Tickets or Privacy? Understand the Ecosystem of Chinese Ticket Grabbing Apps

Yijing Liu and Yiming Zhang, *Tsinghua University;* Baojun Liu, *Tsinghua University; Zhongguancun Laboratory;* Haixin Duan, *Tsinghua University; Quancheng Laboratory;* Qiang Li, *Qihoo 360;* Mingxuan Liu, *Zhongguancun Laboratory;* Ruixuan Li and Jia Yao, *Tsinghua University*

https://www.usenix.org/conference/usenixsecurity24/presentation/liu-yijing

This paper is included in the Proceedings of the 33rd USENIX Security Symposium.

August 14–16, 2024 • Philadelphia, PA, USA

978-1-939133-44-1

Tickets or Privacy? Understand the Ecosystem of Chinese Ticket Grabbing Apps

Yijing Liu[1] Yiming Zhang[1] Baojun Liu[1,2,*] Haixin Duan[1,3]
Qiang Li[4] Mingxuan Liu[2] Ruixuan Li[1] Jia Yao[1]

[1]*Tsinghua University,* [2]*Zhongguancun Laboratory,* [3]*Quancheng Laboratory,* [4]*Qihoo 360*

Abstract

Due to the prevalence of scalping and the promotion of real-name ticketing systems, user-oriented mobile ticket grabbing apps have become a popular pattern for scalpers. Compared with traditional scalper-oriented scalping, ticket grabbing apps pose security and privacy risks to users directly. In our study, we take the first step towards revealing the ticket grabbing app ecosystem from the perspectives of app developers, app users, and target platforms synthetically.

We built a large-scale dataset of ticket grabbing apps in the wild within China, containing 758 Chinese ticket grabbing apps with 3,121 versions. Based on the detailed analysis of these apps, we found that ticket grabbing has formed a mature industrial chain, with various specialized technical characteristics to enhance the success rate, such as the abuse of Android accessibility services. We also revealed the profit model of ticket grabbing apps, and disclosed severe security and privacy hazards they pose to end users, including the collection of sensitive information and continuous screenshots. We further conducted an online survey involving 184 participants to get users' usage and privacy concerns on ticket grabbing apps, and regrettably found that users prioritize "tickets" over "privacy". Finally, we proposed an "Indirect Combat" approach to assist in the defense mechanisms. In summary, our findings provide target platforms and users with a better understanding of the ticket grabbing app ecosystem in China, enabling them to better detect and combat these apps.

1 Introduction

For some limited resources such as train tickets, concert or sports tickets, the resource providers frequently fall short of meeting market demands. However, people often exhibit significant enthusiasm towards these sought-after resources. In order to fulfill the intense demand of consumers, "scalping"[1] behavior came into being. Scalping behavior is a global phenomenon with significant negative consequences. It could disrupt market equilibrium and impose economic losses on original consumers [50, 73, 77]. It is a very frequent and frustrating reality when fans go to the ticket websites in the first time, only to find that there are no more tickets left [41]. Facing the prevalence of scalping, many countries and regions have published legislation to clarify the illegality of scalping [7, 17, 33], such as the "Better Online Ticket Sales (BOTS) Act" [7] in the United States.

As technology advanced, scalpers turned to automate their purchasing behavior. Ticket bots were once the most frequently used tools by scalpers. They snapped up a large number of limited resources in advance and made a profit through price differences during the resale process in the secondary market. We call this traditional scalping *"Scalper-driven model"*. According to the 2023 Bad Bot Report [1] on entertainment websites, 83.4% of total global traffic came from automation. However, with the promotion of the real-name ticket systems [91, 92], it is more difficult for scalpers to directly acquire numerous resources using ticket bots. Moreover, some research has explored the detection measures against ticket bots [90]. Therefore, the scalpers changed their grabbing method [68]. They developed user-oriented mobile "ticket grabbing apps" that allow users to directly access specific resources and the scalpers can generate profit from users during the usage. These apps do not acquire large quantities of resources in bulk. Instead, they collect users' personal information in advance and subsequently make purchases on behalf of the users. We call this scalping *"User-driven model"*, which is the primary focus of our research.

Motivation. In our work, we concentrated on the ticket grabbing apps and tried to explore the ecosystems behind these apps. Ticket grabbing apps, compared to ticket bots, represent an upgraded version that adapts to real-name ticket systems. Ticket bots are run by scalpers, while the ticket grabbing apps interact directly with users, posing more immediate risks to them and some apps themselves are even viruses [31]. Moreover, despite the government's continuous advocacy to the

[*] Corresponding author: Baojun Liu.

[1] Scalping refers to the "acquisition of goods or services in a manner that a normal user would be unable to undertake manually." [24]

public not to use unofficial channels to buy tickets, the usage of ticket grabbing apps has not been effectively alleviated. Users' understanding of ticket grabbing apps, and their privacy concerns are still unclear.

Due to the potential risks and chaos posed by ticket grabbing apps, it is of great significance to understand the ecosystem of ticket grabbing apps which can help maintain market fairness and protect the privacy of consumers. However, there is currently a lack of research focusing on ticket grabbing apps. Although some work [68,73] has analyzed the implementation of ticket bots based on real cases, there has been no large-scale measurement of ticket grabbing apps. Therefore, our work marks the first exploration of the ecosystem of ticket grabbing apps, investigating both their software implementation and user privacy perception. Certainly, the ecosystem of mobile ticket grabbing apps involves three parties: app developers, app users, and target platforms (which supply resources to be snapped up). So, we raised our research questions from the perspectives of these three parties separately.

- RQ1: [App Developers] What technical tricks do developers employ to implement their grabbing functions? How do they make money from ticket grabbing apps?
- RQ2: [App Users] What security risks do ticket grabbing apps pose to users? Are users aware of these risks?
- RQ3: [Target Platforms] How is the relationship between ticket grabbing apps and target platforms? What methods do the apps utilize to evade detection by target platforms?

Our work. Considering that there is no publicly available dataset for ticket grabbing apps and the apps might be swiftly removed or frequently updated, we collaborated with a renowned security vendor in China to build a large-scale dataset. The dataset consists of 758 distinct ticket grabbing apps, totaling 3,121 versions. Then, using a combination of static and dynamic analysis methods, we gained insights into the technical characteristics of these apps and the potential security risks. Additionally, we conducted an online survey involving 184 participants to gauge the social characteristics of the apps and assess participants' privacy concerns. Finally, according to the analysis results, we summarized the gang characteristics and bypass tricks of ticket grabbing apps. Additionally, we proposed suggestions on cracking down the development and distribution of ticket grabbing apps. While our research focused on a China-wide dataset, yet ticket grabbing apps are not exclusive to China. Similar tools are used worldwide [34,35] and our findings have broader applicability. We chose China as a representative case due to the substantial demand and usage for ticket grabbing apps in this region.

Specifically, from the perspective of developers (Section 4), we found that ticket grabbing apps are highly automated. Some apps use scalper servers as intermediaries to send purchase requests to target servers, while others abuse Android accessibility services to achieve automated grabbing directly at the user end. In addition, we identified the profit models employed by ticket grabbing apps, often using the free version as bait. Then, from a user's perspective (Section 5), we revealed that ticket grabbing apps have certain security and privacy risks, such as the acquisition of numerous dangerous permissions and the collection of sensitive information. Although 70.59% of the participants are aware of the risks of ticket grabbing apps, due to the strong desire for tickets, 84.5% of the participants still used ticket grabbing apps. Finally, from the perspective of the target platforms (Section 6), we noticed that different ticket grabbing apps may have similar user interface (UI) designs, and they commonly conceal developers' identity information within their signatures. We also summarized the bypass tricks used by ticket grabbing apps to evade the continuously improving detection mechanisms of target platforms, such as CAPTCHA solving services, multiple IP proxies, etc. Furthermore, we proposed an "Indirect Combat" method to prevent the development and spread of ticket grabbing apps. Overall, our work is the first to examine the ecosystem of the ticket grabbing apps in China, providing new insights into understanding these apps.

Contributions. Contributions of the paper are as follows:

- *New underground industry.* We take the first step towards analysis of ticket grabbing apps ecosystem, which is a rapidly developing underground industry. Our work built the largest dataset of Chinese ticket grabbing apps to date, unveiling both the technical and social characteristics behind the apps.
- *New privacy risk.* Our work revealed for the first time the security risks posed by ticket grabbing apps and obtained users' usage and privacy concerns about the apps.
- *New insights.* Our analysis of gang characteristics and bypass methods provides new insights into the governance of ticket grabbing apps. We put forward an "Indirect Combat" approach and three specific suggestions that could indirectly crack down on the emergence of ticket grabbing apps.

2 Background

Workflow. Figure 1 illustrates the typical workflow of ticket grabbing apps. To delegate the grabbing task, users need to provide the order information and other necessary information (e.g., login credentials) to the app. Typically, this step entails an additional payment to the developer (①) to activate the grabbing functionality. Subsequently, the ticket grabbing apps interact with the target platforms to procure the desired goods on behalf of users (②), and relay the result back to them (③). Upon successful acquisition of the goods, users can find their pending orders on the target platform and complete the necessary payments to finish the transaction (④). Note that step ④ can be skipped, as certain apps offer users the option to directly input their payment passwords in step ①, thereby facilitating automated payment processing.

Comparison with ticket bots. Both ticket bots and ticket grabbing apps are tools for acquiring tickets, yet their operational models vary significantly. Ticket bots employ mass account creation to increase their chances of securing more

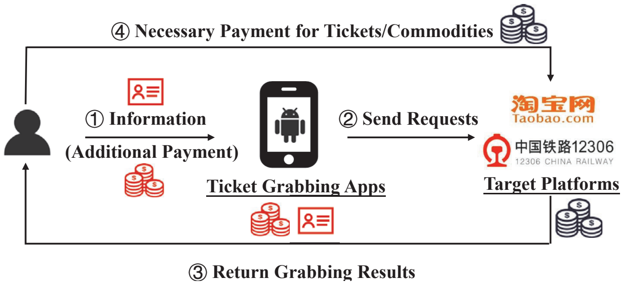


Figure 1: Typically workflow of ticket grabbing apps.

resources. They are primarily operated by scalpers, who then profit by reselling tickets. In contrast, ticket grabbing apps are operated directly by users, and profit from their promised ticket acquisition capabilities. These apps can be seen as successors to ticket bots, adapting to the ongoing improvements in ticketing systems, particularly the real-name authentication mechanism that restricts bulk registration.

Threat model. Ticket grabbing apps severely disrupt the order of market transactions, putting anyone at risk of falling victim. Those who do not use these apps may miss out on desired tickets due to others using such apps. For app users, ticket grabbing apps could directly access their private information and control their devices. Our research focuses on the security threats of ticket grabbing apps, with app users emerging as the direct victims, i.e., individuals who are eager to acquire sought-after tickets or commodities. These users turn to ticket grabbing apps with the expectation of increasing their chances of successful purchases. However, as depicted in Figure 1, using ticket grabbing apps requires users to disclose necessary information, including credentials and sometimes payment passwords. This raises privacy risks concerning the potential disclosure of personal information to unauthorized third parties and also instills security concerns for users regarding their devices being abused for malicious operations. Besides, users may be enticed to pay an additional fee to activate the grabbing feature (see Section 4.2). However, it remains uncertain whether this payment results in more successful grabbing endeavors, potentially leading innocent users into scams.

