ModZoo: A Large-Scale Study of Modded Android Apps and their Markets

Luis A. Saavedra
luis.saavedra@cl.cam.ac.uk
University of Cambridge
Cambridge, Cambridgeshire, UK

Hridoy S. Dutta
hridoy.dutta@cl.cam.ac.uk
University of Cambridge
Cambridge, UK

Alastair R. Beresford
alastair.beresford@cl.cam.ac.uk
University of Cambridge
Cambridge, UK

Alice Hutchings
alice.hutchings@cl.cam.ac.uk
University of Cambridge
Cambridge, UK

ABSTRACT

We present the results of the first large-scale study into Android markets that offer modified or *modded* apps: apps whose features and functionality have been altered by a third-party. We analyse over 146k (thousand) apps obtained from 13 of the most popular modded app markets. Around 90% of apps we collect are altered in some way when compared to the official counterparts on Google Play. Modifications include games cheats, such as infinite coins or lives; mainstream apps with premium features provided for free; and apps with modified advertising identifiers or excluded ads. We find the original app developers lose significant potential revenue due to: the provision of paid for apps for free (around 5% of the apps across all markets); the free availability of premium features that require payment in the official app; and modified advertising identifiers. While some modded apps have all trackers and ads removed (3%), in general, the installation of these apps is significantly more risky for the user than the official version: modded apps are ten times more likely to be marked as malicious and often request additional permissions.

CCS CONCEPTS

• **Security and privacy → Mobile platform security**; *Economics of security and privacy.*

KEYWORDS

mobile, Android, sideloading, applications, security, pirating, analysis, survey, online markets

1 INTRODUCTION

The Android operating system has an open design philosophy, allowing users to easily install apps outside Google Play. As a result, alternative markets run by third-parties have emerged for a variety of purposes. Third-party markets allow developers to share their apps in countries where Google Play is not present, including China and North Korea, as well as charging for apps in Cuba, Russia and Belarus where access to paid apps and in-app purchases is not available on Google Play [18–20, 37]. There are also open source markets such as F-Droid, and device manufacturers like Samsung, Huawei and Amazon may pre-install apps which provide access to their own markets.

This paper focuses on Android app markets that include many modified or *modded* apps in their catalogue. In other words, apps whose code or metadata have been modified by an unauthorised developer or third-party. Modded apps may bypass or unlock subscription features for free, provide infinite in-app or in-game currency, offer paid apps for free and eliminate adverts. This allows users to save money or try apps, games, and subscriptions before obtaining them from legitimate sources; to enjoy an ad-free experience; and to have an advantage over others or save time on games.

While modded markets are niche when compared to the size and scale of Google Play, they form an important part of the Android ecosystem with around 400 markets in operation collectively offering millions of apps. The potential benefits to users are often clearly stated, with modded apps claiming to provide desirable features. However, the extent of the modifications, the security implications for users, and the incentives for developers and market operators are less obvious and so far unstudied. We fill this gap in knowledge by first identifying 423 modded markets, studying their size, presence of ads and blog spots, as well as ranking by popularity. We then analyse over 146k (thousand) apps and their metadata obtained from the 13 most popular modded markets over a 3-month monitoring period.

We statically analyse and match these apps with their Google Play equivalents where possible, allowing a direct comparison between modded and official versions. The larger modded app markets operate at scale, with an average of over 37k apps (max 221k) and around 2k apps added every fortnight.

The presence of these markets is likely to reduce income for app developers and official markets. Around 5% of the apps are free copies of paid apps available on Google Play, with a total value of USD $33 975 and estimated lifetime revenue (current price × Google Play installs) of around $2 billion. We also find premium features in popular apps, which are usually charged via In-App Purchases (IAPs), available for free in modded versions. Examples include the availability of TikTok coins, and ad-free audio in Spotify; public accounts report billions of IAP revenue per year for these two apps alone [10, 29, 42]. We also find that 22% of modded apps with ad IDs (advertiser IDs) have different IDs to the official version in Google Play, and 6% of modded apps include additional advertising libraries, potentially redirecting ad revenue away from the original developer to a third-party. Modded apps are also more risky for the consumer. While many modded apps claim to remove ads, only 3% do so. Furthermore, 23% of modded apps request additional permissions and nearly 9% of apps are marked as malicious by VirusTotal, around 10 times the rate found in Google Play versions.

The ability to install apps outside the official market mirrors the status quo found in consumer laptop and desktop operating systems and has several potential benefits, including providing consumers with more choice and allowing developers to sell apps and premium features directly to customers without paying a percentage of revenue to the official market. Nevertheless our work shows that, in the case of modded apps at least, there are also significant negative effects. This work is timely, and thus important for regulators, since they need to balance competition and fair market access on the one hand, and consumer and intellectual property protection on the other. The question of whether mobile devices should allow the installation of apps outside the official market is under investigation, including by the European Parliament's Committee on Internal Market and Consumer Protection (IMCO) in relation to their Digital Markets Act [15, 16], as well as the UK's Competition and Markets Authority (CMA) [14]. Our work suggests that, while third-party markets have the potential to benefit consumers and developers, some regulation may be required to ensure consumer security and protect developer revenue streams.

