facilitate the process of app releases, including app signing and app downloading. Any developer can use it to sign their apps and use their download service. We have observed 21 gambling apps downloaded from this site. These 52 verified third-party app distribution services are abused to distribute 449 (13.3%) illegal gambling apps in total. For the remaining distribution websites, we cannot get the homepages or other useful information. According to their relationships between gambling websites and apps, we believe they are quite possible to be operated by the same gambling campaign.

Answer to RQ1: *The illegal gambling apps are prevalent in the wild. We have identified over 3,000 different gambling apps (with over 6,000 versions) by exploiting the gambling domain-to-app relations. These apps are distributed using covert channels beyond app markets to evading detection and supervision. Their distribution channels suggest that a number of gambling apps and websites are operated in groups. We further identify a number of legitimate third-party app distribution services that have been abused by gambling apps.*

5 CHARACTERIZING GAMBLING APPS

In this section, we present the characterization of gambling apps, including their *network infrastructure*, *malicious behaviors*, and *abused third-party services*.

5.1 Network Infrastructure

5.1.1 Connected Server Addresses. To analyze the network-level behaviors of gambling apps, we seek to collect the network traffic of gambling apps at runtime.

Method. To be specific, we take advantage of DroidBot [58], a widely used Android app automated input generation tool for dynamically exercising the gambling apps. As previous work [39] suggested that many mobile apps have embedded checking code to hide their sensitive behaviors when the app is being experimented on emulator environments, we hence run all the collected gambling apps on real phones, i.e., Google Pixel smartphones. Note that, we do not attempt to traverse all the UI activities of the gambling apps. Instead, we only want to collect its contacted server addresses. Thus, for each gambling app, we limit its testing time to only 1 minute. Our preliminary study suggests that it is enough to extract the contact servers of the gambling apps. Furthermore, during runtime, we take advantage of `tcpdump` to record all the network traffic and `netstat` to distinguish gambling apps traffic from other apps, including the connected servers (domains).

Filtering Common Server Addresses. As third-party libraries are widely used in Android apps [42], a number of third-party service addresses will be identified during our exploration. To accurately pinpoint the gambling server addresses, we should filter those common server addresses first. To this end, we decide to harvest domain names that belong to third-party libraries from a large number of ordinary apps. The key idea is that, *the common server addresses would be connected by a number of apps embedded in the corresponding third-party libraries*. Thus, we can leverage the domain clustering results in large-scale Android apps to identify the common one. Specifically, we have crawled 100,000 Android apps from Google Play and Tencent Myapp (one of the largest Chinese app markets). We follow the same dynamic exploration process to collect the traffic generated by these apps. Then, by clustering the domain names, we mark the domains that used in at least 50 apps and at least 20 developers as the common server addresses. Note

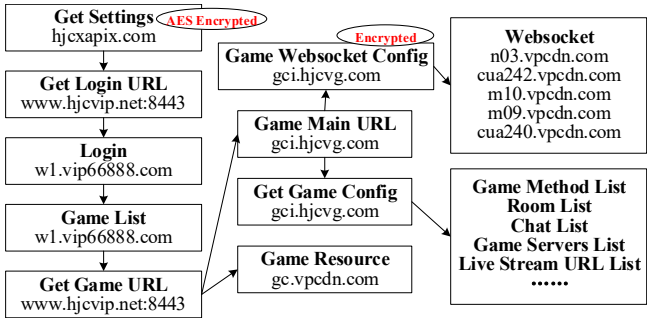


Figure 4: An example of servers used in gambling apps.

that, the thresholds are empirically set based on our initial observation and previous third-party library detection studies [24, 31, 42]. We further utilize the Alexa top 10,000 domains [3] to filter the common domains collected from the network traffic of gambling apps. After the filtering process, we have identified 11,320 domains that are highly suspicious to be correlated with gambling services. We remind the readers that all the domains are initially triggered by gambling apps.

Result Analysis. We found that *gambling apps usually connect to a number of different server addresses (8 on average), and their communication process is more complicated than ordinary apps.* Figure 4 shows an example. The gambling app *com.hmobile.core* would first connect to `hjcxapix.com` during app initialization and get a list of app settings. When the user wants to login, it will then request the real login function URL `w1.vip66888.com` from `www.hjcvip.net:844`, and send account information to it. After log in successfully, a list of gambling games will be returned from `w1.vip66888.com`. Then the user may choose one game to play, and the app will request the main game URL `gci.hjcvg.com` and its resource loading URL `gc.vpcdn.com` from `www.hjcvip.net:844`. Note that, in this case, the media resource will be loaded from the resource URL, while other configurations, including web socket servers, will be loaded from the game's main URL. Finally, the user can start the gambling game. From this example, we can observe that, a typical gambling app will use a number of different domain names for data communication. Different domain names are used in various kinds of app functionalities, e.g., initialization, data processing, resource loading, and gambling game main services, etc.

5.1.2 Domain Analysis. We further characterize the network infrastructure used by gambling apps.

Top Level Domains. Table 3 shows the most popular TLDs of the gambling domain names used by gambling apps. Although most gambling apps tend to rely on traditional gTLDs (generic top-level domains) such as `.com` and `.net`, there are also a number of gambling apps that choose new gTLDs such as `.xyz`, `.vip`, `.app` and special ccTLDs (country code top-level domains) like `.me`, `.cc`. The gTLDs and ccTLDs have the advantages of low price and loose regulation [10], which are favored by the underground economy, including illicit online gambling. Note that registering a domain under TLD `.cn` requires an ICP (Internet Content Provider) license in China. However, it is interesting to see that, `.cn` is the third most popular TLD in our dataset, suggesting the ineffectiveness of ICP regulation. This finding is in line with the previous study [56] on gambling websites.

Table 3: The distribution of TLDs used in gambling apps.