3 Methodology

To answer the above research questions, we first built a large-scale dataset of Chinese ticket grabbing apps, then performed app analysis to uncover their technical features and privacy implications. We also conducted an online survey to get the usage and social characteristics of ticket grabbing apps from the user's perspective. The results of RQ1 and RQ2 are based on the app analysis and user study synthetically. The results of RQ3 are obtained separately from app analysis. In this section, we will introduce the methods of data collection and app analysis, as well as the design of the user study.

3.1 Dataset

Currently, there is no publicly available dataset specifically targeting ticket grabbing apps. Furthermore, considering that the ticket grabbing apps might be swiftly removed or frequently updated, it is difficult to directly crawl the apps from app stores. Therefore, we collaborated with a renowned Chinese security vendor named "Qihoo 360" to build up the first dataset of Chinese ticket grabbing apps in the wild. We first obtained some typical ticket grabbing apps from the search engines as the seed dataset, then expanded our dataset with more apps in "360 App Assistant" using snowball sampling [56]. The process of data collection is depicted in Figure 2.

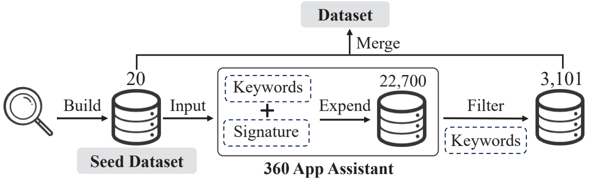


Figure 2: The steps of data collection.

Specifically, we first simulated a user's behavior who is eager to search for a ticket grabbing app. Utilizing Baidu, the most widely used search engine in China, we entered the keyword "ticket grabbing apps" and identified four common types of ticket grabbing apps according to the results, including (1) train grabbing apps, (2) concert/sports grabbing apps, (3) hospital appointment grabbing apps, and (4) commodity grabbing apps. Then, we conducted a manual selection process based on descriptions from the search results and finally selected the five most typical ticket grabbing apps for each type. A total of 20 apps were selected as the seed dataset for the next phase of data collection.

Then, we used the snowball sampling method to expand the seed dataset. We searched apps with similar application names from "360 App Assistant", a well-known free app store in China. The keyword we used was "grabbing" as almost all the application names in the seed dataset contain this word. Among the queried apps, we sorted them according to the number of installations and selected the top 200. Leveraging the app signatures, we established connections with other apps that shared the same signature, as it indicated that they likely originated from the same developer and offered similar functionalities. This step resulted in the expansion of the dataset to a total of 22,700 apps.

Finally, we again filtered the apps by application name using more precise keywords as there are a lot of apps unrelated to ticket grabbing. The keywords we used include "Maotai grabbing", "Shoe grabbing", "Ticket grabbing", "Appointment grabbing", and we finally got 3,101 apps after filtration.

Dataset overview. The data collection spanned April 2023 and we collected 3,121 ticket grabbing apps, including the 20 apps in the seed dataset. Within the dataset, “Ticket Grabbing Assistant” was the most prevalent application name, accounting for 43.7%. The average installation count for these apps was 16,198. Furthermore, to provide a basis for contrasting ticket grabbing apps with other benign apps, we randomly selected 50 apps from the top download lists of both “Google Play” and “360 App Assistant” (the same source as the selected grabbing apps) respectively, amounting to 100 apps in total. The selected ticket grabbing apps and the list of benign apps are publicly available on the online appendix [22].

3.2 App Analysis

After data collection, to obtain the technical characteristics of ticket grabbing apps, as well as the privacy and security risks posed by them, we conducted a synthetical analysis of the apps in our dataset. To avoid non-redundant analysis [69], we excluded the apps sharing identical package names, signatures, and app names, considering them as different versions of the same app. We chose the apps with the highest number of downloads as the representative and finally selected 758 unique apps as the primary subjects of our analysis. In addition, among the rest of the apps with duplicate versions, we chose seven app groups that have more than 100 versions for additional version analysis. Details of these app groups are provided in Table 2 in Appendix A. Analysis methods include simulated runs, static analysis, and dynamic analysis. Note that the primary conclusions in this paper are derived from 758 distinct applications. We only use the seven groups of applications for version analysis.

- **Simulated runs.** To gain an intuitive understanding of the functionality of ticket grabbing apps, we conducted simulated runs using LDPlayer [18], a free Android emulator. By creating a virtual runtime environment, we gained insights into how the apps are used and explored the various functions they offer. To ensure compatibility, we utilized a rooted Android 9 (64-bit) phone, as some apps require a rooted environment to function properly. In our experiments, we manually selected a set of representative apps that encompassed the four types of ticket grabbing apps for simulation.
- **Static analysis.** AndroGuard [3] provides robust capabilities for comprehensive analysis of various aspects of an Android application. We leveraged this tool to extract crucial information such as the application’s signature, requested permissions, and embedded strings. By utilizing AndroGuard, we were able to gain valuable insights into the inner workings and characteristics of the ticket grabbing apps.
- **Dynamic analysis.** Our work incorporates two components of dynamic analysis. Firstly, during the simulated runs, we used Fiddler [13] to capture the traffic data generated by the apps. Additionally, we employed the Qianxin sandbox [27] to conduct more extensive dynamic analysis on 758 apps in our dataset. Based on the sandbox results, we can retrieve the requested domain names, dynamic behaviors, and real-time screenshots of the apps. With the comprehensive results of dynamic analysis, we gained deeper insights into the apps’ behavior, interactions, and potential security vulnerabilities.

3.3 User Study

In addition to the app analysis, we conducted an online survey to uncover the real impact of ticket grabbing apps on users and gain a more comprehensive understanding of the ecosystem.

3.3.1 Survey Instrument

The questionnaire was divided into three parts, and the full questionnaire is provided in the online appendix [22].

- **Usage details.** To get users’ experiences with ticket grabbing apps, we initially asked participants if they had ever contacted scalpers or used mobile ticket grabbing apps. For those who had used ticket grabbing apps, we asked about their usage scenarios and whether they had successfully obtained the desired goods. We also inquired about the channels through which they downloaded such apps and whether they incurred any additional charges during usage.
- **Privacy concerns.** For those who ever used ticket grabbing apps. We inquired about the types of information participants provided to ticket grabbing apps and their privacy concerns regarding the apps. For all the participants, we finally investigated their awareness of the legality of ticket grabbing apps.
- **Demographic collection.** As customary, we collected participants’ demographic information at the end of the questionnaire, including age, gender, level of education, and whether they were employed in computer-related occupations.

3.3.2 Recruitment

To eliminate potential bias in the questionnaire, we conducted a pre-test within our laboratory. We gathered 17 responses and subsequently refined the questionnaire’s content and wording based on their feedback. Afterward, we distributed the questionnaire through social media and got 66 responses. Then we recruited more participants via Wenjuanxing [37], the most popular online survey platform in China. The survey lasted from early September to late September 2023, culminating in 184 responses and the participants were all from China. The average completion time for the questionnaire was 163.66 seconds. Based on the local hourly wage standard, we provided each participant with 3 CNY as a reward. The demographic distribution is summarized in Table 3 in Appendix A.

3.4 Ethic

Although our institution does not have a formal ethics committee like an Institutional Review Board (IRB), we proactively sought approval from organizations that served similar functions in our institution. Specifically, in the process of app collection, we followed a similar approach to previous studies on cybercrime [58, 74, 81] by collaborating with security companies to obtain ticket grabbing apps. For the user study, we followed the standard ethical research procedures [52, 65, 71] to ensure the confidentiality of information provided by participants. We conducted informed consent before the questionnaire and allowed users to quit the questionnaire any time they felt uncomfortable. We did not collect any personal information and the results were saved on the local computer of the researcher.

3.5 Limitations

- **App collection.** All our analyses were conducted exclusively on Chinese ticket grabbing apps. Ticket grabbing tools are also utilized in other countries [34, 35]. However, due to the highly dynamic nature of ticket grabbing apps, collecting such apps from app stores at scale is challenging. Our research has been facilitated by collaborating with a Chinese security company that has extensively collected historical versions of ticket grabbing apps. Nevertheless, China has been widely reported [11, 12, 25] for its extensive usage of ticket grabbing apps. Hence, our analysis of Chinese apps could serve as one representative example to reveal key aspects of this industry. Our preliminary investigation [36] also revealed that some popular automation apps [6, 20, 30, 32] share similar implementation principles with ticket grabbing apps, suggesting our analytical findings could offer insights for them as well.

Additionally, we only focused on Android apps and did not include IOS apps. This is mainly because of the more stringent requirements imposed by the IOS app store, resulting in a limited number of ticket grabbing apps available. Besides, we primarily gathered apps according to the names of apps. Although we expanded the scale of the dataset as much as possible, apps with unconventional names may be overlooked. Similarly, for the selection of benign apps, we strive for best-effort coverage of more apps and enhance sample diversity. However, we need to admit that the dataset we set up cannot cover all ticket grabbing apps and benign apps on the market, but can adequately represent commonly used ones.

- **App analysis.** During the simulated runs, we opted to employ emulators rather than physical devices. It is important to acknowledge that emulator-based simulations may not completely mirror real-world conditions, as certain apps can detect the emulator environment. Consequently, we excluded the apps that did not perform adequately in the emulator during the simulated runs. Nevertheless, considering that some applications require rooted devices to run, emulators can offer this environment without adding extra complexity.

Moreover, when using the Qianxin sandbox for dynamic analysis, due to the limited interaction of the sandbox, the traffic generated and the behavior displayed are limited. However, the sandbox we utilized operates in an out-of-the-box analysis mode, surpassing other in-the-box sandboxes like Cuckoo [9] in terms of analysis capabilities.

- **User study.** Lastly, in the online questionnaire, the results are based on the participants' own reports, which may lead to potential omissions regarding their usage and specific details of ticket grabbing apps. The actual situation can be more serious and widespread than the answers we received.

4 RQ1: From the Perspective of Developers

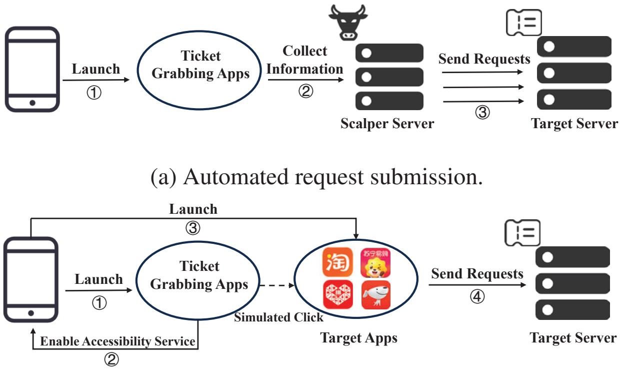
From this section, we systematically addressed our research questions and presented our findings. In this section, we showed the results from the perspective of developers, including the technical characteristics of ticket grabbing apps and the profit model designed by app developers.