In summary, we make the following contributions:

- An overview of the modded app ecosystem and the first in-depth study of markets containing modded Android apps.
- Monitoring, data collection and analysis of 13 of the most popular modded markets over a three-month period, collecting over 146k modded Android apps.
- We make our dataset, *ModZoo*, available to other researchers.
- By matching apps from modded markets with their Google Play counterparts, we find around 90% of apps are modified in some form and 75% have modified code.
- The presence of these modded markets is likely to reduce income for app developers and official markets due to: the widespread availability of paid apps for free; premium features offered for free; and the redirection of ad revenue, including 22% of apps with altered ad IDs.
- Modded apps are more risky for the consumer: 23% of modded apps requested additional permissions and nearly 9% were marked as malicious by VirusTotal.

2 RESEARCH QUESTIONS AND METHODOLOGY

This section introduces our research questions and methodology. We discuss the ethics of our research in Section 6. *Modded apps* are defined in this study as any app that has had its code or metadata modified by an unauthorised developer or third-party. This can include changes to advertising libraries, certificates used to sign the apps, app permissions, or even methods in the code to provide users with premium features, or in-app/ in-game resources, IAPs, etc. Therefore, *modded markets* are those sites that provide interfaces similar to Google Play and third-party markets, but focus on or advertise a large catalogue of predominantly modded apps.

2.1 Research questions

The main research questions this paper answers are:

RQ1 What do the modded markets and apps ecosystem look like, and what is its size?

RQ2 What are the financial incentives for operating modded app markets? How does this affect the original developers and markets?

RQ3 What are the security implications of installing apps from these markets?

2.2 Identifying modded markets

We obtain a list of 423 sideloading and modded app markets by querying two popular search engines: Google Search and DuckDuckGo, with keywords such as 'Android app stores', 'mod APK', 'download premium APK', 'YouTube mod', etc. The complete list of keywords can be found in Appendix A. The search was run in four languages: English, Chinese, Russian and Hindi. Chinese and Russian were chosen due to the limited availability of Google Play in China and Russia. Hindi was added as preliminary results included Indian domains ('.in'). We manually verified the existence of apps advertised as modded apps in the markets.

2.3 Market ranking methodology

All 423 markets cannot be analysed in depth, thus a popularity-based ranking of the markets was curated using Google Trends. While only useful to compare the popularity of keywords over time, pair-wise comparisons for the 6-month period leading to our study allow us to obtain a relative ranking for all 400+ markets. We then cross referenced this ranking with the Tranco ranking corresponding to the 9-month period leading up to our study [36]. We found that the markets we analyse cover the top 7 markets in the Tranco ranking, 9 out of the top 10, and the other 4 markets are within the top 35. Thus, we cover the most popular markets, as measured by both Google searches and the Tranco list. Interestingly, out of those in our list of 423 markets, only 38 out of the top 60 in the Tranco list still offer modded apps three months later. In other words, 22 markets were no longer in operation or changed their focus to other activities such as offering news articles.

2.4 Nomenclature

The 146k *modded apps* in our study each have a unique hash and correspond to 48 384 unique package names, i.e. they are different modded versions of 48k unique apps. We refer to *exact matches* where we find an app one market with the exact same package name and version code as seen in another market. Unless stated otherwise, we use exact matches for all our comparisons. We use the *non-exact, latest-available match* when comparing a potentially malicious apps found on a modded market with the latest version of an app with the same package name on Google Play. Non-exact matches are a reasonable proxy when studying maliciousness as we assume later versions of the same app on Google Play are at worst similarly malicious to older versions. Non-exact latest-available matches are also the latest and only versions available in Google Play, so sections looking at app and IAP prices use the latest version metadata directly from Google Play as we were unable to find a reliable source of historic price data. Modded APKs and their Google Play matches are analysed and the resulting profiles are stored for later comparison. We will refer to apps on Google Play which cost money as *paid apps*, while any exact matches on modded markets are referred to as *pirated apps* because they are offered without charge on modded markets.

In later sections we discuss five different types of app. *Hash-identical* apps are those where the entire binary is hash-identical, i.e. where the entire packaged application (APK) is bit-for-bit identical, including manifest, libraries, code, etc. We also explore *code-identical* apps: those whose code (.dex) files are the same, but other aspects, including permissions and manifest might differ. Similarly, *certificate-identical*, *permission-identical*, *ad library-identical*, and *ad ID-identical* apps, are those whose signing certificate, permission set, ad libraries set and advertising IDs are the same as found in their Google Play version, respectively. Their counterparts are *code-modded*, *certificate-modded*, *permission-modded*, *ad library-modded*, and *ad ID-modded* apps.

2.5 ModZoo dataset collection

Our ModZoo dataset consists of 146 162 downloaded modded apps, their metadata and analysis results as well as their 87 792 exact and non-exact, latest-available matches from Google Play.

We obtain Google Play apps from AndroZoo, a dataset which includes 20 million apps from Google Play, including different versions of the same app [2]. We scraped the 13 most popular modded markets (see §2.3) between September and December of 2022 every 10-14 days to build our dataset of modded apps. Our custom parallelised scrapers are written in Python to quickly obtain all relevant pages and APKs from the 13 modded markets. We used a set of proxies around the world to perform our data collection. Some of the scrapers use only HTML requests, while others also require Selenium and Mozilla's Gecko Driver to imitate user interaction. For other markets, we scraped their website first, obtained the APK IDs, and then contacted the endpoints used by their custom market app. All information pages were stored, including the download pages, and all available modded APKs were downloaded.