TLD	Category	TLD Manager	Count	Percentage
.com	gTLD	VeriSign Global Registry Services	8,368	73.51%
.xyz	New gTLD	XYZ.COM LLC	1,977	17.37%
.cn	ccTLD	China Internet Network Information Center(CNNIC)	307	2.70%
.net	gTLD	VeriSign Global Registry Services	203	1.78%
.vip	New gTLD	Minds + Machines Group Limited	97	0.85%
.cc	ccTLD	eNIC Cocos (Keeling) Islands Pty. Ltd. Island Internet Services	93	0.82%
.me	ccTLD	Government of Montenegro	49	0.43%
.app	New gTLD	Charleston Road Registry Inc.	45	0.40%
.top	New gTLD	Jiangsu Bangning Science & Technology Co.,Ltd.	35	0.31%
.co	ccTLD	.CO Internet S.A.S.	29	0.25%

IP Addresses. We used Qihoo 360 passive DNS [2] and VirusTotal [1] to collect IP addresses corresponding to the identified gambling servers. Since our data collection process lasted for a few months, and the gambling APPs could easily change their IPs, we used these two data sources that contain the historical records to collecting IP records. Overall, we have collected 13,931 unique IPs[3]. Then, we used `IP-to-ASN` mapping tables [21] to get the ASNs of these IP addresses. Table 4 shows the most popular ASNs (autonomous system number) of gambling app servers. As these illegal gambling apps are targeting Chinese users, over 49% of the IP addresses are located in mainland China. This result partly suggests the ineffectiveness of gambling service regulation. Besides, over 50% of the gambling server addresses are located outside of mainland China, including Hong Kong, the United States, Singapore, Malaysia, and other regions that have no/less regulation on gambling services.

Table 4: Top 10 ASNs of gambling app servers.

ASN	Country	ASN Description	IP Count	Percentage
4837	CN	CHINA169-BACKBONE	2,134	15.32%
59371	HK	Dimension Network & Communication Ltd.	1,392	9.99%
45102	CN	Alibaba (US) Technology Co., Ltd.	1,239	8.89%
4808	CN	China Unicom Beijing Province Network	879	6.31%
37963	CN	Hangzhou Alibaba Advertising Co.,Ltd.	479	3.44%
13335	US	Cloudflare, Inc.	318	2.28%
45753	HK	NETSEC-HK NETSEC NOC	294	2.11%
45090	CN	Shenzhen Tencent Computer Systems Company Limited	283	2.03%
134963	SG	Alibaba.com Singapore E-Commerce Private Limited	281	2.02%
55720	MY	Gigabit Hosting Sdn Bhd	275	1.97%

Registrants and Registrars. We next analyze the domain name registrants and registrars of gambling services by collecting their WHOIS records, which can reflect the ownership and provide a variety of domain name information. During our study, we found that several gambling domains had been expired for a long time (one or two years ago), which prevented us from obtaining information about these domain names. One possible reason behind this might be that, some gambling app servers rely on self-built DNS over HTTPS servers[4], which is easily for developer to distribute with some open source projects including DoH service like AdGuard Home [17]. Table 5 shows the top-10 registrars of gambling app servers. Although the most used domain name registrar is `GoDaddy.com`, many Chinese domain name registrars (e.g., Alibaba) are favored. Table 6 shows the top-10 registrant emails of gambling

[3]One gambling server may correspond to more than one IP address.
[4]https://en.wikipedia.org/wiki/DNS_over_HTTPS

app servers. Interestingly, almost all of them have enabled privacy settings. As such, we cannot get any useful information from the registrant email addresses. This evidence implies that *illegal gambling apps have the tendency to hide their operators' identities.*

Table 5: Top 10 registrars of gambling app servers.

Registrar	Count	Percentage
GoDaddy.com, LLC	3,896	34.42%
Alibaba Cloud Computing (Beijing) Co., Ltd.	612	5.41%
Name.com, Inc.	400	3.53%
eName Technology Co., Ltd.	214	1.89%
XINNET TECHNOLOGY CORPORATION	151	1.33%
MAFF Inc.	149	1.32%
22NET, INC.	141	1.25%
Alibaba Cloud Computing	105	0.93%
ALIBABA.COM SINGAPORE E-COMMERCE PRIVATE LIMITED	104	0.92%
Chengdu west dimension digital technology Co., LTD	92	0.81%

Table 6: Top 10 registrant emails of gambling app servers.

Registrant Email	Count	Percentage
abuse@godaddy.com	3,842	54.18%
DomainAbuse@service.aliyun.com	728	10.27%
abuse@name.com	407	5.74%
abuse@ename.com	214	3.02%
search-apnic-not-arin@apnic.net	182	2.57%
supervision@xinnet.com	154	2.17%
abuse@22.cn	141	1.99%
ipas@cnnic.cn	138	1.95%
abuse@namecheap.com	106	1.49%
abuse@namesilo.com	96	1.35%

5.2 Malicious Behaviors

We further want to explore, beyond the illegal gambling activities, whether these gambling apps have served for any malicious purposes. Here, we take advantage of VirusTotal [1], a widely-used online service that aggregated over 60 anti-virus engines, to identify potential malware among the gambling apps. We observe that, although most of the gambling apps were not flagged by anti-virus engines on VirusTotal, 2,988 APKs (56%) were flagged by at least one engine, and 21 APKs were flagged by at least 10 AV engines. We further use AVClass [37], a widely-used malware family labeling tool, to assign a family name for the malicious gambling apps reported by VirusTotal. Table 7 shows the top-10 malicious gambling apps ranked by the number of reported engines. For example, the app *yyc.app.web* is a malware reported by 20 anti-virus engines, which belongs to the *boogr* family. This type of malware attempts to disguise themselves as popular apps and can download other malicious files, send SMS messages to premium-rate numbers, or connect the victim's smartphone to the attacker's command-and-control server [22]. It shows that, *some gambling apps can also serve for other malicious purposes, although the percentage is not high.*

5.3 Third-party Services

We then analyze third-party services abused by illegal gambling apps. As we have collected the generated traffic of gambling apps during runtime (see Section 5.1), based on the labeled common domains obtained from 100,000 apps, we have successfully identified over 800 different third-party services used in 4,918 APKs. Table 8 shows the top 10 common third-party domain names used in gambling apps. We further label these common domain names (e.g., advertising, development, app analytics, etc.) by either visiting the domains or investigating them in search engines. We found that the third-party services used by gambling apps are mainly concentrated on development services, including bug collection (e.g., `android.bugly.qq.com`), third-party push notification services (e.g., `s.jpush.cn`) and self-upgrade services (e.g., `openinstall.io`), etc. This experimental result shows that *existing third-party services have almost no regulations on the apps using them.* However, for some services (e.g., push notification services), they should strictly limit the contents distributed on them, e.g., disallowing to push gambling related contents.