4.1 Technical Characteristics

The network traffic patterns generated during the runtime of an application can provide great insights into observing its workflow, especially the backend domains it connected. Based on the sandbox results from 758 apps, we extracted 866 domain names from the Domain Name System (DNS) queries. We further categorized these domains by either accessing them directly or investigating them in search engines. After removing certain noisy domain names (e.g., reverse DNS queries and echo requests), we categorized the accessed domain names into three groups: (1) scalper servers (2) target servers, and (3) third-party services. By combining domain name information and our observation on simulated runs, we have summarized three technical characteristics of the ticket grabbing apps, which are introduced in detail as follows.

4.1.1 Automated Grabbing Modes

The core function of ticket grabbing apps is to automatically snap up the desired goods from target platforms. By integrating the connected domains and service calls made during the simulated run, we identified two distinct automated grabbing modes employed by ticket grabbing apps.



(a) Automated request submission.

(b) Automated simulation click.

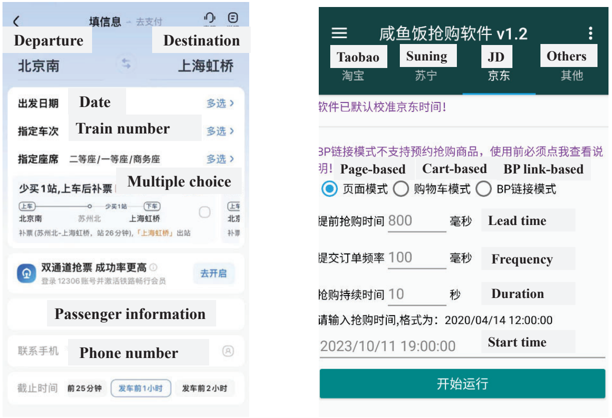
Figure 3: Two grabbing modes of ticket grabbing apps.

- **Automated request submission.** We found several apps that leverage dedicated "scalper servers" for automated grabbing operations. As shown in Figure 3a, instead of directly connecting to the target platforms, the ticket grabbing apps will submit the collected user credentials and order information to scalper servers. These servers are usually cloud-based servers with significantly better performance and concurrency than normal cellphones. Then, scalper servers would repeatedly submit purchase requests to the target server on behalf of the users, acting as an intermediary.
- **Automated simulated click.** This method leverages the Android accessibility service to simulate clicks and directly submit purchase requests to target platforms from the user end. Accessibility service is an inherent Android feature that provides convenience for users with disabilities, especially those with visual impairments. Unfortunately, in recent years, this service has been widely abused in the underground industry [69]. As depicted in Figure 3b, the ticket grabbing apps first prompt users to modify system settings on their devices, i.e., enable the accessibility services. Then, users need to set specific trigger conditions on the ticket grabbing apps, such as the start time of grabbing. Afterward, they have to open the target app and navigate to the desired page. Once the trigger conditions are met, the accessibility service automatically performs user actions like clicking buttons and filling out forms to complete the purchase process. Additionally, Auto.JS Pro is commonly employed in conjunction with accessibility services to accomplish automated tasks through the utilization of automated JavaScript scripts on Android devices.

Based on our simulated runs, we observed that the first mode was adopted by various ticket grabbing apps, including train ticket grabbing apps, concert/sports ticket grabbing apps, and hospital appointment grabbing apps. In contrast, the second mode was primarily utilized by commodity grabbing apps. This is because commodity grabbing apps' target servers are more likely to verify the traffic source to ensure it originates from their own apps. As a result, scalpers face challenges in using the first grabbing mode, prompting commodity grabbing apps to employ accessibility services for automated clicks on users' devices directly.

We then classified the dataset based on the grabbing modes, using whether the accessibility service needs to be enabled as the classification criterion. This can be identified in the static analysis through the *manifest.xml* as apps that require the accessibility service are obligated to register this service in the *manifest.xml*. Utilizing this approach, we identified that 540/758 (71.2%) apps had the accessibility service enabled, falling into the second kind of ticket grabbing apps. Conversely, 218 apps belonged to the first kind.

Clearly, both grabbing modes involve specific security and privacy risks. In the first mode, users are required to provide their credentials and personal information to the untrusted scalper servers, posing privacy concerns. The second mode necessitates users to grant the app extensive permissions, as it enables accessibility services. The apps could then monitor user actions, access current windows, capture entered text, and perform actions like tapping, swiping, and clicking. A more detailed examination of the security and privacy risks of ticket grabbing apps will be provided in Section 5.2.



北京南 ⇆ 上海虹桥

出发日期 多选 ›

指定车次 多选 ›

指定座席 二等座/一等座/商务座 多选 ›

少买1站,上车后补票

上车 北京南 — 苏州北 少买1站 — 下车 上海虹桥 □

补票 (苏州北-上海虹桥，站26分钟)，「上海虹桥」出站

双通道抢票 成功率更高 去开启

登录12306账号并激活铁路畅行会员

联系手机

截止时间 前25分钟 发车前1小时 发车前2小时

咸鱼饭抢购软件 v1.2

淘宝 苏宁 京东 其他

软件已默认校准京东时间!

BP链接模式不支持预约抢购商品，使用前必须点我查看说明!

● 页面模式 ○ 购物车模式 ○ BP链接模式

提前抢购时间 800 毫秒

提交订单频率 100 毫秒

抢购持续时间 10 秒

请输入抢购时间,格式为：2020/04/14 12:00:00

2023/10/11 19:00:00

(a) A train ticket grabbing app. (b) A commodity grabbing app.

Figure 4: The screenshots of different ticket grabbing apps.

4.1.2 Grabbing Tricks

Different target platforms have different business logic, so ticket grabbing apps perform different automated grabbing modes and creatively propose different grabbing tricks to improve the success rate of automated purchasing behaviors. Next, we put together some tricks taken by the ticket grabbing apps to improve the grabbing ability.

- **Integrated grabbing platforms.** For the commodity grabbing apps, the target platforms are diverse, including Taobao (*"taobao.com"*), JD (*"jd.com"*), Suning (*"suning.com"*), and other popular shopping websites in China. To minimize development costs, we found that these apps generally provide integrated support for multiple platforms in one app, as shown in Figure 4b. Based on the results of simulated runs, each commodity grabbing app supports 2.64 shopping platforms on average, with Taobao being the most common. Such integration stems from the diverse e-commerce landscape of commodities in China. In contrast, since "12306" holds an exclusive train ticketing platform in China, train ticket grabbing (as shown in Figure 4a) apps do not need to support multi-platform. Their target domain would only be *"kyfw.12306.cn"*.
- **Time synchronization.** Due to the time delay between the user's devices and the target servers, the start time of grabbing could be delayed compared to the actual release time. Even though the time difference is only a millisecond, the target goods may be grabbed by others just in this time gap [23]. Therefore, some ticket grabbing apps perform time synchronization. Most of them utilize time synchronization APIs provided by the target platforms if available, such as Taobao's

"api.m.taobao.com/rest/api3.do?api=mtop.common.getTimestamp" and Suning's *"quan.suning.com/getSysTime.do"*. Through these APIs, the delay time between the target server can be determined. Based on the results of simulated runs on 50 commodity grabbing apps, 40/50 (80%) offer this feature.

As for the other kinds of apps, since the target platforms like "12306" do not provide such APIs, we found that six apps directly employ the Network Time Protocol (NTP) to obtain Coordinated Universal Time (UTC) by accessing *"2.android.pool.ntp.org"*, instead of aligning with the target server time. Furthermore, two apps were observed accessing *"time.tianqi.com"* to determine the time offset relative to "Beijing Time" for time synchronization.

• **Customized configuration.** For commodity grabbing apps, they allow users to personalize the grabbing start time (even before the release time), duration, and frequency as shown in Figure 4b, all of which contribute to an increased likelihood of successfully getting the target goods. In the case of other apps, users have the option to simultaneously grab tickets for multiple trains as shown in Figure 4a, thereby enhancing their chances of successful booking.

• **Bypass (BP) links.** As shown in Figure 4b, the apps offer various grabbing ways, including cart-based grabbing, page-based grabbing, and bypass (BP) link-based grabbing. The BP link is a URL that includes all the specifications of the product to be purchased. When clicking on the link, it directly takes users to the order submission page, eliminating the need to go through the process of selecting parameters. This minimizes delays caused by extensive interactions and provides users with a more convenient grabbing experience. Each product has its unique BP link, and the apps offer precise guidelines to users for generating the corresponding BP links.

• **High-performance support.** Apart from well-known cloud server providers, we observed that the ticket grabbing apps also utilize services from other cloud service providers to deliver higher performance. For example, "Xiequ Network" (*"xiequ.cn"*) offers millions of high-quality IP addresses and fully anonymous HTTPS proxy services. These features are sought after by ticket grabbing apps to evade detection and enhance grabbing performance. Additionally, certain apps utilize "Hexin Cloud" (*"hexiny.com"*), a group control platform that allows users to control hundreds of cloud machines on one physical device. This enhances the success rate of grabbing through high concurrency.

4.1.3 Third-Party Services

We assessed the dependence of ticket grabbing apps on third-party services according to the domain names in DNS queries. We mainly analyzed the identified app development platforms, image hosting platforms, and third-party SDKs.

• **Development platforms.** Our observations revealed that several app development platforms were used by ticket grabbing apps to simplify their development processes, such as *"appbsl.com"*, *"yimenyun.cn"*, *"dcloud.io"*, *"apps.xiaok1.cn"*, and *"longxinboke.top"*. These platforms provide the fundamental elements for app development, including integrated development environments (IDEs), frameworks, databases, cloud services, and the like. Notably, in addition to normal development support, they also offer paid services to help underground apps succeed. Regardless of whether an app complies with the platform policies and legal requirements, they can assist the developer to successfully release the app on mainstream markets by providing the regulation materials. Some of the development platforms even guarantee a "full refund in case of any issues during the app market release process". Such illicit support has strongly contributed to boosting underground industrial apps like grabbing apps.

• **Image hosting platforms.** Image hosting platforms provide reliable services for image uploading, sharing, and storing. As previously mentioned, some ticket grabbing apps do not have scalper servers, leading them to rely on image hosting platforms. Among the top ten domain names, we recognized four image hosting platforms, namely *"s1.ax1x.com"*, *"s3.bmp.ovh"*, *"images.shejidaren.com"*, and *"pic1.win4000.com"*. In addition, based on our analysis of the app groups, we also noted that some apps may change the image hosting platform due to business adjustments. For example, from *"shejidaren.com"* to *"pic.imgdb.cn"*. Moreover, these image hosting platforms could also support other illegal activities, such as child sexual [57] and phishing emails [28].

• **Software Development Kits (SDKs).** We first utilized LibRadar [19] to identify the third-party libraries within ticket grabbing apps. In total, 4,962 third-party libraries were detected, with 14 of them found in at least 50 apps. Combined with these results, we used the keywords in domain names to match the third-party SDKs within ticket grabbing apps and manually categorized them based on the functionality.