We compute SHA256 hashes of all APKs, ensuring we only store each app with a particular hash once. We map modded apps to their Google Play counterparts to enable a comparison between modded APKs and the official versions of those same apps found in Google Play. ModZoo also includes the VirusTotal analysis results of 175 584 APKs, including 103 914 modded and 71 670 Google Play (AndroZoo) APKs. The difference between the size of our ModZoo dataset and the number of VirusTotal analysis results is due to the use of existing results, as previous studies have found VirusTotal results to be more reliable after repeated scans [45, 46, 56].

We make the ModZoo dataset available to the research community (see §9).

2.6 Static analysis methodology

Static analysis allows relatively quick results, ideal for the ModZoo dataset of more than 146k modded apps and their almost 88k Google Play counterparts.

Our analysis pipeline starts by obtaining the latest data from AndroZoo. Then, it analyses the modded APKs yet to be analysed in parallel, returning and storing their metadata and closest AndroZoo match, as well as whether it is an *exact* or *non-exact, latest-available* match (see §2.4). It then analyses the obtained AndroZoo match and stores the results.

In order to obtain an analysable folder for each modded app, we run the third-party reverse engineering tool Apktool [27]. We use the UNIX 'keytool' command to obtain the certificate information from each app. Our *Certificate Parser* returns the certificate 'owner', 'issuer', 'serial number', 'certificate SHA256', 'signature algorithm', etc. We then run Apktool again, which creates the 'apktool.yml' file, which we parse to obtain the APK's filename, minimum and target SDK versions, and version name and code. We parse the 'AndroidManifest.xml' file using a third-party Python XML parser library. This *Manifest Parser* returns metadata attributes including the app's package name and version, as well as permissions, activities, providers, receivers, intents, etc.

To detect advertising libraries in the analysed APKs and their Manifest files, a 'safelist' of ad library package names was created and iteratively extended as explained below. The *Manifest Parser* analyses the 'application' 'meta-data' and 'activity' attributes thoroughly, as this is where AppLovin and GoogleAds ad IDs, as well as the presence of IAPs can be found. We check whether the application attributes are present in our ad libraries safelist. If not in our list, it is added to a list of potential candidates to join the list, to be manually checked later. Thus, we have continuously expanded our safelist of ad libraries and reanalysed apps which analysis was older than the latest version of the safelist. All of the information gathered is stored as a profile in JSON format. The results returned to the analysis pipeline are: the package and version name, the JSON profile, ad IDs and ad libraries found, and ad library candidates. Then, the package name obtained from the manifest file and version code from the Apktool step are used to obtain from AndroZoo – where available – the *exact* or *non-exact, latest-available* match Google Play app.

The AndroZoo match app analysis follows the same steps described above, except the AndroZoo step is skipped. The modded app analysis results are stored, as well as those of their AndroZoo match.

2.6.1 *Modded apps Google Play matching.*

A total of 136 620 out of our 146 162 downloaded modded apps were matched with a Google Play app present in AndroZoo using the methods described above. The 6.5% unmatched modded apps correspond mostly to paid apps and games not available in AndroZoo, and a small proportion of apps not allowed in Google Play or exclusive to modded markets.

Out of those matched, 88.6% correspond to exact matches, i.e. those with the same version number and package name, and only 11.4% are non-exact, latest-available matches, i.e. those with the same package name but the latest version number available at the time of the analysis.

3 THE MODDED APP ECOSYSTEM

This section tackles RQ1: "What do the modded markets and apps ecosystem look like, and what is its size?" We leverage insights from our manual analysis of the 423 markets and the static analysis of our 146k app dataset obtained from the 13 most popular modded app markets.

3.1 Analysis of modded apps and markets

Our technical analysis focused on the 13 highest-ranked modded markets, as determined in §2.3. Their average estimated size is 37 486 apps with a mean of 15 719 apps downloadable, counting different versions of each app both in terms of modded features and version numbers (see Table 1). We estimate their size based on the number of apps listed, the difference between this number and number of apps downloaded is due to the unavailability of some apps and broken download links.

Furthermore, those markets marked with an asterisk (∗) in Table 1 were only partially scraped because they label the modded apps, clearly distinguishing what they consider their modded catalogue from the rest. While this approach made scraping these markets feasible, further analysis revealed the market definition of a modded app differs from ours (a limitation we discuss in §3.1.2).

Unlike the modded markets, Google Play only offers the latest compatible version of each app, and only one version per package name. Our smallest market analysed is 'AN1' with 2 696 unique apps, and 'MODDROID' is the biggest in terms of unique apps downloaded with more than 30K. Finally, 'Appvn' has the biggest estimated size, with more than 220k apps (see Table 1).

The number of distinct apps is halved when counting unique packages names and occurs because markets often provide multiple versions of each app (see Table 1).