Table 7: Top 10 malicious gambling apps ranked by the number of flagged engines on VirusTotal.

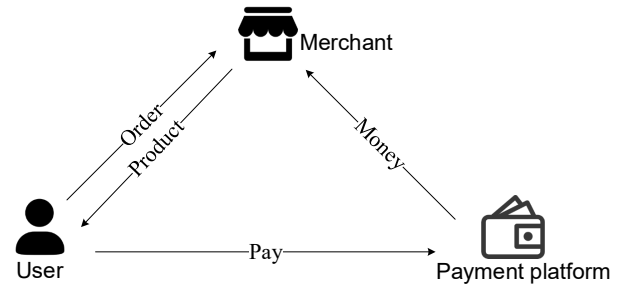
Package	Version	# AV Engines	Family
yyc.app.web	3.4.1	20	boogr
wt.nc	2.4	18	-
com.pkrss.pkstock	1.01	17	cnzz
org.haotan.abyl	1.0	12	hypay
com.hulk.example.xinpuj190303	1	12	jiagu
com.rb.android.XPJ04	1.0	11	-
com.pack.hongbaoshi824	2	11	jiagu
com.rb.android.ZGFC	1.0	11	jiagu
com.hulk.fenghcp	1	11	jiagu
com.rb.android.YB03	1.0	11	jiagu

Table 8: Top 10 abused third-party domains/third-party libraries/CNAMEs.

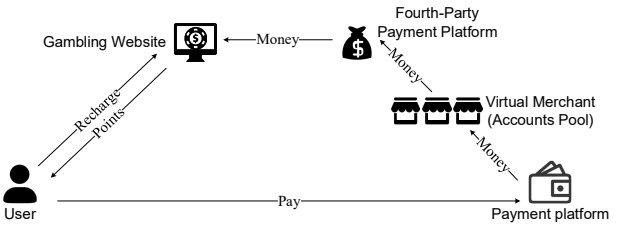
Third-party Domains	Apps	Third-party Libraries	Apps	CNAMEs	Domains
cfg.imtt.qq.com	1269	okhttp3	3853	yb550.com	1781
log.tbs.qq.com	1266	okio	3731	yb559.com	761
tbs.imtt.qq.com	1048	com/google/gson	3396	lxwaf.com	602
s.jpush.cn	923	com/tencent	2875	cdngslb.com	470
ali-stats.jpush.cn	819	com/bumptech/glide	2869	cdn-discuz.com	400
tsis.jpush.cn	790	com/google/zxing	2647	ppbk4.com	312
android.bugly.qq.com	788	io/reactivex	2301	c3-anti-ddos.com	264
gd-stats.jpush.cn	708	org/apache	2138	incapdns.net	254
openinstall.io	679	cn/jpush/android	2055	greycdn.net	188
play.googleapis.com	639	cn/jiguang	2026	rbnetidc.com	174

Next, we take advantage of LibRadar [31], a widely-used third-party library detection tool [59] for identifying the embedded third-party libraries in gambling apps. We detect 4,962 third-party libraries in total. On average, each app uses 88 third-party libraries. 849 libraries are used by at least 100 gambling apps, and 96 libraries are used by more than 1,000 gambling apps. Table 8 (column 2) shows the top-10 third-party libraries. Development-aid libraries, including `okhttp3` (network library), `okio` (I/O library), `Google gson` (Java serialization/deserialization library), are widely used in gambling apps. Besides, push notification services (e.g., `Jpush`) are favored by gambling apps as well, which is in line with our previous observations.

We notice that many gambling server names use the Canonical Name Record (CNAME) for DNS resolution, which makes it difficult for us to analyze the real server of the gambling APP. We leverage Qihoo 360 passive DNS [2] to collect all the history DNS



(a) Traditional payment process



(b) Fourth-party payment platform process

Figure 5: Traditional payment service (a) VS. fourth-party payment service (b).

data. We have collected 15,865 CNAME records in total, with 7,537 unique ones. For the 11,320 gambling app servers we collected, 5,541 of them have exploited CNAME services. On average, each gambling app server corresponds to 2.9 CNAME records. Table 8 (column 3) shows the top 10 CNAMEs we identified. It is interesting to infer their purposes from their names, e.g., Anti-DDOS (`c3-anti-ddos.com`), WAF (`lxwaf.com`) and CDN (`cdngslb.com`). But there are also some domains with many usages we can not know their functions. For example, `yb550.com` was used by 1,675 gambling server domains, while we cannot guess its purpose based on its name only.

5.4 Payment Services

Illegal gambling apps have the incentive to make a profit. In general, traditional payment channels, including online banking, credit card, and third-party payment services (e.g., Wechat pay and Alipay), are widely used in mobile apps in China. However, to avoid supervision, illegal gambling apps usually adopt *hidden payment channels*. To hide their identities and avoid potential risks, a preferred way is to utilize a number of *intermediate money transfer accounts* to receive payments, which means that the recipient accounts can always be different. However, considering that payment accounts can be blocked from time to time, they need to change accounts frequently. The cost of changing accounts is high. To mitigate this, we find in this work that gambling apps have "innovatively" used a new type of payment service, which we call it the *fourth-party payment service*. To the best of our knowledge, no previous work has mentioned the fourth-party payment service, while we observe that it has offered great benefit to the underground businesses.

Figure 5(a) shows the workflow of fourth-party payment service, which *is built atop third-party payment channels, with a mixing layer additional introduced.* The traditional third-party payment process

Table 9: A list of identified fourth-party payment services.