Map-related SDKs are the most commonly used especially in train ticket grabbing apps as they always provide users with a series of additional services related to travel, such as online taxi services. They generally integrate Baidu map SDK (*"api.map.baidu.com"*) and Amap SDK (*"abroad.apilocate.amap.com"*) to provide positioning functions. Another common category of SDKs is used for data statistics, such as Umeng (*"alog.umeng.com"*), Talkingdata (*"tdcv3.talkingdata.net"*), and Tencent Statistics (*"av1.xdrig.com"*). The third category pertains to message push, including Baidu push (*"api.tuisong.baidu.com"*), Xiaomi push (*"register.xmpush.global.xiaomi.com"*), and JPush (*"s.jpush.cn"*). Additionally, some apps integrated customer service SDKs to support online customer service, such as Qiyu (*"qy-swallow.qiyukf.com"*) and Meiqia (*"meiqia.com"*). We also observed SDKs related to payment, such as Alipay (*"openapi.alipay.com"*). This suggests that certain apps integrated online payment functionality, facilitating direct payments within the apps. Moreover, we also found SDKs for single sign-on (SSO) (*"oblog.ctobsnssdk.com"*, *"opencloud-*

wostore.cn") and Tencent Bugly (*"android.bugly.qq.com"*), which offer error monitoring and crash analysis services. These results align with Gao et al.'s [55], highlighting that the ticket grabbing apps abuse numerous third-party SDKs and those SDK providers do not conduct any review before offering the services.

4.2 Profit Model

Rather than profiting through resale, ticket grabbing apps sell users the ability to secure tickets. They directly act on the user end to assist in purchasing. According to our observation, all the apps we collected are free to download. However, these free versions of apps are just used to attract potential users. Developers gain money from users during their usage. Following are two types of profit models.

4.2.1 Paid Versions

Generally, the free version provides a trial opportunity, and if users wish to continue using it, they need to purchase a license key to activate the apps as shown in Figure 7a in Appendix A. Similarly, certain commodity grabbing apps induce users to purchase the latest version of the app through the pathways (e.g., links or chat groups) provided in the free version. Users could receive a key or the latest app upon completing payment, enabling them to begin using the full functionality of the ticket grabbing apps. Based on the domain names requested by the apps, we found the developers primarily utilized two web-based authentication systems, Paojiaoyun (*"paojiaoyun.com"*) and Eydata (*"eydata.net"*), to support the fee-charging feature. We also found that apps could switch from free to paid during version iteration. Taking "com.app.coomc" as an example, versions released after November 2022 have started accessing the "Rukey Network Verification" API (*"api.ruikeyz.com"*) to verify if the users have purchased the paid license keys.

According to our results, 14/50 (28%) of the simulated commodity grabbing apps employed this profit model. We then checked on the pathways provided by these apps and obtained the prices. We found that apps offer various service durations, including daily, monthly, quarterly, and yearly. Among these, daily service cater to short-term users, with an average price of 28.5 CNY. Yearly services are the most expensive, ranging from 50 CNY to 2,458 CNY, with an average price of 494.1 CNY. The prices are set by the app developers and may be related to the difficulty of grabbing target commodities.

4.2.2 Paid Services

Additionally, ticket grabbing apps provide some paid additional services in the free versions, such as the acceleration packages in train ticket grabbing apps as shown in Figure 7b in Appendix A. By buying this service, users can enhance the success rate of ticket purchases as claimed. In the online questionnaire, 38.82% of participants stated that they have made additional payments during the usage of ticket grabbing apps (Q9), and they are more likely to pay when using train tickets grabbing apps. These payments were not for acquiring the apps but rather for the mentioned paid service.

According to our observations on simulated runs, apps offer varying prices corresponding to different capabilities, ranging from 10-80 CNY. In order to attract more users to purchase paid services, ticket grabbing apps commonly employ persuasive tactics on the payment page to instill anxiety in users. They lead users to believe that only by paying can they successfully secure tickets, with a higher payment equating to a greater chance of success. Examples of such persuasive statements include "It's highly difficult to grab tickets at low-tier (free) services", "52% of users have paid for the top-tier(highest price) grabbing services", etc. Besides, to increase revenue, ticket grabbing apps also provide other paid services, such as the use of high-speed CDN servers (49 CNY), manual ticket grabbing services (20-40 CNY), and compensation services for unsuccessful grabs (60 CNY). These services can be used in combination, and users can spend up to 229 CNY to obtain the highest guarantee for ticket grabbing.

4.2.3 Effectiveness of Paid Features

Additional payments are purported to increase the likelihood of securing tickets. However, from the user end, it's hard to assess the effectiveness of a "probabilistic" service. Users could only observe two outcomes: "grabbed", or "missed", without insight into whether the probability has indeed increased. Given the difficulty of evaluating the effectiveness systematically, we conducted a preliminary case study.

We targeted one train with no available tickets, which is the most common scenario for using grabbing apps. In this scenario, the apps persistently send purchase requests to target servers, monitoring ticket availability and conducting the grabbing operations. We randomly selected 5 train ticket grabbing apps that offered paid services and purchased various acceleration packages ranging from 10 CNY to 80 CNY. We also established a control group to purchase the same train ticket directly from the official platform. Note that despite the official platform displaying no available tickets, it offers a wait-listing feature to provide tickets to users who join the list when others are returned. According to our tests, none of the apps succeeded in grabbing the designated tickets, even the one with the top-tier acceleration packages (80 CNY). However, the control group, which directly queued on the official platform, successfully purchased the corresponding tickets. In other words, at least for this case study, the paid service failed to provide any advantage or achieve the desired "acceleration". The official platform also explicitly discouraged users from using ticket grabbing apps and recommended its free waiting list mechanism. Nonetheless, there were still users who opted for paid ticket grabbing services due to their urgent need for tickets, as we revealed in Section 5.1.

Answers to RQ1: We found that ticket grabbing apps are highly automated. Some apps use scalper servers as intermediaries to automatically send purchase requests, while others abuse Android accessibility services to directly achieve automated clicks. There are several grabbing tricks used by the apps to improve the success rate, such as using time synchronization and BP links. We revealed that the apps may rely on a large number of third-party services to support their functions and bypass the censorship. In addition, we found that app developers could make money through the usage and they only use the free version as a way to attract potential users.

5 RQ2: From the Perspective of Users

In this section, we first presented the popularity and usage details of ticket grabbing apps. Then, we delved into the prevalent risk behaviors associated with these apps and investigated users' privacy concerns.

5.1 Usage of Ticket Grabbing Apps

In the online survey, we first asked participants whether they have used any ticket grabbing apps or contacted scalpers for ticket purchases. 46.2% of the participants indicated that they had ever used ticket grabbing apps, while 26.63% of the participants resorted to direct contact with scalpers. Ticket grabbing apps, due to their ease of access and user-friendly operation, tend to be preferred by users. For participants who did not use ticket grabbing apps or contact scalpers, we further inquired about their future intentions. Among them, 55.13% (43/78) of the participants expressed the possibility of using ticket grabbing apps or engaging with scalpers for future purchases. This indicates that scalping behavior is quite common in practice and has a considerable number of potential users.

For participants who had previously used ticket grabbing apps, we additionally inquired about some usage details.

• **Usage scenarios.** 89.41% of the participants used ticket grabbing apps for purchasing train tickets, while 42.53% used them for concert or sports tickets. This is probably attributed to the Spring Festival travel rush in China, involving a significant number of individuals engaging in ticket purchases.

• **App sources.** According to participants' responses, 83.35% of them obtained the ticket grabbing apps from popular app stores provided by major smartphone manufacturers, e.g., Xiaomi App Store [38], Huawei AppGallery [16]. This indicates that ticket grabbing apps are currently disseminated through mainstream app markets and these app markets do not effectively detect or intervene in the ticket grabbing apps. Meanwhile, a considerable number of participants acquire them through unregulated channels. 12.94% of the participants directly downloaded it through search engines, and 21.18% of the participants found the apps on social media. The security of these apps has not undergone any scrutiny either.

• **Success rate.** Although ticket grabbing apps utilize technical methods to surpass original customers in obtaining the hot resources, they do not guarantee a 100% success rate. In the survey, only 10.59% of the participants reported being successful in each grabbing attempt. The majority of participants (81.18%) indicated a mix of successful and unsuccessful attempts, and even 8.24% of the participants had not succeeded in obtaining the target goods despite using the ticket grabbing apps. Moreover, 90.91% of the participants reported that they were still unable to successfully grab tickets even after payment. Through the Chi-square test [84], we found that participants who paid fees during the ticket grabbing process did not exhibit a higher success rate ($p > 0.05$). This indicates that the paid services offered by ticket grabbing apps may not improve the success rate as claimed, aligning with the results of our analyses detailed in Section 4.2.2.

5.2 Security and Privacy Risks

Similar to malware, ticket grabbing apps may introduce uncertain security and privacy risks as well. Based on the results from the sandbox, 89/758 (11.74%) apps were identified as malicious. Combining the results of static analysis, we identified five typical security risks and other malicious behaviors.

• **Abuse of dangerous permission.** With the help of AndroGuard [3], we found each app requests 83.7 permissions on average, including 9.4 dangerous permissions. Figure 5 shows the requested dangerous permissions. Compared to the analysis results of mobile gambling apps [62], ticket grabbing apps request more dangerous permissions, especially on SMS-related permissions, indicating greater security risks. That is probably because ticket grabbing apps need to read the SMS code sent to the phone to complete the purchase. Besides the necessary permissions for ticket grabbing functionality, dangerous permissions like CAMERA, CALL_PHONE, and READ_CONTACT are also obtained by over 50% ticket-grabbing apps, posing serious privacy risks to users [10]. For instance, granting CAMERA permission can lead to the capture of unauthorized images or videos once misused.

In addition to the dangerous permissions, 685/758 (90.4%) of the apps request the android.permission.WAKE_LOCK permission, which could control the wake lock of the device and prevent it from entering sleep mode. By requesting this permission, ticket grabbing apps can continuously perform grabbing process without interruption, potentially resulting in battery drain and adverse effects on device performance.

• **Get root.** In simulated runs, the emulator we used was rooted. Throughout the experiment, we observed that some apps requested root privileges, gaining the highest level of control over the device. Based on the results from the sandbox, 577/758 (76.1%) of the apps applied for root privileges. In contrast, as shown in Figure 8 in Appendix A, among the

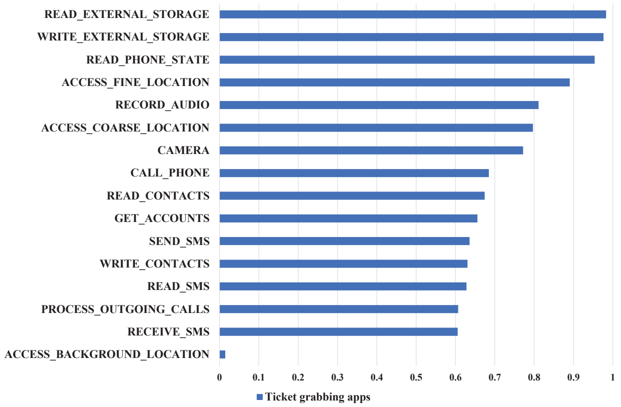


Figure 5: The percentage of dangerous permissions requested by ticket grabbing apps.

benign apps we selected, only 18/100 (18.0%) requested root privileges. These apps are all utility-type applications, such as "360 Mobile Security", "ZArchiver", "Sohu Input Method", etc. They request root permissions to access and manipulate system files stored on the device, as well as to optimize device settings by optimizing startup items. In the case of ticket grabbing apps, using root access could activate accessibility services without user confirmation. However, gaining root can also lead to other sensitive activities beyond the claimed scope and without any precautions.