Apps on these markets change frequently, with around 4k new apps added weekly across all markets. However, when looking at hash-identical apps as defined in §2.5, around 25% of apps are hash-identical duplicates, with 39 988 APKs out of the 146 162 unique apps found in more than one market. The total number of unique apps obtained from each market over the course of the study is shown in Table 1. This is followed by the number of APKs that are hash-identical within the market, i.e. those advertised as different versions or apps that are actually hash-identical duplicates of other apps in the same market.

All modded markets we studied lack any payment mechanism, thus the paid apps included in Table 1 are available for free, and likely pirated copies of paid Google Play apps. The mean percentage of paid apps available for free across all markets is 4.7%, with only 'Malavida' hosting 0.0% (6 apps in total).

3.1.1 Modded apps and modified code. The 'Modded Apps' and 'Unchanged Apps' columns present the number of apps per market that have been modified in any way, and those that are hash-identical copies of Google Play apps, respectively. Focusing on exact matches, the number of code-identical and code-modded apps as defined in §2.5 is computed. A total of 81 250 apps (68.1%) have received changes to their code, shown per-market in the 'Modded Code' column in Table 1. Code-modded apps are closely related to permission-modded apps, as discussed later (see §3.3). Interestingly, although the markets focus on code-modded and modded apps, it is clear from these results that some of them have more code-identical than code-modded apps. This could be due to several reasons, the simplest being trying to offer a wider catalogue of apps. Some changes such as making apps ad-free might be made easily without modifying the code.

3.1.2 Apps labelled as unmodded. We found apps not labelled as 'modded', or labelled 'unmodded' or 'original' are rarely hash-identical to their exact matches from Google Play, highlighting the inaccuracies of these labels. We obtained more than 10k app samples from 'Appvn', 'Androeed', and '5play' and compared them to those in AndroZoo based on their (SHA256) hashes since all signatures, metadata and code should be identical for unchanged apps. We obtained the following results for the self-reported unmodded apps: Appvn had 35.8% hash-identical apps, higher than the 0% found in the modded side of the market; Androeed had only 6.5%, up from 5.7%; and 5Play had 8.3% hash-identical apps, down from 10.0% in the rest of the market.

3.1.3 Categories. We were interested in whether modded markets focused on particular types of apps. To determine this we found the latest-available match on Google Play for each modded app and then obtained the Google Play category for those apps. We also computed the popularity of app categories on Google Play by obtaining a random sample of 100k Google Play apps. As shown in Figure 1, the 9 most popular categories in modded markets are game categories: 'Action', 'Simulation', 'Arcade', 'Puzzle', 'Casual', etc. Google Play categories, however, are led by 'Education', 'Business', 'Tools', 'Health and Fitness', 'Lifestyle', 'Finance', etc. most of which are at the tail end of modded app categories. We therefore conclude that modded markets focus heavily on games when compared with the Google Play app ecosystem.

3.1.4 Modded features. Modded markets typically provide descriptions of modded app features to inform and entice potential users. The Android catalogue has gradually shifted towards 'Freemium' apps [26]: apps with IAPs or subscriptions, and typically in-app advertising. Thus, popular modifications include 'mod money', 'unlimited money', 'free shopping', and 'premium unlocked' (see top 10 list in Appendix B) – mainly associated with Freemium apps and games.

3.1.5 Paid (pirated) apps. There are 6 984 pirated apps in the 13 markets, which correspond to 2 241 unique package names. The total value of the apps is USD $33 975, and $9 674 when counting each app (package name) only once. The approximated total lifetime revenue of the paid apps found in these 13 modded markets is $2.28 billion. We obtained this as the product of their current price in the US Google Play times the number of installs as reported by Google; our estimate excludes any possible in-app purchases.

We estimate modded market operators would have spent at least $9 674 to get these paid apps from Google Play before hosting them in their markets if they had worked together and shared all their apps, or $26 901 if they had to buy each of the paid apps they host once. It is possible market operators downloaded paid apps and requested refunds after making copies [21], resulting in $0 of revenue for the original developers. The price distribution in Google Play is shown in Fig. 2, showing around 40% of the apps are priced at more than $5. Around 48% of them are very popular, with more than 500k installs in Google Play.

3.1.6 In-App purchases (IAPs). There are 100 118 apps with IAPs in modded markets, with prices of up to $1 024 per item in Google Play. Their total price is at least $3.7 million, based on the values reported on Google Play. This is an underestimate, as Google Play only

Table 1: Overview of the modded market ecosystem and proportion of modded apps and code. Markets and results marked with an asterisk (*) correspond to partially scraped markets.

Market	Estimated Size	Unique Apps	Unique Packages	Duplicates	Paid (%)	Modded Apps	Unchanged Apps	Modded Code
Appvn*	221 039	*4 389	*1 866	*8	*5.0	*4 297	*0	*3 586
RevDl	42 540	30 477	9 599	187	6.4	23 217	3 466	5 068
HappyMod	41 385	26 737	17 249	12	3.7	19 996	4 098	12 826
MODDROID	34 312	30 738	17 152	13	3.7	23 316	4 005	15 153
APKMODY	33 914	10 516	3 081	214	4.5	8 857	420	4 550
androeed*	24 252	*15 450	*6 869	*6 195	*3.4	*12 069	*731	*9 163
Rexdl	22 988	14 262	5 824	24	8.4	11 666	1 822	2 621
5play*	19 014	*19 674	*15 859	*16 203	*8.1	*16 095	*1 610	*9 917
Malavida	16 519	19 648	16 128	16	0.0	14 333	4 115	3 084
APKDONE	11 080	14 908	3 232	113	4.3	10 099	139	7 341
ApkVision	8 491	7 983	6 900	16	5.9	5 632	1 055	2 683
LMHMOD	7 880	6 865	4 317	6 303	3.7	5 577	229	3 597
AN1	3 906	2 696	1 198	44	4.0	2 629	22	1 661