Platform	Alipay	WechatPay	UnionPay	EBank	QQPay	JDPay
henuser.ulan.com.cn	Y	Y				
www.wantong-pay.com	Y	Y				
www.777zhifu.com	Y					
newapi.xfuoo.com	Y	Y	Y	Y		
api.zfth666.com	Y		Y			
ppp.jqkpay.cc	Y					
www.duoshan8888.net	Y	Y				
sg.qingmigou.com	Y					
shop.dsf98.com	Y					
api.dypay68.com	Y					
lt.24boluo.com		Y				
5413548.tencent.com.hbasechina.org		Y				
7f5ebe92cff940f8874dac2b90018cfb.mahfvwy.cn		Y				
juhepay.ftzgm.com				Y		
app.ltjvxrd.cn					Y	
dianfeng789.top						Y
dzwm0mdgmu94w.cloudfront.net	Y					
bblnnu.corn100.xyz		Y				
www.slingn.cn				Y		
aa.luchengpay.com	Y					
amod.chenyingruye.com	Y					
47.56.213.98	Y					
8.210.55.75		Y				
47.99.161.39	Y					
api.zhifutong888.com	Y			Y		
jngzbl.com	Y					
api.cuicanxing888.com	Y					
mmblrgvdbe.6785151.com	Y	Y	Y			
api.tai3pay.com	Y					
www.buyibk.cn	Y					
47.115.32.159	Y					
goodhy.io	Y					
jid028.lklonghua.com				Y		
api.100.ccd100.com				Y		
pay.ffb169.com				Y		
api.aazzpay.com	Y					
zz.yashuli.com	Y					
s.huahaiteng.com	Y					
show.wiipay.cn					Y	
online-pay-qrcode.oss-cn-hangzhou.aliyuncs.com	Y					
gateway2.decimal25.com	Y					
api.hy8963.com	Y					
api.gtr8.net	Y	Y				
okex444.com			Y			

consists of *users*, *payment platforms* (e.g., Alipay and WechatPay), and merchants (i.e., the gambling app). In this scenario, the payment recipient can be explicitly identified. As a contrast, in the fourth-party payment, the payment platform acts as a *mixing service*. Their sole purpose is to *make the money flow difficult to track*. Thus, the payment platform usually provides many virtual merchant accounts to hide the real payment recipients' information. In this way, it is difficult for users to know where the money they paid ultimately went. Also, it poses challenges to regulatory authorities to trace the money flow and the involved gambling campaigns.

To further characterize the fourth-party payment platforms, we conducted a manual analysis of 10 gambling apps. We interact with these apps to reach the payment UI and intercept the network traffic to pinpoint the fourth-party payment platforms. Table 9 shows the results of our analysis. Surprisingly, these ten apps have used, in total, 44 fourth-party payment platforms in total. These payment platforms often provide different payment methods based on traditional third-party payment platforms (e.g., *Alipay*, *WechatPay*, *UniconPay*, *EBank*, *QQPay*, and *IDPay*).

Furthermore, we conduct a field study by automatically placing a large number of orders but not making actual payments. In this way, we can get a number of *virtual merchant accounts*. Note that, some gambling apps will ban abnormal accounts. As such, we cannot get too much data. Table 10 shows our experimental results on five payment platforms. We found that every platform utilizes many accounts to collect payments. A large number of virtual accounts ensures that payments are difficult to track. Thus, these kinds of payment methods are favored by illegal gambling apps.

Table 10: Payment accounts from platforms

Payment Platform	Order Count	Account Count
decimal25.com	2,176	193
buyibk.cn	1,103	723
luchengpay.com	531	86
huahaiteng.com	210	40
dianfeng789.top	57	6
Total	4,077	1,048

Answer to RQ2: *We have identified over 11K domains that are highly suspicious to be correlated with gambling services. Our experimental analyses reveal that (1) the regulation of gambling apps in China is quite ineffective, i.e., roughly half of the gambling servers are located in mainland China, (2) Gambling apps could be used to serve malicious purposes, (3) Third-party services (e.g., push notification services, CNAME service) could be abused by gambling apps, and (4) illegal gambling apps often leverage the so-called fourth-party payment platforms to handle transactions, in order to avoid supervision.*

6 INFERRING NEW GAMBLING APPS

In this section, we want to further investigate the underground groups of gambling apps, aiming at identifying illegal campaigns that create and operate gambling services or apps. Moreover, we want to identify new emerging gambling services by mining their relationship with the ones we have collected.

6.1 App Clustering

Our preliminary research found that many gambling apps in our dataset have similar UI structures. For example, as shown in Figure 6, the UI structures of three gambling apps are almost identical, i.e., only the app names and icons are different. Thus, we are wondering whether they are created using the same templates. Based on this hypothesis, we conduct a clustering analysis of gambling apps based on code-level similarity and their developer signatures to analyze the apps' relationships. There is no doubt that apps signed with the same developer signatures (except several common Android signatures) are created by the same illicit campaigns. Apps with high code-level similarity also suggest that they are highly suspicious of belonging to the same gambling family.

(a) APP 1 (b) APP 2 (c) APP 3

Figure 6: Gambling apps with similar UI structures.

Table 11: Top 10 gambling app clusters. ***# App indicates the number of gambling apps in this cluster, # Cert denotes the number of developer certificates used in this cluster, % Top-1 Cert indicates the proportion of gambling apps signed by the most popular certificate, # Prefix denotes the number of different kinds of package name prefixes, and % Top-1 Prefix indicates the the proportion of gambling apps shared by the most popular package prefix.***

# Apps	# Certs	% Top-1 Cert	# Prefix	% Top-1 Prefix
852	1	100.00%	4	99.53%
362	89	64.09%	121	30.66%
253	61	51.38%	1	100.00%
135	1	100.00%	1	100.00%
93	90	2.15%	60	33.33%
67	67	1.49%	1	100.00%
52	2	78.85%	2	78.85%
48	3	79.17%	48	2.08%
45	1	100.00%	2	97.78%
43	1	100.00%	3	93.02%

To analyze the code-level similarities between applications, we used FSquaDRA2 [14], a widely used app clone detection tool based on code similarity and resource similarity, to make pairwise comparisons between all apps. We calculate the similarity between any two apps of the 3,366 unique gambling apps. Note that, as we have collected multiple versions for some gambling apps, for each gambling app, we take the latest APK version to fulfill the clustering process. Then, we use DBSCAN clustering algorithm (Density-based spatial clustering of applications with noise) [53] on the similarity matrix to cluster the apps. We empirically set the threshold as 80% according to previous studies [24], which means that the apps will be clustered into the same group only when their similarity score is higher than 80%. Note that, for apps signed with the same app developer signature, we will merge them together, as they definitely belong to the same gambling campaign.