Furthermore, we also observed that some apps contain "rikka.shizuku.ShizukuProvider". Shizuku [29] is an open-source project that allows applications to access sensitive system-level operations on the device without requiring root privileges. This presents a similarly high-security risk as root privileges. Our data reveals that 446/758 (58.8%) apps utilize Shizuku, while no instances were found in benign apps.

• **Insecure configuration.** We found that lots of ticket grabbing apps had insecure configurations. Firstly, our results revealed that 161 apps (21.2%) set "allowBackup" to "true", allowing backup of the app data to the cloud or local storage. While it is beneficial for data restoration across different devices, it poses a disadvantage for safeguarding sensitive data as user information can be at risk of leaking to untrusted locations. Secondly, 596 (78.6%) apps have configured "networkSecurityConfig" to "none", which means the apps will ignore all network security configurations, such as certificate validation and other network security measures. This would allow the apps to establish connections with insecure or untrusted servers and potentially expose the apps to "man-in-the-middle" attacks [75]. Thirdly, 105 (13.9%) apps set "requestLegacyExternalStorage" to "true" to access the external storage (i.e., "/sdcard"). However, ticket grabbing apps typically do not need to require access to external storage. This configuration could cause unnecessary access to files stored by users and provide the necessary attacking conditions for the "man-in-the-disk" attack [21].

• **Lack of privacy policy.** According to legal requirements [8, 15,26], privacy policy is required to clearly inform users about the types and purposes of collected information. Additionally, the use of third-party SDKs should be disclosed in the privacy policy, as these SDKs may also collect users' personal information. However, based on observations during the simulated runs, we found that some ticket grabbing apps do not have a dedicated privacy policy and fail to transparently disclose the information collection to users. To further validate the presence of privacy policy in ticket grabbing apps, we filtered the URL strings obtained from the static analysis of *strings.xml* and counted the URLs containing "privacy". According to the statistics, only 57/758 (7.5%) apps have privacy policies.

• **Information collection.** While most ticket grabbing apps lack privacy policies, it doesn't imply that they do not collect any information. During the simulated runs, we found that some apps require users to provide the login credentials of the target platforms (e.g., usernames and passwords) to facilitate the ticket grabbing process. These credentials are then sent to the scalper server, posing significant security risks. If the information is leaked or shared, it could potentially compromise the user's passwords on other platforms [64]. Moreover, to enhance the speed of ticket grabbing, 18/50 (36.0%) of the simulated commodity grabbing apps implement the "quick payment" feature, which requires the user to input their payment passwords in advance within the ticket grabbing app. Upon successfully getting the target goods, the apps can directly use these passwords to complete the transaction on target platforms without further user actions.

• **Other malicious behaviors.** In addition to the representative security risks, we also uncovered certain malicious behaviors that are not widespread but occur in a few apps. For instance, we found two apps that persistently captured screenshots and uploaded them to remote servers. They use the MediaProjection API provided by Android for implementation. User's operations on other apps would also be obtained by the ticket grabbing app making the app transformed into spyware. Furthermore, the sandbox analysis revealed that three apps monitor the incoming SMS messages, corresponding to the requests for dangerous SMS-related permissions (e.g., SEND_SMS and READ_SMS). The apps use "SELECT _id, address, person, body FROM content://sms" to query all the message information on the mobile phone, which is a serious violation of the user's privacy.

5.3 Users' Privacy Concerns

In the online survey, we asked all participants about their perception of the legality of ticket grabbing apps. 39.13% of them believed that such apps are illegal, while 8.7% held the opposite attitude. 38.59% of the participants thought that the legality of ticket grabbing apps is not absolute. Among participants who had previously used ticket grabbing apps, 64.71% of them had shared their personal information with these apps. We further asked these participants to specify the types of personal information provided. Nearly all of them mentioned the identification number and mobile phone number, both of which are considered sensitive information.

Subsequently, we employed the 5-point Likert Scale method [44] to assess the level of users' privacy concerns. The results revealed an average score of 3.67/5.0, indicating that the majority (70.59%) of participants have privacy concerns regarding ticket grabbing apps. Specifically, 58/60 (96.67%) of the participants expressed concerns about the potential leakage or misuse of their provided personal information. Similarly, for participants who had not used the ticket grabbing apps, this is also the main reason why they chose not to use the apps, cited by 29/78 (37.18%) of the participants.

In summary, through the comprehensive analysis of our online survey, we can observe no matter whether participants have used ticket grabbing apps or not, over half (70.59%) of the participants were aware of the security risks of scalping (Q13). However, due to users' urgent need for the tickets, 84.5% of the participants still opt to use ticket grabbing apps despite their privacy concerns (Q1). They actively sought out ways to acquire the efficient apps through various channels (Q8) and were willing to pay additional fees to enhance the success rate (Q9). They also overlooked the collection and use of personal information by the ticket grabbing apps, providing their personal information on their own initiative (Q12).

Answers to RQ2: We found that ticket grabbing apps have a significant user base, with train ticket grabbing apps being the most prevalent in practice. Ticket grabbing apps request more dangerous permissions than mobile gambling apps and have several security and privacy risks. We revealed that more than half of the participants were aware of the privacy risks posed by ticket grabbing apps. Nevertheless, due to the pressing need for these apps, the majority of users continue to utilize them and willingly provide personal information during usage, as they prioritize tickets over privacy.

6 RQ3: From the Perspective of Platforms

In this section, we analyzed the gang characteristics and bypass tricks of the ticket grabbing apps, which can help target platforms to better identify and detect this kind of apps.

6.1 Gang Analysis

Through the gang analysis, we tried to reveal the clustering characteristics of ticket grabbing apps. Additionally, our analysis showed the communication channel and provided valuable intelligence about the exposed gangs.

6.1.1 Clustering Characteristics

In the app analysis, we obviously found that some different ticket grabbing apps exhibited a similar UI design. In order to further verify the prevalence of this phenomenon and identify the clustering characteristics among these apps, we employed FSquaDRA [14] to calculate code similarity between the apps. The results revealed 14 distinct groups, encompassing a total of 205 apps. Each group consisted of apps with a similarity exceeding 90%, indicating that these apps share similar purposes and usage scenarios. These apps are basically commodity grabbing apps, showing a development trend of the apps. Additionally, we found different ticket grabbing apps with the same signature often exhibit similar appearances and higher code similarity. Among the apps with the same signature, 24.7% of them demonstrated a similarity over 90%.

In order to figure out more gangs characteristics, we used the signature-based grouping method to group our dataset. As is well-known, the signature of an app generally contains the developer's identity information, such as the developer's name, organization, and location. We used AndroGuard [3] to obtain the signature and extract the developer's identity information, as apps with the same developers can be considered from the same gang. According to the grouping results, 37 apps (4.9%) were not signed jar files, indicating that they were potentially risky. Of the remaining 721 apps, there are 269 signatures, containing 167 developer's identity information.

Next, we tried to do gang clustering by the developer information in the signatures. However, we observed that some signatures used by certain groups do not contain any developer's identity information. (1) Some groups use default signatures provided by the IDEs, such as "emailAddress=android@android.com, CN=Android, OU=Android, O=Android, L=Mountain View, ST=California, C=US" and "CN=Default, O=Auto.js Pro". (2) Additionally, there are cases where the groups use custom signatures, but the signature is not in compliance. For example, "CN=inject.keystore, OU=inject, O=inject, L=inject, ST=inject, C=inject". Each parameter of the signature is an invalid placeholder and does not include any developer information. This situation is quite common, which suggests a tendency among ticket grabbing apps to conceal their developer identity. (3) Furthermore, we found that some groups use the signature "CN=Android Debug, O=Android, C=US". It is an Android debug signature, indicating that the app is just a debug version. This type of signature is usually not used for official release on Google Play or other app distribution channels as it lacks the security required for formal release. The above illustrated that ticket grabbing apps tend to hide the developer information in the signature. It is challenging to trace the gangs of ticket grabbing apps through signatures.

6.1.2 Communication Channel

During the simulated runs, we observed that some ticket grabbing apps always set a chat group encouraging users to join. These groups have been used as a more covert app distribution and could help scalpers facilitate long-term customer relationships. They solve users' problems during usage and provide the paid versions of ticket grabbing apps. Therefore, we can use different chat channels as the identifier to distinguish different gangs, as unrelated developers are unlikely to gather in the same channel for user communication. In total, we identified 20 chat groups during the simulated runs and utilized these chat groups to gain insights into the exposed gangs of ticket grabbing apps.

By inputting the numbers of chat groups, we could retrieve the group names and member counts without joining the groups. We discovered that these group names usually have numerical values at the end, serving as an index to represent a series of related sub-groups. The highest number encountered was "83" indicating the possibility of another 82 sub-groups. Interestingly, sensitive keywords such as "ticket grabbing" were generally avoided in the group names. Instead, they used terms like "information exchange" or "pre-sales communication". Some group names appeared entirely unrelated, such as "salted fish security team", "fangfang's learning welfare", or "octopus brother's pink pillow", likely to evade detection by the chat platform. Additionally, we collected the number of members in each group, which averaged 919.71 individuals per group. Seven groups had more than 1,000 members, indicating a significant user base of ticket grabbing apps.

In Section 4.2, we explored the profit model and individual payment amounts of ticket grabbing apps. However, to understand the whole profitability scale, we also need to know the amount of paying users, which is hard to ascertain in this study. Nonetheless, the number of chat group members can serve as a conservative estimate of the paying user base. Since the aim for users to join the chat group is to gain paid versions, its membership size is reasonable for estimation. Moreover, considering that there may be other paid channels other than the chat groups we identified, this number should be viewed as a minimum estimation. Thus, using the average number of users per chat group (919.7 users) and the average fee amount (494.1 CNY), we estimate that a single ticket grabbing app could make an average annual profit of at least 454,429 CNY.

6.2 Tension Between Platforms and Apps

Ticket grabbing apps could undermine the profit model of target platforms and harm their reputation. Consequently, platforms are motivated to protect their users and maintain a fair ecosystem. Meanwhile, developers of ticket grabbing apps persistently seek ways to evade detection measures, resulting in an ongoing tension between platforms and the apps.

6.2.1 Detection and Interception on Target Platforms

The target platforms constantly improve their detection capabilities, striving to intercept the relevant traffic originating from abnormal users. Most target platforms deploy a cost-effective solution known as the "Completely Automated Public Turing test to tell Computers and Humans Apart" (CAPTCHA) to differentiate between humans and machines, thereby preventing ticket grabbing apps. Furthermore, target platforms may employ more advanced detection systems for identifying the machine traffic. For example, through the deployment of the MTEE system [2], Taobao has repelled 18.87 million scalpers' behaviors and protected 800 million purchase orders from normal customers. Generally, the detection system employs machine learning techniques to identify machine traffic synthetically, including the device environment, account profile, and behavioral patterns of the traffic [4].