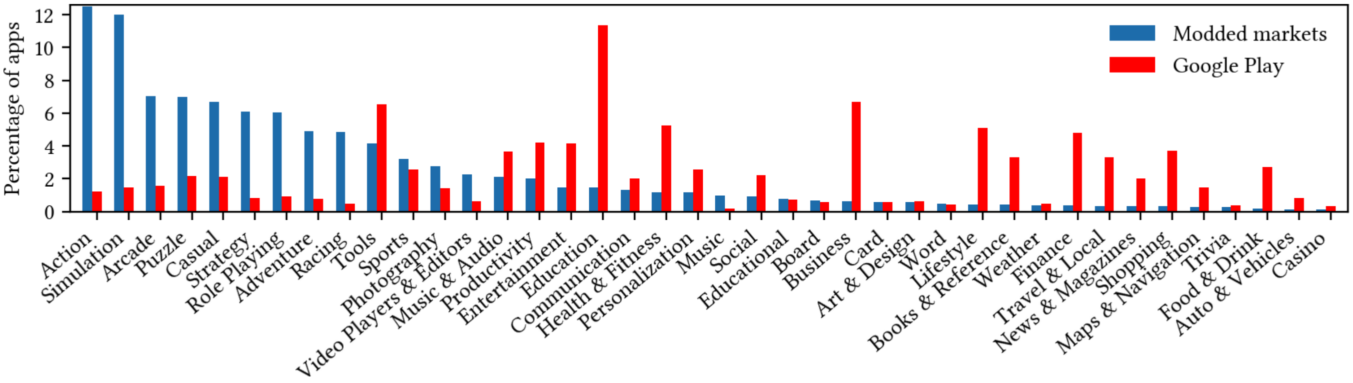

Figure 1: Partial distribution of app categories.

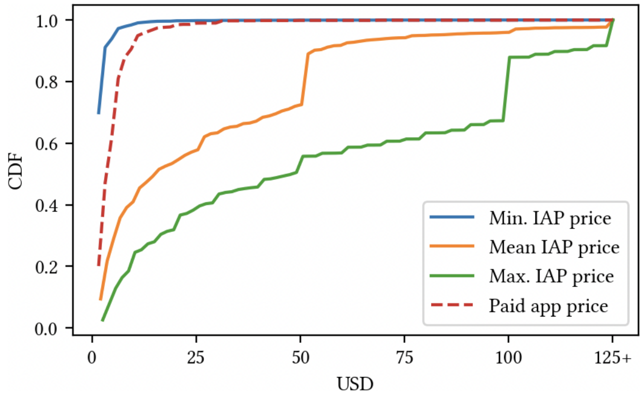

Figure 2: Google Play paid apps and IAPs price CDF.

reports a price range and many IAPs represent periodic purchases (subscriptions) or consumable items that can be purchased multiple times. Much IAP content and features are provided for free in the modded apps (see §3.2). The maximum, minimum and mean IAPs prices (in Google Play) are shown in Fig. 2, showing over 40% have a maximum price over $100.

3.1.7 Countermeasures and market changes. Markets employ different countermeasures against scraping and automated downloads, which we encountered during data collection. This is understandable as operators want to prevent other market from obtaining apps from their market at scale. We observed that all markets analysed contain duplicate apps also found in others, as mentioned in §3.1. The most common defence to limit scraping is a waiting period, which serves two purposes: it preventing users from downloading multiple APKs in a short period of time and also provides an opportunity to show users ads while waiting. Some markets used CAPTCHA tests to limit automated scraping, and a small number implement Cloudflare DDoS and bot protection [13].

We noticed some markets introduced anti-scraping protections during the course of our research. It is possible that our effort to contact markets for comment, our scraping activity, or both made the operators suspicious and more security-conscious. One market removed their 14 social media links (including Github, LinkedIn and YouTube) from the English but not the Vietnamese version of their site during our study.

Some popular modded markets require users to download a proprietary app to download the APKs they host. These include MODDROID, Jojoy, and HappyMod which use shared endpoints, hosting mostly the same APKs in a shared back-end. Their websites merely point users to their market app download link. Our scrapers were adapted to obtain the metadata from the website and contact the app back-end to download the APKs directly without using the app. A couple of months into the study HappyMod started redirecting users to Jojoy, and Androeed completely redesigned both their Russian and English websites.

3.2 Case study

The five all-time most popular apps and five most popular games (as of March 2023) from Google Play are presented as a case study. We manually test the Google Play version alongside two or three modded versions from different markets to analyse their modded features and help us assess the scale of revenue loss caused by modded apps. Google Play Protect is supposed to protect users by warning them of harmful apps on their devices, even when installed from other sources. It may also deactivate or remove harmful apps. During our case study Play Protect issued warnings stating “Unsafe app blocked” for 2 out of the 30 apps tested (28 apps and games and 2 market apps), these were a game and the ‘APKMODY’ market app. The user only needs to click “Install anyway” to install the apps. Whether Google Play Protect succeeds in protecting users at scale is not something we investigated further.