Finally, 3,048 gambling apps (90.6%) were clustered into 193 groups, with 318 apps remaining to be isolated. This clustering result suggests that gambling apps are usually operated in groups. Table 11 shows the top 10 clusters. Most apps in the same cluster share high similarities in package names, i.e., with the same package name prefix. For example, for the largest cluster, over 99% of the apps share the same package name prefix `com.yibo.app`, e.g., `com.yibo.app.b539` and `com.yibo.app.b479`.

6.2 Identifying New Gambling Apps

Our previous exploration suggests that gambling apps have close relationships, which enables us to identify new gambling services. Here, we propose a "guilt-by-association" expansion method to identify new suspicious gambling services and apps.

6.2.1 Inference based on HTTPS certificates. HTTPS (Hypertext Transfer Protocol Secure) is used for secure communication over a computer network, and is widely used in today's websites, including gambling websites. In the process of HTTPS-encrypted communication, websites need to use an authoritatively signed certificate to

prove its identity. An HTTPS certificate can be used for multiple domain names belonging to one entity, i.e., all the SAN (Subject Alternative Name) domain names in a certificate are owned by the same entity. Thus, we can leverage the information collected from the gambling domains to discover more related gambling services.

We use VirusTotal to collect the latest certificate information of all the gambling domain names we identified and then extract the SAN data in the certificate. By analyzing these data, we are able to identify 140,819 new domain names correlated with the known gambling services. To verify whether these newly discovered domains are indeed gambling services, we crawled all the contents of these domains using a headless browser. Since many domains are inaccessible (e.g., they are used as download sites), we can only crawl a total number of 53,749 websites successfully in the end. After that, we selected 1,000 websites for manual verification and found that 961 domains (96.1%) are gambling websites.

We then seek to identify more gambling apps from newly discovered gambling domains. Here, we take advantage of a feature provided by VirusTotal, i.e., we can trace the apps that communicate with these gambling servers. In this way, we have identified a total number of 16,973 apps. Among them, 2,043 apps have already been collected by us, and 8,198 apps share the same developer certificates with the gambling apps we collected. In order to determine whether other apps are gambling apps, we download 1,000 of them from VirusTotal for manual verification. Our exploration suggests 879 of them are actually gambling apps.

6.2.2 Inference based on app developer signatures. As the app developer signatures are privately owned by the developers, we further seek to expand the gambling app dataset based on analyzing their signatures. Note that some gambling apps developers may use known common keys in the community to sign apps. The most famous keys are the publicly known private keys included in the AOSP project. The standard Android build uses four known keys, all of which can be found at build/target/product/security. For example, TestKey is the generic default key for packages that do not otherwise specify a key. Other publicly-known keys include Platform (key), Shared (key), and Media (key). Thus we collect these keys and compare them with the signatures we extracted, and one of them was identified. For other developer signatures, we further search them on Google to confirm they are not publicly known signatures. At last, we have 1133 private signatures left.

Next, we use these private signatures to search for apps released by them on Koodous [23], one of the largest Android app repositories with over 65 million apps in total by the time of our study. At last, we have identified 41,995 apps in total, of which 252 apps are already in our dataset. The most popular 35 private signatures have released 38,084 apps in total. To further verify whether they are all gambling apps, we randomly select 100 apps for each popular private signature, with 3,500 apps in total. For 20 private signatures, all of their sampled apps are verified to be gambling apps. For the remaining 15 private signatures, besides gambling apps, we also observed a number of malware released by them. This observation suggests that gambling apps indeed fulfill a huge underground market, with thousands of developers and operators.

Answer to RQ3: *Using code-level and signature-level clustering analyses, we are able to identify over 200 gambling campaigns behind existing gambling apps. This finding further allows us to identify over 139K new suspicious gambling servers and thousands of gambling apps, following a "guilt-by-association" expansion method.*

7 DISCUSSION

7.1 Implication

Our observations in this paper can provide practical implications. First, although online gambling is illegal in China, we indeed observe a number of gambling servers deployed in mainland China, and .cn remains the third most popular TLD in our dataset, which suggests the ineffectiveness of gambling regulation. Second, hidden fourth-party payment services are widely used in gambling apps to hide the identities of money recipients, which facilitate the accessibility of illegal online gambling. We argue that, the regulators should pay special attention to these payment channels. Third, our investigation suggests that we can use a "guilt-by-association" method to identify new suspicious gambling services and apps, which could guide us to raise alarms when new related services/apps are found. Furthermore, we argue that all the legal parties should collaborate to address the issues introduced by illegal gambling apps. Indeed, ISP could leverage a blacklist of gambling websites to block apps from distribution. Following modern SNS platforms, which have already actively blocked gambling-related activities, IM software companies should also join the force to limit online gambling promotion and recruitment, and the payment channels have the responsibility to help track the money flow, etc.