6.2.2 Bypass Tricks of Ticket Grabbing Apps

In the face of the target platform's continuously evolving detection methods, the ticket grabbing apps also improve technical implementation to bypass the detection of the platform. To determine the update intervals across various versions, we extracted the compiled time from *manifest.xml* of apps with different versions and presented the update timestamps in Figure 6 in Appendix A. On average, the update cycle for each app is approximately 41.73 days. Based on the version update notes within a few apps, we determined that the version iterations primarily focused on 1) increasing the supported grabbing platforms, 2) optimizing functionalities (e.g., adding "quick payment"), and 3) updating circumvention measures (e.g., implementing automatic IP address switching).

As mentioned, ticket grabbing apps updated at a rapid pace to counter the detection mechanisms. When faced with CAPTCHA challenges, ticket grabbing apps resort to CAPTCHA solving services to bypass the verification. These CAPTCHA solving services employ crowdsourcing or machine learning to recognise the CAPTCHA in very few seconds and offer convenient APIs for app developers to integrate into their apps. Generally, these services are not offered for free. The ticket grabbing apps necessitate users to purchase the CAPTCHA solving services in advance, and then require them to enter their service credentials into the ticket grabbing apps to enable the utilization of these CAPTCHA solving services within the apps. In our dataset, we identified three CAPTCHA solving services supported by the ticket grabbing apps, including "TTshitu" (*"ttshitu.com"*), "Fateadm" (*"fateadm.com"*), and "Ruokuai" (*"ruokuai.com"*).

To against the detection system deployed by the target platforms, the core idea behind the bypass methods is to closely replicate the environment and behavior of genuine human users. Those detection systems are primarily designed for large-scale machine traffic, focusing on detecting organized scalpers involved in bulk purchasing activities. However, ticket grabbing apps are mainly used by the original customer, which generates a smaller scale of traffic. These apps run on users' mobile devices and use their personal accounts for ticket purchases. Therefore, it becomes challenging to detect the ticket grabbing app users through the device and account dimensions. Additionally, in order to mimic user behavior as much as possible, some ticket grabbing apps use accessibility services to simulate the click actions of users as mentioned in Section 4.1.1. The traffic generated by this approach closely resembles human-generated traffic, making it difficult to distinguish from the traffic dimension.

Additional bypass tricks were also discovered during the app analysis and we already revealed them. For example, ticket grabbing apps employ multiple IP proxies to avoid being intercepted and leverage development platforms to bypass the content review when releasing on the app markets.

Answers to RQ3: We discovered that some ticket grabbing apps have similar UI designs and the apps intend to hide the developer's identity in the signature. Ticket grabbing apps use social media as a covert communication and distribution channel. Even though the target platform has already deployed robust detection systems, ticket grabbing apps are still active in the app markets. They take advantage of ever-improving bypass tricks, such as CAPTCHA solving services, accessibility services, multiple IP proxies, and so on.

7 Discussion

When faced with tickets and privacy, similar to users' choices on the loan apps in Kenya [72], users prioritize privacy at a significantly lower level of consideration. Due to the significant supply-demand disparity, the urgency of securing tickets surpasses concerns about channel legitimacy. Even though ticket grabbing apps offer no guarantees of success rate or information processing, users willingly assume responsibility for the risks tied to their authorized proxy actions. Many public announcements urging consumers to use official channels for high-demand purchases, however, their impact seems limited. Nevertheless, it is still necessary to continue strengthening user education and ensure they are fully aware of the security and privacy risks tied to ticket grabbing apps.

Besides, one of the most efficient strategies to deter users from resorting to ticket grabbing apps is to impede the development and distribution of such apps. According to the analysis, ticket grabbing apps rely on a series of tools to realize the grabbing function and bypass the detection system. Therefore, those tools serve as the bottleneck of ticket grabbing apps. We proposed an "Indirect Combat" approach for the interception of ticket grabbing apps. The core idea is to impose restrictions on these bottleneck, which can indirectly disrupt the ticket scalping ecosystem and promote fairer access to tickets for genuine customers as a result. Table 1 shows auxiliary tools used by ticket grabbing apps. In particular, we propose the following three practical suggestions.

- **Strengthen the review of service recipients.** We found that ticket grabbing apps may abuse some legitimate services such as accessibility services and third-party SDKs. These service providers should conduct a more comprehensive review of the purpose of the objects to be served, to avoid providing services to malicious applications and becoming accomplices. This suggestion is practicable as Auto.JS Pro has currently implemented business compliance rectification measures since 12 February 2023. They conducted code reviews and mandated real-name authentication for developers [5]. Similarly, those third-party SDK providers should also introduce measures (e.g., pre-service disclosure and review mechanism), akin to the actions taken by Auto.JS Pro, to ensure that services are not abused for illegal purposes.
- **Prohibit services catering to illegal industry.** As shown in Table 1, ticket grabbing apps employ a range of support services, including CAPTCHA solving services, development platforms, image hosting services, and specific cloud servers. Typically, most of these services cater to malicious apps and are not employed by benign apps. Hence, detecting these services can be an effective means of identifying ticket grabbing apps. Specifically, a unified reporting interface is required for users to report services found to be used in the illegal industry. Prohibiting these support services through strong national measures can potentially stifle the development of ticket grabbing apps at their core.
- **Enhance the review on the app release.** Based on the user surveys, it is evident that ticket grabbing apps primarily originate from mainstream app markets, suggesting a lack of specialized categorization and review processes for these apps. In order to deter the release of apps using fraudulent credentials provided by the development platforms, it is essential to strengthen the review process. Our publicly available dataset and surveys on ticket grabbing apps can serve as a reference for app markets in detecting similar applications. Firstly, by verifying developer information and UI design, app markets can reduce the likelihood of ticket grabbing apps entering their platforms. This is because most of these apps tend to conceal their developer information and have typically similar interfaces. In addition, analyzing app composition, including indicative features like Android accessibility services and Shizuku, can aid in identifying ticket grabbing apps. Overall, app markets should perform a better investigation and app analysis to block such illegal apps, thereby maintaining a healthy and trustworthy app ecosystem for users.

8 Related work

8.1 Analysis of Ticket Bots and Ticket Systems

Before the popularity of ticket grabbing apps, ticket bots served as a focal point for studying scalpers' behavior. Many researchers focused on scalping and ticket bots from the perspectives of economic and legal [40, 59, 60, 83, 87]. They highlighted the disruptive impact of ticket bots on the regular market order. As for the technical analysis, extensive research has been dedicated to comprehending the behavior of these ticket bots. For example, Nissan et al. [73] gave a high-level

Table 1: The auxiliary tools used by ticket grabbing apps and their purposes.

Type	Auxiliary tools	Purpose
Ubiquitous and legal	Third-parity SDKs	Provide additional features such as data analysis, payment, message push, etc.
	Accessibility services	Support simulated click.
	Auto.JS Pro	Create the script for automatic simulated click.
	Time synchronization APIs	Get the timestamp of the target server to reduce the time delay.
Catering to illegal industry	Development platforms	Simplify the development process and provide app releasing services.
	Image hosting platforms	Host image resources and realize image sharing.
	Multiple IP proxy services	Provide multiple IPs to prevent detection.
	Group control platforms	Control multiple devices on a single physical device.
	CAPTCHA solving services	Bypass the CAPTCHA challenges on target platforms.

analysis of a ticket-check-bot, and Lin et al. [68] introduced the behavior of ticket bots through a real case solved by Criminal Investigation Bureau (CIB). Accordingly, it is important to detect the bots by analyzing their characteristics. Rahman et al. [78] proposed a forensic framework to verify if the crime is done using automated bots and whether the bots are good or bad. Xie et al. [89] used the event logs of the mobile system and Wu et al. [88] proposed a new integrated method based on network traffic characteristics to identify the ticket bots.

In addition to the analysis of ticket bots, some work tried to make changes to ticket systems. Yang et al. [90] proposed a safe, usable, and easy-to-deploy ticket system to prevent scalpers and restore fair competition. Feulner et al. [53] demonstrate that the self-sovereign identity based ticket system can enable efficient secondary market control. Moreover, Elefant et al. [51] discussed the possibility of moving the ticket system onto an open source blockchain. Additionally, Isaksson et al. [63] reviewed the potential benefits blockchain applications may have in the ticket market, including the prevention of ticket bots and fake tickets. Nevertheless, such ticket systems are unable to completely prevent the presence of ticket grabbing apps. Within our dataset, we identified 147 apps with the ".nft" suffix. These apps are specifically designed for making purchases on NFT marketplaces, which operate on blockchain technology.

8.2 Analysis of Applications

App analysis is a prevalent research method used to comprehend app functionality and identify security issues. A significant amount of work focused on analyzing the reviews of apps in the app market. By performing sentiment analysis on these reviews [76], bad reviews can be identified [85], and useful feedback can be provided to the developers [48]. In addition, emerging issues such as new bugs can be detected timely by analyzing the reviews [54]. As for the analysis of security, static analysis is the frequently used method and there are many high-performance tools proposed by researchers to recognize the dangerous behaviors of the apps. For example, COVERT [42] for permission leakage, CHEX [70] for component hijacking vulnerabilities, AspectDroid for possible unwanted activities [39], and ApkCombiner [67] for information leakage on inter-component communication. Klieber et al. [66] and Ravitch et al. [79] gave an information flow analysis that can show potentially dangerous data flows.

There is also research that analyzed specific types of apps, particularly focusing on health-related apps [43, 49, 82, 86] and education-related apps [46, 61, 80]. Likewise, some work analyzed the ecosystem of specific underground industries through the apps. For instance, Hong et al. [62] and Gao et al. [55] analyzed the mobile gambling scam ecosystem, identifying strategies to disrupt such fraudulent activities. Chatterjee et al. [47] provided the first in-depth study of the intimate partner surveillance (IPS) spyware ecosystem and Bivens et al. [45] analyzed the features present in anti-rape apps to prevent sexual violence. Our research shares similar goals with these studies, focusing on a novel scenario involving mobile ticket grabbing apps, which has not been previously explored.

9 Conclusion

With the promotion of real-name ticketing systems, ticket grabbing apps have become a popular pattern for scalpers, especially in China. Our work takes the first step toward revealing the ecosystem and building up the large-scale dataset of ticket grabbing apps in the wild. According to the results of app analysis on 3,121 apps and our online survey on 184 participants, we revealed the technical and social characteristics of ticket grabbing apps. In addition, we highlighted that ticket grabbing apps have a number of security and privacy risks. Although the majority of users have privacy concerns about ticket grabbing apps, they still use the apps and provide them with their sensitive information. Finally, we summarized the auxiliary tools used by ticket grabbing apps and proposed an "Indirect Combat" approach. In summary, our work brings new insight into the mobile ticket grabbing ecosystem, which can help better detect and intercept such scalping behaviors.