Many of the modded apps we checked showed a small badge, logo, or pop-up window stating the name of the market or in a few cases the modder that created the mod. In some cases the market name displayed did not correspond to the market we obtained the app from. In terms of the advertising IDs present in the 28 modded versions of the 10 apps and games studied, we found 14 versions and their equivalent Google Play apps had Google Mobile Ads and/or AppLovin ad IDs present. Of these, one TikTok and one Truecaller version had their Google Mobile Ads IDs removed, the other 12 versions had unchanged ad IDs.

TikTok generates revenue through ads, but also through IAPs in the form of coins users can send to creators during livestreams, triggering animations and resulting in revenue for creators [33, 50]. These come in bundles costing USD $0.07–249.00 for 5 to 17 500 coins. TikTok reported $1.5 billion IAP revenue in 2022 [33]. The descriptions of modded TikTok apps claimed to offer unlimited coins, downloading without watermarks and geolocation restrictions removed. Downloading without watermarks worked well, but coins were not included in the modded versions we tried.

SHAREit offers premium features for $1.99/month, including removing ads, exclusive customer service, and regular cleanup and antivirus. Modded versions of this app claim to remove all ads and include all premium features. None of the versions we tested removed all ads, only one provided regular cleanups and none provided premium customer service.

Telegram Premium costs $4.99/month, or $35.99/year and includes no ads and doubled limits (channel size, download speeds, document size, etc). Modded versions of Telegram claim to provide the premium features, including no ads. Ads could not be checked using our test accounts, as ads are only shown in public channels and were not served to our accounts in the genuine nor modded versions. The modded versions we tested did not provide any other Premium features, with download speeds limited in the same manner as the free version.

Spotify Premium costs $9.99/month and provides an ad-free experience, higher sound quality, playing songs in any order, unlimited skips, downloads and offline listening. Spotify reported having 2 million users running modded versions of their app to avoid audio ads and subscriptions [42]. They also reported a revenue of €10.25 billion from Premium subscriptions and €1.5 billion in ad-supported (i.e. non-premium) users in 2022 [24, 29, 42]. Modded versions are advertised as having the premium features. Many, but not all, markets make it clear they cannot provide downloads, offline listening and higher quality audio. All versions tested were ad-free and provided unlimited skips, the ability to play any song and play them in any order. As suspected, none provided downloads, higher quality audio, nor the ability to select a different device to play on.

Truecaller Premium costs $4.99/month or $49.99/year and includes no ads, advanced spam blocking, seeing who viewed your profile, incognito mode, etc. Modded versions claim to have all premium features. However, although they have no ads, the modded versions we tested show all users as Gold members, with no effect for genuine users. Only one version showed who viewed or searched the user’s profile.

Subway Surfers offers different ‘coins’ and ‘keys’ bundles for $0.99–99.99. Modded versions claim to have all these IAPs unlocked, some even offer ‘God mode’ game-play advantages such as unlimited jumps, flying, etc. Tapcore stated piracy had cost this game $91 million by 2017 [32]. The modded versions we tested provided an unlimited amount of in-game currency and free IAPs.

Candy Crush Saga offers many perks bundles from $0.99–99.99 and ‘gold’ for $1.99–99.99. All modded markets advertise having all levels unlocked, infinite lives, boosters, etc. Such offerings render gold, lives, and other IAPs useless. The modded versions we tested worked as advertised.

Free Fire offers subscriptions for weapons and perks costing between $1.99/week and $12.99/month and ‘diamonds’ for $0.99–49.99. Modded markets advertise unlimited money and diamonds, with gameplay-related mods including aim-assist, no recoil, and hacks [54]. The first few modded versions we tested did not work at all, and none of these features were present in the versions of this game that did work.

My Talking Tom offers monthly subscriptions for $4.99 for perks, as well as ‘diamonds’ for $1.99–99.99. A purchase is required to remove all ads. Modded versions advertise unlimited coins and in-game items plus no ads. From our testing, most content is unlockable through unlimited coins, but ads including full-screen pop-up ads are still present and subscription perks locked.

Hill Climb Racing sells perks and in-game currencies in multiple bundles for $1.99–59.99. Unlocking all vehicles costs $29.99, all levels $29.99, and unlocking everything $54.99. Specific bundles remove ads when bought. Modded versions advertise having all content unlocked or offering unlimited in-game currencies. All versions tested provide unlimited in-game currencies, letting users unlock all content in the game, although with banner and pop-up ads still present.

3.3 Code, permissions and ad libraries

We studied changes to the code, stored in the 'classes[n].dex' file(s) in relation to changed permission sets, ad libraries, and ad IDs for all apps available from the modded markets analysed with an exact match from Google Play. Of these pairs present in our ModZoo dataset, a majority (75.0%) are code-modded, having received some modifications to their DEX files, as can be seen in Figure 3. In terms of exact matches, each Google Play app is matched with an average of 1.57 modded apps.