7.2 Limitation

Our study carries several limitations. First, due to the resource limitation, we limit our study to illegal gambling apps in China, without considering illegal gambling apps in other countries. As different countries have diverse legal regulations on online gambling activities, more factors should be considered when studying the corresponding gambling apps. Second, when implementing the contributions, we attempt to choose straightforward (yet effective) rather than complicated solutions to achieve our purpose. Whenever possible, we always implement automated scripts to complete our tasks. Actually, most of the aforementioned tasks can be achieved automatically. Nevertheless, because of the various types of data sources and open-source tools involved in this work, it is hard to achieve our purpose fully automatically (e.g., we must register and login to different apps in order to assess the payment services). We hence resorted to some ad-hoc manual analysis to supplement the aforementioned automated methods. Third, we did not study the actual contents of gambling apps, i.e., the offered games. Note that it is non-trivial to categorize gambling apps automatically. During our initial exploration, we attempted to label them based on a topic-modeling approach, but the results are not satisfactory. We thereby resort to a semi-automated process to further categorize the previously confirmed gambling apps by manually analyzing the runtime app UIs we recorded. We observed that most of the gambling apps contain multiple types of gambling games including sports gambling, casino, poker, and lottery, etc. In fact, the gambling app is likely to have scams in the game, although we cannot easily detect them using traditional program analysis techniques.

8 RELATED WORK

8.1 Online Gambling Analysis

There are a few studies on analyzing illegal online gambling. Hao et al. [56] characterized the ecosystem of web-based online gambling services. They proposed a method for identifying gambling websites based on machine learning and successfully identified a lot of suspicious illegal gambling websites. They characterized these illegal gambling websites from a number of perspectives. In our work, we used a similar method to identify gambling websites. Some studies were focused on illegal online gambling from a social and economic perspective. Blaszczynski et al. [5] have characterized the relationship between gambling and crime, especially illegal gambling. Wang et al. [51] analyzed the Chinese gambling organizations and their countermeasures in economy, marketing, debt collection, and police suppression. Brooks et al. [6] examined the relationship between the regulated online gambling sectors. Online gambling apps, to the best of our knowledge, have not been touched by our research community.

8.2 Understanding the Mobile App Ecosystem

A number of efforts in our community have been focused on analyzing the mobile app ecosystem [11, 15, 19, 20, 34, 43, 47, 48]. For example, Wang et al. [48] conducted a large-scale analysis of over 6 million Android apps to understand various features of several Chinese Android app stores and how they compare to Google Play. Gamba et al. [15] performed a large-scale study of the ecosystem of pre-installed Android apps. Several recent studies [11, 19] have analyzed the ecosystem of incentivized mobile app install campaigns that require users to install mobile apps and perform in-app tasks.

8.3 Mobile App Analysis

Various kinds of techniques have been proposed to detecting and analyzing emerging issues in the mobile app ecosystem, including mobile malware [16, 33, 49, 61], privacy and security issues [13, 28, 30, 32, 45, 55], fraudulent behaviors [7–9, 27, 36], app clone and fake apps [14, 18, 42], attack and vulnerabilities [38, 46, 50, 57], and gray behaviors [4, 29, 44, 60], etc. Our work can take advantage of existing techniques to understand the characteristics of the unexplored mobile gambling apps.

9 CONCLUSION

In this paper, we make the first step to characterize the ecosystem of illegal mobile gambling apps. We first make efforts to create a large gambling app dataset and then analyze them from a number of perspectives, including distribution channels, network infrastructure, malicious behaviors, third-party and payment services, etc. We further propose a "guilt-by-association" expansion method to identify new suspicious gambling services and apps. Our experimental findings demonstrate the necessity to better regulate illegal gambling apps so as to protect users from potential risks.

ACKNOWLEDGMENTS

This work was supported by the National Key Research and Development Program of China (grants No.2018YFB0803600), the National Natural Science Foundation of China (grants No.62072046), Hong Kong RGC Projects (No. 152223/17E,152239/18E, CityU C1008-16G), the Australian Research Council (ARC) under a Discovery Early Career Researcher Award (DECRA) Project DE200100016 and a Discovery Project DP200100020.