Acknowledgments

We would like to thank our shepherd and the anonymous reviewers for their insightful comments. Special thanks to all the people who took part in our study. This work is supported by the National Key Research and Development Program of China (No. 2023YFC3321303), the National Natural Science Foundation of China (62102218, 62302258), CCF-Tencent Rhino-Bird Young Faculty Open Research Fund (CCF-Tencent RAGR20230116). Haixin Duan is supported by the Taishan Scholars Program. Yiming Zhang is partially supported by the Shuimu Tsinghua Scholar Program.

References

[1] 2023 imperva bad bot report. https://www.imperva.com/resources/resource-library/reports/2023-imperva-bad-bot-report/. 2023-9.

[2] Ali risk control: Exploration and practice of big data application. https://help.aliyun.com/document_detail/157938.html. 2023-9.

[3] Androguard. https://github.com/androguard/androguard. 2023-9.

[4] Anti-cheat service. https://dun.163.com/locale/en. 2023-9.

[5] Auto.js pro. https://pro.autojs.org/. 2023-9.

[6] Automate. https://play.google.com/store/apps/details?id=com.llamalab.automate. 2024-2.

[7] Better online ticket sales act. https://www.ftc.gov/legal-library/browse/statutes/better-online-ticket-sales-act. 2023-9.

[8] California consumer privacy act. https://oag.ca.gov/privacy/ccpa. 2023-9.

[9] Cuckoo sandbox. https://cuckoosandbox. 2023-9.

[10] Dangerous permissions detected in top android health apps. https://cybernews.com/security/dangerous-permissions-android-health-apps/. 2023-11.

[11] Does the “magic tool for grabbing tickets” work? https://www.life-china.cn/2023/09/22/does-the-magic-tool-for-grabbing-tickets-work/. 2024-2.

[12] Does the “ticket grabbing tool”’ really work? https://www.lwxsd.com/pcen/info_view.php?tab=mynews&VID=43063. 2024-2.

[13] Fiddler. https://www.telerik.com/fiddler. 2023-9.

[14] Fsquadra. https://github.com/zyrikby/FSquaDRA. 2023-9.

[15] General data protection regulation. https://gdpr-info.eu. 2023-9.

[16] Huawei app store. https://consumer.huawei.com/cn/mobileservices/appgallery. 2023-9.

[17] Law to regulate ticket resales enacted. https://www.loc.gov/item/global-legal-monitor/2019-01-24/japan-law-to-regulate-ticket-resales-enacted/. 2023-9.

[18] Ldplayer-lightweight and fast android emulator for pc. https://www.ldplayer.net/. 2023-9.

[19] Libradar: A detecting tool for 3rd-party libraries. https://github.com/pkumza/LibRadar. 2023-9.

[20] Macrodroid - device automation. https://play.google.com/store/apps/details?id=com.arlosoft.macrodroid. 2024-2.

[21] Man-in-the-disk: A new attack surface for android apps. https://blog.checkpoint.com/security/man-in-the-disk-a-new-attack-surface-for-android-apps/. 2018-8.

[22] Online appendix. https://sites.google.com/view/ticket-grabbing-apps/. 2024-4.

[23] Order procrastination: How to avoid delays in purchasing inventory. https://www.sostocked.com/purchasing-inventory/. 2023-9.

[24] Owsap: Explore the world of cyber security. https://owasp.org/. 2023-9.

[25] People’s commentary: beware of the risks behind ticket grabbing software. https://equalocean.com/briefing/20220127230116027. 2024-2.

[26] Personal information protection law. https://personalinformationprotectionlaw.com/. 2023-9.

[27] Qianxin sandbox. https://sandbox.qianxin.com/tq/. 2023-9.

[28] Remote images are pushing email filters to their limits. https://www.vadesecure.com/en/blog/remote-images-are-pushing-email-filters-to-their-limits. 2023-9.

[29] Shizuku: Let your app use system apis directly. https://shizuku.rikka.app/. 2023-9.

[30] Smart autoclicker. https://play.google.com/store/apps/details?id=com.buzbuz.smartautoclicker. 2024-2.

[31] Spring festival comes, train tickets is difficult to grab. http://news.bandao.cn/news_html/201301/20130118/news_20130118_2063020.shtml. 2023-9.

[32] Tasker. https://play.google.com/store/apps/details?id=net.dinglisch.android.taskerm. 2024-2.

[33] Ticket sales act. https://www.ontario.ca/laws/statute/17t33. 2023-9.

[34] Ticketbots. https://ticketbots.net/. 2024-2.

[35] Ticketmaster bot, pit tickets bot, concert ticket bot, football bot. https://kwork.com/chatbots/25940456/ticketmaster-bot-pit-tickets-bot-concert-ticket-bot-football-bot. 2024-2.

[36] Want apps which use accessibility services to improve android experience. https://www.reddit.com/r/androidapps/comments/15ro6uw/want_apps_which_use_accessibility_services_to/. 2024-2.

[37] Wenjuanxing. https://wjx.cn. 2023-9.

[38] Xiaomi app store. https://app.mi.com. 2023-9.

[39] Aisha Ali-Gombe, Irfan Ahmed, Golden G Richard III, and Vassil Roussev. Aspectdroid: Android app analysis system. In *Proceedings of the Sixth ACM Conference on Data and Application Security and Privacy*, pages 145–147, 2016.

[40] James J Atkinson. The economics of ticket scalping. Technical report, mimeo, 2004.

[41] Nevra Azerkan. Sold out: Why the music industry needs to urge lawmakers to regulate how concert tickets are distributed. *Whittier L. Rev.*, 38:130, 2017.

[42] Hamid Bagheri, Alireza Sadeghi, Joshua Garcia, and Sam Malek. Covert: Compositional analysis of android inter-app permission leakage. *IEEE transactions on Software Engineering*, 41(9):866–886, 2015.

[43] David Bakker, Nikolaos Kazantzis, Debra Rickwood, Nikki Rickard, et al. Mental health smartphone apps: review and evidence-based recommendations for future developments. *JMIR mental health*, 3(1):e4984, 2016.

[44] Dane Bertram. Likert scales. *Retrieved November*, 2(10):1–10, 2007.

[45] Rena Bivens and Amy Adele Hasinoff. Rape: is there an app for that? an empirical analysis of the features of anti-rape apps. *Information, Communication & Society*, 21(8):1050–1067, 2018.

[46] Melissa N Callaghan and Stephanie M Reich. Are educational preschool apps designed to teach? an analysis of the app market. *Learning, Media and Technology*, 43(3):280–293, 2018.

[47] Rahul Chatterjee, Periwinkle Doerfler, Hadas Orgad, Sam Havron, Jackeline Palmer, Diana Freed, Karen Levy, Nicola Dell, Damon McCoy, and Thomas Ristenpart. The spyware used in intimate partner violence. In *2018 IEEE Symposium on Security and Privacy (SP)*, pages 441–458. IEEE, 2018.

[48] Wenhui Chen, Yao Yao, Tianyu Chen, Wei Shen, Sheng Tang, and Hian Kee Lee. Application of smartphone-based spectroscopy to biosample analysis: A review. *Biosensors and Bioelectronics*, 172:112788, 2021.

[49] Roberto Collado-Borrell, Vicente Escudero-Vilaplana, Cristina Villanueva-Bueno, Ana Herranz-Alonso, and Maria Sanjurjo-Saez. Features and functionalities of smartphone apps related to covid-19: systematic search in app stores and content analysis. *Journal of medical Internet research*, 22(8):e20334, 2020.

[50] Pascal Courty. Ticket resale, bots, and the fair price ticketing curse. *Journal of Cultural Economics*, 43(3):345–363, 2019.

[51] Sammi Elefant. Beyond the bots: Ticked-off over ticket prices or the eternal scamnation. *UCLA Entertainment Law Review*, 25(1), 2018.

[52] Florian M Farke, David G Balash, Maximilian Golla, Markus Dürmuth, and Adam J Aviv. Are privacy dashboards good for end users? evaluating user perceptions and reactions to google's my activity. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 483–500, 2021.

[53] Simon Feulner, Johannes Sedlmeir, Vincent Schlatt, and Nils Urbach. Exploring the use of self-sovereign identity for event ticketing systems. *Electronic Markets*, 32(3):1759–1777, 2022.

[54] Cuiyun Gao, Jichuan Zeng, Michael R Lyu, and Irwin King. Online app review analysis for identifying emerging issues. In *Proceedings of the 40th International Conference on Software Engineering*, pages 48–58, 2018.

[55] Yuhao Gao, Haoyu Wang, Li Li, Xiapu Luo, Guoai Xu, and Xuanzhe Liu. Demystifying illegal mobile gambling apps. In *Proceedings of the Web Conference 2021*, pages 1447–1458, 2021.

[56] Leo A Goodman. Snowball sampling. *The annals of mathematical statistics*, pages 148–170, 1961.

[57] Enrique Guerra and Bryce G Westlake. Detecting child sexual abuse images: Traits of child sexual exploitation hosting and displaying websites. *Child Abuse & Neglect*, 122:105336, 2021.

[58] Shuang Hao, Kevin Borgolte, Nick Nikiforakis, Gianluca Stringhini, Manuel Egele, Michael Eubanks, Brian Krebs, and Giovanni Vigna. Drops for stuff: An analysis of reshipping mule scams. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 1081–1092, 2015.

[59] Stephen Happel and Marianne M Jennings. The eight principles of the microeconomic and regulatory future of ticket scalping, ticket brokers, and secondary ticket markets. *JL & Com.*, 28:115, 2009.

[60] Stephen K Happel and Marianne M Jennings. Assessing the economic rationale and legal remedies for ticket scalping. *J. Legis.*, 16:1, 1989.

[61] Kate Highfield and Kristy Goodwin. Apps for mathematics learning: A review of" educational" apps from the itunes app store. *Mathematics Education Research Group of Australasia*, 2013.

[62] Geng Hong, Zhemin Yang, Sen Yang, Xiaojing Liaoy, Xiaolin Du, Min Yang, and Haixin Duan. Analyzing ground-truth data of mobile gambling scams. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 2176–2193. IEEE, 2022.

[63] Conny Isaksson and Gustav Elmgren. A ticket to blockchains, 2018.

[64] Blake Ives, Kenneth R Walsh, and Helmut Schneider. The domino effect of password reuse. *Communications of the ACM*, 47(4):75–78, 2004.

[65] Haojian Jin, Boyuan Guo, Rituparna Roychoudhury, Yaxing Yao, Swarun Kumar, Yuvraj Agarwal, and Jason I Hong. Exploring the needs of users for supporting privacy-protective behaviors in smart homes. In *CHI Conference on Human Factors in Computing Systems*, pages 1–19, 2022.

[66] William Klieber, Lori Flynn, Amar Bhosale, Limin Jia, and Lujo Bauer. Android taint flow analysis for app sets. In *Proceedings of the 3rd ACM SIGPLAN International Workshop on the State of the Art in Java Program Analysis*, pages 1–6, 2014.

[67] Li Li, Alexandre Bartel, Tegawendé F Bissyandé, Jacques Klein, and Yves Le Traon. Apkcombiner: Combining multiple android apps to support inter-app analysis. In *ICT Systems Security and Privacy Protection: 30th IFIP TC 11 International Conference, SEC 2015, Hamburg, Germany, May 26-28, 2015, Proceedings 30*, pages 513–527. Springer, 2015.