For code-identical pairs of apps, permissions and ad libraries remained completely unchanged in 99.9%, and 100% of the pairs, respectively. Their ad IDs either remained unchanged (65.3%) or no ad ID was found. This suggests many apps hosted in these markets are copied directly from Google Play without alteration. This might be done to expand the catalogue in order to support user engagement even if a modded version is unavailable. Many of the markets studied advertise both modded and unmodified versions of each app, although we found inaccuracies in these labels (see §3.1.2). Even unmodified apps may be useful to users, since the availability of older versions may offer distinct features or be necessary for compatibility with older versions.

We found 38.4% of code-modded apps are also permission-modded, with the majority (59.4%) including additional permissions. Some code-modded apps require further permissions for reasons related to the modifications, but the reason behind many of the additions was unclear. Code-modded apps are mainly ad library-identical (83.3%), and 11.1% of them have fewer ad libraries. In terms of ad IDs, 36.2% of modded apps had none, of those with ad IDs, 21.6% had them changed. Given that the altered ad IDs occur in a significant proportion of code-modded apps, we hypothesise permissions are sometimes added to code-modded apps in order to increase ad revenue for the modder.

3.4 App signing certificates

Code-modded apps certificates give us insight into the origin of some modded apps. We found most markets use mainly debugging and default Android Studio certificates unfit for app publishing, typically followed by certificates with empty fields. Some markets have market-specific signatures such as '5play', which uses its signature to sign the majority of code-modded apps with the same certificate. So does APKMODY, which includes the operator's name 'Anh Pham' but mainly uses default signatures. Others use mainly 'A1 Lazyland RU', present in most markets in some proportion, Appvn uses mainly 5play.ru as well. All markets have 5play.ru and/or APKMODY certificates except Malavida.

There are third-party markets' and modders' certificates in a smaller proportion in all markets, with many of these certificates including websites and Telegram links. A small portion were signed with 'AntiLVL', an Android License Verification Library Subversion Tool. This analysis confirmed our previous findings of cross-market duplicate apps.

> This section has looked at pirated apps with an estimated USD $2.28 billion lifetime revenue on Google Play, as well as IAPs and a case study of popular apps and games. The most popular categories and modded features in these markets relate to games. Our analysis indicates many of these apps are code-modded, mostly to remove the need to pay for subscription features or in some cases remove intrusive ads. Others added permissions, ad libraries, and changed ad IDs. We also found 'unmodded' labels cannot be trusted in the markets analysed, nor descriptions of supposedly 'ad-free' apps.

4 MARKET OPERATOR MOTIVATIONS AND INCOME

This section tackles the second research question: "What are the financial incentives for operating modded app markets? How does this affect the original developers and markets?" We approach these questions based on our observations and analysis results to analyse possible revenue streams in modded markets and operators' economic incentives.

4.1 Blog spots, sponsored posts and ads

We manually studied the blog spots, sponsored posts and ads present in all 423 markets we identified, manually confirming blog spots are present in a third of modded markets. Blog spots are separate sections in these markets, populated with articles about new updates to the modded apps, installation guides, etc. and often also news articles, tips and tricks, rankings, and even product or app reviews. Many markets openly displayed their pricing for anyone interested in advertising through different formats of ads, product reviews, or writing guest posts. Others, however, were open to contact via email for information, or "custom requests". Only a minority claimed not to accept sponsored posts or ads. Some blogs are inactive and 13% have 5 or fewer posts, suggesting their main sources of revenue are the ads shown in the website and in-app ads. One market priced sponsored posts and ads at USD $250–300; others accepted guest or sponsored posts for $100. Most had lower prices starting at around $30 for general posts, $45 for casino-related posts, and more for those related to "gambling, adult, dating, vaping, CBD, or cannabis". Others offered different ad sizes and types including sidebar and pop-ups for $50–200/month. Most blog posts are admin-uploaded, making it difficult to quantify the number of sponsored posts.

4.2 Advertising libraries and advertiser IDs

Based on our analysis, the 13 most popular modded markets offer 6 984 pirated paid apps (see §3.1.5). However, 'Freemium' apps are increasingly popular [26]. Many code-modded apps offer subscription features, bypass subscriptions entirely, or include IAPs for free. Some are even advertised as ad-free versions of Google Play apps. We use a safelist (see §2.6) to confirm if they deliver on these promises, and what other changes might have been introduced.

Google Mobile Ads and AppLovin ad libraries (two of the most popular) include their advertising ID in the AndroidManifest file, allowing us to compare these ad IDs of modded and original apps. Using this method, we found 20.5% of modded apps with ad IDs had them altered. It therefore appears to be widespread practice to