REFERENCES

[1] [n.d.]. VirusTotal. https://www.virustotal.com/.
[2] 360.com. [n.d.]. Network Security Research Lab at 360. https://netlab.360.com/.
[3] Alexa. [n.d.]. Alexa top websites. http://www.alexa.com/topsites/category/Top/Computers/Internet/DomainNames.
[4] Benjamin Andow, Adwait Nadkarni, Blake Bassett, William Enck, and Tao Xie. 2016. A study of grayware on google play. In *2016 IEEE Security and Privacy Workshops (SPW)*. IEEE, 224–233.
[5] James Banks. 2016. *Online gambling and crime: Causes, controls and controversies*. Routledge.
[6] Graham Brooks. 2012. Online gambling and money laundering:"views from the inside". *Journal of Money Laundering Control* (2012).
[7] Jonathan Crussell, Ryan Stevens, and Hao Chen. 2014. Madfraud: Investigating ad fraud in android applications. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services*. 123–134.
[8] Feng Dong, Haoyu Wang, Li Li, Yao Guo, Tegawendé F Bissyandé, Tianming Liu, Guoai Xu, and Jacques Klein. 2018. Frauddroid: Automated ad fraud detection for android apps. In *Proceedings of the 2018 26th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 257–268.
[9] Feng Dong, Haoyu Wang, Li Li, Yao Guo, Guoai Xu, and Shaodong Zhang. 2018. How do mobile apps violate the behavioral policy of advertisement libraries?. In *Proceedings of the 19th International Workshop on Mobile Computing Systems & Applications*. 75–80.
[10] Kun Du, Hao Yang, Zhou Li, Haixin Duan, and Kehuan Zhang. 2016. The Ever-Changing Labyrinth: A Large-Scale Analysis of Wildcard {DNS} Powered Blackhat {SEO}. In *25th {USENIX} Security Symposium ({USENIX} Security 16)*. 245–262.
[11] Shehroze Farooqi, Álvaro Feal, Tobias Lauinger, Damon McCoy, Zubair Shafiq, and Narseo Vallina-Rodriguez. 2020. Understanding Incentivized Mobile App Installs on Google Play Store. In *Proceedings of the ACM Internet Measurement Conference (IMC '20)*. 696–709.
[12] P. Faruki, A. Bharmal, V. Laxmi, V. Ganmoor, M. S. Gaur, M. Conti, and M. Rajarajan. 2015. Android Security: A Survey of Issues, Malware Penetration, and Defenses. *IEEE Communications Surveys & Tutorials* 17, 2 (2015), 998–1022.
[13] Adrienne Porter Felt, Elizabeth Ha, Serge Egelman, Ariel Haney, Erika Chin, and David Wagner. 2012. Android permissions: User attention, comprehension, and behavior. In *Proceedings of the eighth symposium on usable privacy and security*. 1–14.
[14] Olga Gadyatskaya, Andra-Lidia Lezza, and Yury Zhauniarovich. 2016. Evaluation of Resource-based App Repackaging Detection in Android. In *Proceedings of the 21st Nordic Conference on Secure IT Systems (NordSec 2016)*. 135–151.
[15] J. Gamba, M. Rashed, A. Razaghpanah, J. Tapiador, and N. Vallina-Rodriguez. 2020. An Analysis of Pre-installed Android Software. In *2020 IEEE Symposium on Security and Privacy (SP)*. 1039–1055. https://doi.org/10.1109/SP40000.2020.00013
[16] Ren He, Haoyu Wang, Pengcheng Xia, Liu Wang, Yuanchun Li, Lei Wu, Yajin Zhou, Xiapu Luo, Yao Guo, and Guoai Xu. 2020. Beyond the virus: A first look at coronavirus-themed mobile malware. *arXiv preprint arXiv:2005.14619* (2020).
[17] AdGuard Home. [n.d.]. AdGuard Home. https://github.com/AdguardTeam/AdGuardHome.
[18] Yangyu Hu, Haoyu Wang, Ren He, Li Li, Gareth Tyson, Ignacio Castro, Yao Guo, Lei Wu, and Guoai Xu. 2020. Mobile app squatting. In *Proceedings of The Web Conference 2020*. 1727–1738.
[19] Yangyu Hu, Haoyu Wang, Li Li, Yao Guo, Guoai Xu, and Ren He. 2019. Want to earn a few extra bucks? a first look at money-making apps. In *2019 IEEE 26th International Conference on Software Analysis, Evolution and Reengineering (SANER)*. IEEE, 332–343.
[20] Yangyu Hu, Haoyu Wang, Yajin Zhou, Yao Guo, Li Li, Bingxuan Luo, and Fangren Xu. 2018. Dating with scambots: Understanding the ecosystem of fraudulent dating applications. *arXiv preprint arXiv:1807.04901* (2018).
[21] iptoasn.com. [n.d.]. Free IP address to ASN database. https://iptoasn.com/.
[22] kaspersky. [n.d.]. Trojan.AndroidOS.Boogr. https://threats.kaspersky.com/en/threat/Trojan.AndroidOS.Boogr/.
[23] koodous.com. [n.d.]. Koodous. https://koodous.com/.
[24] Li Li, Tegawendé F Bissyandé, Jacques Klein, and Yves Le Traon. 2016. An Investigation into the Use of Common Libraries in Android Apps. In *The 23rd IEEE International Conference on Software Analysis, Evolution, and Reengineering (SANER 2016)*.
[25] Li Li, Jun Gao, Médéric Hurier, Pingfan Kong, Tegawendé F Bissyandé, Alexandre Bartel, Jacques Klein, and Yves Le Traon. 2017. AndroZoo++: Collecting Millions

of Android Apps and Their Metadata for the Research Community. *arXiv preprint arXiv:1709.05281* (2017).

[26] Jialiu Lin, Shahriyar Amini, Jason I. Hong, Norman Sadeh, Janne Lindqvist, and Joy Zhang. 2012. Expectation and Purpose: Understanding Users' Mental Models of Mobile App Privacy through Crowdsourcing. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing (UbiComp '12)*. 501–510.

[27] Bin Liu, Suman Nath, Ramesh Govindan, and Jie Liu. 2014. {DECAF}: Detecting and characterizing ad fraud in mobile apps. In *11th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 14)*. 57–70.

[28] Minxing Liu, Haoyu Wang, Yao Guo, and Jason Hong. 2016. Identifying and analyzing the privacy of apps for kids. In *Proceedings of the 17th International Workshop on Mobile Computing Systems and Applications*. 105–110.

[29] Tianming Liu, Haoyu Wang, Li Li, Guangdong Bai, Yao Guo, and Guoai Xu. 2019. Dapanda: Detecting aggressive push notifications in android apps. In *2019 34th IEEE/ACM International Conference on Automated Software Engineering (ASE)*. IEEE, 66–78.

[30] Tianming Liu, Haoyu Wang, Li Li, Xiapu Luo, Feng Dong, Yao Guo, Liu Wang, Tegawendé Bissyandé, and Jacques Klein. 2020. MadDroid: Characterizing and Detecting Devious Ad Contents for Android Apps. In *Proceedings of The Web Conference 2020*. 1715–1726.

[31] 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.

[32] Rahul Pandita, Xusheng Xiao, Wei Yang, William Enck, and Tao Xie. 2013. {WHYPER}: Towards automating risk assessment of mobile applications. In *22nd {USENIX} Security Symposium ({USENIX} Security 13)*. 527–542.

[33] Sancheng Peng, Shui Yu, and Aimin Yang. 2013. Smartphone malware and its propagation modeling: A survey. *IEEE Communications Surveys & Tutorials* 16, 2 (2013), 925–941.

[34] Thanasis Petsas, Antonis Papadogiannakis, Michalis Polychronakis, Evangelos P Markatos, and Thomas Karagiannis. 2017. Measurement, modeling, and analysis of the mobile app ecosystem. *ACM Transactions on Modeling and Performance Evaluation of Computing Systems (TOMPECS)* 2, 2 (2017), 1–33.

[35] Google Play. [n.d.]. Real-Money Gambling, Games, and Contests. https://play.google.com/about/restricted-content/gambling/.

[36] Mizanur Rahman, Nestor Hernandez, Ruben Recabarren, Syed Ishtiaque Ahmed, and Bogdan Carbunar. 2019. The Art and Craft of Fraudulent App Promotion in Google Play. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. 2437–2454.

[37] Marcos Sebastián, Richard Rivera, Platon Kotzias, and Juan Caballero. 2016. Avclass: A tool for massive malware labeling. In *International Symposium on Research in Attacks, Intrusions, and Defenses*. Springer, 230–253.