[68] Chang-Cheng Lin and Hsu-Chun Hsiao. Need tickets? a case study of bot-enabled ticket scalping. 2017.

[69] Enze Liu, Sumanth Rao, Sam Havron, Grant Ho, Stefan Savage, Geoffrey M Voelker, and Damon McCoy. No privacy among spies: Assessing the functionality and insecurity of consumer android spyware apps. *Proceedings on Privacy Enhancing Technologies*, 1:1–18, 2023.

[70] Long Lu, Zhichun Li, Zhenyu Wu, Wenke Lee, and Guofei Jiang. Chex: statically vetting android apps for component hijacking vulnerabilities. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 229–240, 2012.

[71] Peter Mayer, Yixin Zou, Florian Schaub, and Adam J Aviv. " now i'm a bit {angry:}" individuals' awareness, perception, and responses to data breaches that affected them. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 393–410, 2021.

[72] Collins W Munyendo, Yasemin Acar, and Adam J Aviv. "desperate times call for desperate measures": User concerns with mobile loan apps in kenya. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 2304–2319. IEEE, 2022.

[73] Ben Nissan. An arms race on broadway: Bot-based ticket scalping hounds blockbuster musicals. https://www.cs.tufts.edu/comp/116/archive/fall2017/bnissan.pdf, 2017.

[74] Arman Noroozian, Jan Koenders, Eelco Van Veldhuizen, Carlos H Ganan, Sumayah Alrwais, Damon McCoy, and Michel Van Eeten. Platforms in everything: Analyzing {Ground-Truth} data on the anatomy and economics of {Bullet-Proof} hosting. In *28th USENIX Security Symposium (USENIX Security 19)*, pages 1341–1356, 2019.

[75] Alberto Ornaghi and Marco Valleri. Man in the middle attacks. In *Blackhat Conference Europe*, volume 1045, 2003.

[76] Sebastiano Panichella, Andrea Di Sorbo, Emitza Guzman, Corrado A Visaggio, Gerardo Canfora, and Harald C Gall. Ardoc: App reviews development oriented classifier. In *Proceedings of the 2016 24th ACM SIGSOFT international symposium on foundations of software engineering*, pages 1023–1027, 2016.

[77] Dylan C Porcello. A fixed game: The frustrations of ticket scalping and the realities of its solutions. *Brook. L. Rev.*, 84:259, 2018.

[78] Rizwan Ur Rahman and Deepak Singh Tomar. A new web forensic framework for bot crime investigation. *Forensic Science International: Digital Investigation*, 33:300943, 2020.

[79] Tristan Ravitch, E Rogan Creswick, Aaron Tomb, Adam Foltzer, Trevor Elliott, and Ledah Casburn. Multi-app security analysis with fuse: Statically detecting android app collusion. In *Proceedings of the 4th Program Protection and Reverse Engineering Workshop*, pages 1–10, 2014.

[80] Betty Sargeant. What is an ebook? what is a book app? and why should we care? an analysis of contemporary digital picture books. *Children's Literature in Education*, 46:454–466, 2015.

[81] Nolen Scaife, Christian Peeters, and Patrick Traynor. Fear the reaper: Characterization and fast detection of card skimmers. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 1–14, 2018.

[82] Nelson Shen, Michael-Jane Levitan, Andrew Johnson, Jacqueline Lorene Bender, Michelle Hamilton-Page, Alejandro Alex R Jadad, David Wiljer, et al. Finding a depression app: a review and content analysis of the depression app marketplace. *JMIR mHealth and uHealth*, 3(1):e3713, 2015.

[83] James L Swofford. Arbitrage, speculation, and public policy toward ticket scalping. *Public Finance Review*, 27(5):531–540, 1999.

[84] Ronald J Tallarida, Rodney B Murray, Ronald J Tallarida, and Rodney B Murray. Chi-square test. *Manual of pharmacologic calculations: With computer programs*, pages 140–142, 1987.

[85] Jianyu Wang, Rui Wen, Chunming Wu, Yu Huang, and Jian Xiong. Fdgars: Fraudster detection via graph convolutional networks in online app review system. In *Companion proceedings of the 2019 World Wide Web conference*, pages 310–316, 2019.

[86] Joshua H West, P Cougar Hall, Carl L Hanson, Michael D Barnes, Christophe Giraud-Carrier, and James Barrett. There's an app for that: content analysis of paid health and fitness apps. *Journal of medical Internet research*, 14(3):e72, 2012.

[87] Andrew T Williams. Do anti-ticket scalping laws make a difference? *Managerial and Decision Economics*, 15(5):503–509, 1994.

[88] Chuting Wu, Ke Yu, and Xiaofei Wu. Scalping anomaly detection based on mobile internet traffic data. In *Proceedings of the 2nd International Conference on Telecommunications and Communication Engineering*, pages 237–244, 2018.

[89] Cheng Xie, Hongming Cai, Yun Yang, Lihong Jiang, and Po Yang. User profiling in elderly healthcare services in china: Scalper detection. *IEEE journal of biomedical and health informatics*, 22(6):1796–1806, 2018.

[90] Hao-Cheng Yang, Hsun Lee, and Hsu-Chun Hsiao. Poster: Challenges in stopping ticket scalping bots. In *Proceedings of the 15th ACM Asia Conference on Computer and Communications Security*, pages 931–933, 2020.

[91] Jiao Yao, Qingyun Tang, and Jiaping He. Evaluation and analysis of a method to real-name check train tickets during the covid-19 pandemic. *Discrete Dynamics in Nature and Society*, 2021:1–13, 2021.

[92] Deng Ying, Li Weixing, Guo Xiang, Shen Bing, and Xia Xiaohui. Research on electronic ticket system and key technology for road passenger transport. In *2021 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS)*, pages 101–104. IEEE, 2021.

A Supplementary Chart

Following are some supplementary figures and tables provided for a better understanding of the paper.

Table 2: For apps sharing the same package name, signature, and application name, we consider them as different versions of one app. We selected seven groups of apps with more than 100 versions for analysis. The table contains information about the corresponding groups as well as the MD5 hash of the highest downloaded app within each group.

Package Name	Category	Number	First Seen	Last Update	MD5 of the Selected Apps
com.zhonglong.qiangpiaodaren	Train	500	2015/3/18	2019/8/13	bfaf6dadd35eef70953a778413ae3eff
com.iqianggou.android	Commodity	498	2013/2/18	2021/10/15	c4e2feeb499659025f116a63e3e6d81b
cn.suanya.train	Train	150	2022/6/21	2023/4/4	c910b78a89b8569c3bf40dff0c3d3305
com.tieyou.train.ark	Train	134	2016/12/2	2023/4/4	ccd74f27576036949c8ff9927c452c0b
com.train.p00070	Train	129	2013/1/10	2014/12/18	89202f5a271eea287eae3fd62366b824
com.yipiao	Train	113	2022/9/14	2023/4/17	27bec30ef9f214a1918d4cc179ee3e1a
com.app.coomc	Commodity	107	2022/3/30	2022/12/17	3cab4bce1758b23add6fee0cecdc7abf

Table 3: We recruited 184 participants to join in our online survey. The table shows the demographics of the participants.

		n	%
Gender	Male	83	45.11
	Female	99	53.8
	No answer	2	1.09
Age	18-25	68	36.96
	26-30	48	26.09
	31-40	47	25.54
	41-50	15	8.15
	51+	5	2.72
	No answer	1	0.54
Education	Below bachelor	8	4.35
	Bachelor	122	66.3
	Master	29	15.76
	Above Master	24	13.04
	No answer	1	0.54
Occupation	Students majored in CS	45	24.46
	Students from other majors	43	23.37
	IT professionals	30	16.3
	Other professionals	60	32.61
	No answer	6	3.26

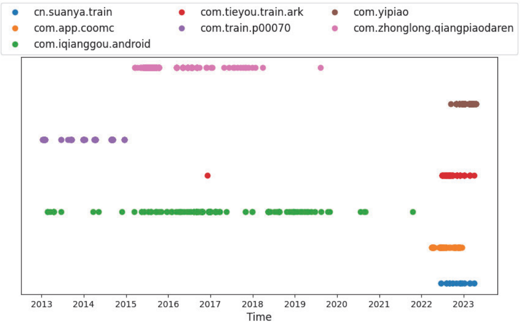
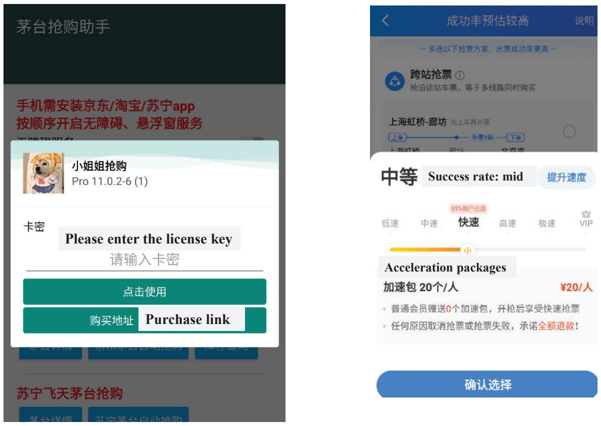


Figure 6: For the seven app groups with a significant number of versions, we obtained the update timestamps and observed that apps exhibit varying levels of activity during different time periods, with updates being relatively sustained.



茅台抢购助手

手机需安装京东/淘宝/苏宁app
按顺序开启无障碍、悬浮窗服务

小姐姐抢购
Pro 11.0.2-6 (1)

卡密 Please enter the license key
请输入卡密

点击使用

购买地址 Purchase link

苏宁飞天茅台抢购

成功率预估较高 说明

— 多选以下抢票方案，出票成功率更高 —

跨站抢票
抢沿途站车票，等于多线路同时购买

上海虹桥-廊坊 先上车再补票

中等 Success rate: mid 提升速度

85%用户已选

低速 中速 快速 高速 极速 VIP

Acceleration packages
加速包 20个/人 ¥20/人

- 普通会员赠送0个加速包，开抢后享受快速抢票
- 任何原因取消抢票或抢票失败，承诺全额退款!

确认选择

(a) Screenshot of paid version. (b) Screenshot of paid service.

Figure 7: Ticket grabbing apps profit from users through two models: paid version and paid service.

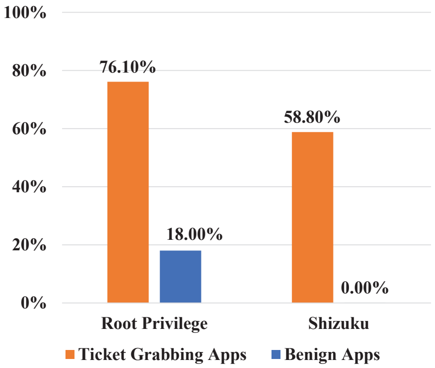


Figure 8: The proportion of getting root compared with ticket grabbing apps and begin apps. The higher proportion of getting root indicates a greater security risk associated with ticket grabbing apps.