[38] Yutian Tang, Yulei Sui, Haoyu Wang, Xiapu Luo, Hao Zhou, and Zhou Xu. 2020. All your app links are belong to us: understanding the threats of instant apps based attacks. In *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering*. 914–926.

[39] Timothy Vidas and Nicolas Christin. 2014. Evading android runtime analysis via sandbox detection. In *Proceedings of the 9th ACM symposium on Information, computer and communications security*. ACM, 447–458.

[40] Nicolas Viennot, Edward Garcia, and Jason Nieh. 2014. A measurement study of google play. In *The 2014 ACM international conference on Measurement and modeling of computer systems*. 221–233.

[41] Haoyu Wang and Yao Guo. 2017. Understanding third-party libraries in mobile app analysis. In *2017 IEEE/ACM 39th International Conference on Software Engineering Companion (ICSE-C)*. IEEE, 515–516.

[42] Haoyu Wang, Yao Guo, Ziang Ma, and Xiangqun Chen. 2015. WuKong: a scalable and accurate two-phase approach to Android app clone detection. In *Proceedings of the 2015 International Symposium on Software Testing and Analysis*. ACM, 71–82.

[43] Haoyu Wang, Hao Li, and Yao Guo. 2019. Understanding the evolution of mobile app ecosystems: A longitudinal measurement study of google play. In *The World Wide Web Conference*. 1988–1999.

[44] Haoyu Wang, Hao Li, Li Li, Yao Guo, and Guoai Xu. 2018. Why are android apps removed from google play? a large-scale empirical study. In *2018 IEEE/ACM 15th International Conference on Mining Software Repositories (MSR)*. IEEE, 231–242.

[45] Haoyu Wang, Yuanchun Li, Yao Guo, Yuvraj Agarwal, and Jason I Hong. 2017. Understanding the purpose of permission use in mobile apps. *ACM Transactions on Information Systems (TOIS)* 35, 4 (2017), 1–40.

[46] Haoyu Wang, Hongxuan Liu, Xusheng Xiao, Guozhu Meng, and Yao Guo. 2019. Characterizing Android app signing issues. In *2019 34th IEEE/ACM International Conference on Automated Software Engineering (ASE)*. IEEE, 280–292.

[47] Haoyu Wang, Zhe Liu, Yao Guo, Xiangqun Chen, Miao Zhang, Guoai Xu, and Jason Hong. 2017. An explorative study of the mobile app ecosystem from app developers' perspective. In *Proceedings of the 26th International Conference on World Wide Web*. 163–172.

[48] Haoyu Wang, Zhe Liu, Jingyue Liang, Narseo Vallina-Rodriguez, Yao Guo, Li Li, Juan Tapiador, Jingcun Cao, and Guoai Xu. 2018. Beyond google play: A large-scale comparative study of chinese android app markets. In *Proceedings of the Internet Measurement Conference 2018*. 293–307.

[49] Haoyu Wang, Junjun Si, Hao Li, and Yao Guo. 2019. Rmvdroid: towards a reliable android malware dataset with app metadata. In *2019 IEEE/ACM 16th International Conference on Mining Software Repositories (MSR)*. IEEE, 404–408.

[50] Hui Wang, Yuanyuan Zhang, Juanru Li, Hui Liu, Wenbo Yang, Bodong Li, and Dawu Gu. 2015. Vulnerability assessment of oauth implementations in android applications. In *Proceedings of the 31st annual computer security applications conference*. 61–70.

[51] Peng Wang and Georgios A Antonopoulos. 2016. Organized crime and illegal gambling: How do illegal gambling enterprises respond to the challenges posed by their illegality in China? *Australian & New Zealand Journal of Criminology* 49, 2 (2016), 258–280.

[52] Wikipedia. [n.d.]. CNAME record. https://en.wikipedia.org/wiki/CNAME_record.

[53] Wikipedia. [n.d.]. DBSCAN. https://en.wikipedia.org/wiki/DBSCAN.

[54] Wikipedia. [n.d.]. Online Gambling. https://en.wikipedia.org/wiki/Online_gambling.

[55] Shengqu Xi, Shao Yang, Xusheng Xiao, Yuan Yao, Yayuan Xiong, Fengyuan Xu, Haoyu Wang, Peng Gao, Zhuotao Liu, Feng Xu, et al. 2019. DeepIntent: Deep icon-behavior learning for detecting intention-behavior discrepancy in mobile apps. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. 2421–2436.

[56] Hao Yang, Kun Du, Yubao Zhang, Shuang Hao, Zhou Li, Mingxuan Liu, Haining Wang, Haixin Duan, Yazhou Shi, Xiaodong Su, et al. 2019. Casino royale: a deep exploration of illegal online gambling. In *Proceedings of the 35th Annual Computer Security Applications Conference*. 500–513.

[57] Quanqi Ye, Yan Zhang, Guangdong Bai, Naipeng Dong, Zhenkai Liang, Jin Song Dong, and Haoyu Wang. 2019. LightSense: A Novel Side Channel for Zero-permission Mobile User Tracking. In *International Conference on Information Security*. Springer, 299–318.

[58] 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)*. 23–26.

[59] Xian Zhan, Lingling Fan, Tianming Liu, Sen Chen, Li Li, Haoyu Wang, Yifei Xu, Xiapu Luo, and Yang Liu. 2020. Automated Third-Party Library Detection for Android Applications: Are We There Yet?. In *2020 35th IEEE/ACM International Conference on Automated Software Engineering (ASE)*. IEEE, 919–930.

[60] Hao Zhou, Haoyu Wang, Yajin Zhou, Xiapu Luo, Yutian Tang, Lei Xue, and Ting Wang. 2020. Demystifying Diehard Android Apps. In *2020 35th IEEE/ACM International Conference on Automated Software Engineering (ASE)*. IEEE, 187–198.

[61] Yajin Zhou and Xuxian Jiang. 2012. Dissecting android malware: Characterization and evolution. In *2012 IEEE symposium on security and privacy*. IEEE, 95–109.