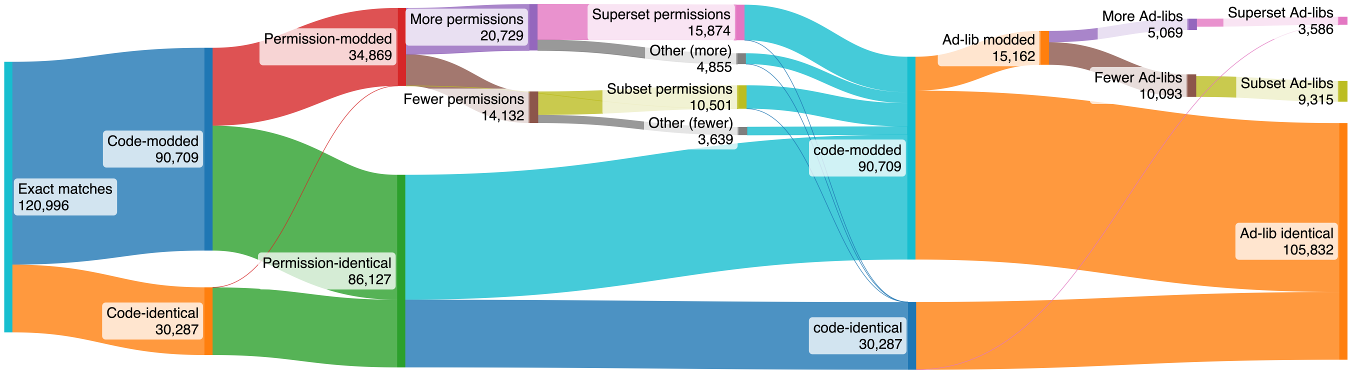

Figure 3: Distribution of permissions and ad libraries in relation to code-identical and code-modded apps.

redirect ad revenue from the original developer to the modders or modded markets. We also found 41 321 apps use ad libraries other than Google Mobile Ads and AppLovin, and in total 10 990 apps present changed ad libraries compared to their Google Play version.

We found 18 628 modded apps contain no ad libraries: 10 353 of these did not contain any originally, but 4 180 (2.86%) had all ad libraries removed with respect to their Google Play counterparts and 2 636 had their AppLovin and GoogleAds advertising IDs removed. So, while some modded apps have had advertising libraries removed, they are very much in the minority. We note that the presence of libraries implies the possibility of including ads in an app, not necessarily active usage. An example is the popular Unity library used in many games ('com.unity3d') which can be used to display ads but also offers significant non-ad functionality. Thus, we may have underestimated the number of ad-free apps. Further dynamic, manual analysis would be needed to confirm this. This is impractical given the scale of our dataset.

The most popular advertising and tracker libraries present in our modded apps are GoogleAds, Facebook, and Unity, followed by AppLovin, ironSource, Vungle, AdColony, Tapjoy and InMobi. Their relative popularity is mostly the same in modded apps and their Google Play counterparts. Providers most affected by 'ad-free' modded apps are GoogleAds (20.3%), Facebook (14.9%), Unity3D (9.6%), and AppLovin (8.1%).

4.3 Advertising libraries, advertiser IDs and permissions

Changes to Android permissions have clear security implications and, in theory, combined with the aspects already presented in relation to advertising libraries and ad IDs, extended permission sets might provide increased revenue to modders or market operators. This might happen through the addition of location permissions, for example, which the advertising library can use to display more relevant ads.

The small proportion of ad-free apps (2.86%), those which contain no ads where their Google Play counterparts do, typically present fewer permissions (88.9%), as shown in Figure 4. Ad-free apps have a strict subset of the original permissions in 76.2% of cases, and only 4.2% are permission-identical. This shows there is a genuine (albeit small) offering of ad-free versions of popular apps that result in smaller permissions sets and equal or better privacy for users in terms of permissions.

Furthermore, as shown in Figure 4, 91.0% of modded apps are ad-library-identical, having the exact same set as their Google Play counterparts, the rest evenly split between added and removed libraries. These ad-library-identical apps tend to also be permission-identical apps (84.3%), with the rest mostly having more (11.4%) and a superset (10.3%) of permissions. Apps with added ad libraries are mainly permission-modded apps (91.3%), with 64.9% of them having more permissions. Those with removed ad libraries were also permission-modded apps (88.6%), but mainly contained fewer permissions (60.9%) or a strict subset of permissions (44.1%). These results show ad libraries are not typically changed in modded apps and are closely linked to permissions. User privacy is typically enhanced in terms of permissions when ad libraries are partially removed, and worsened when ad libraries are added.

Furthermore, when focusing on their ad IDs for AppLovin and GoogleAds, 58.0% of modded apps with ad libraries have unchanged ad IDs, 8.1% had changed ad IDs, and 33.9% of the modded apps had no ad IDs. Ad-ID-identical apps were mostly permission-identical (83.6%), with another 13.6% having more permissions, as shown in Figure 5. Ad-ID-modded apps, however, were permission-modded in 61.1% of the cases, with an even split of more and fewer permissions. Those with no ad IDs were mainly permission-identical (76.7%). This suggests again a strong correlation between permission-modded and ad library and ad-ID-modded apps. This could be due to more complex modding leading to modders wanting to be compensated with ad revenue or more permissions giving more granular data to advertising libraries, increasing ad revenue.

4.4 Modded apps and user displacement

Piracy has been studied in the music and media industries and found to have a negative impact on revenues [44, 49]. Advertising company Tapcore estimated 14 billion app installs were pirated installs in 2017, costing developers $3–4 billion, having cost apps more than $17.5 billion by 2017, and cost Subway Surfers $91 million [32]. It is difficult to estimate the current revenue loss with the growth of IAPs and ad revenue in apps and games since 2017. We were unable to find a study looking specifically at piracy of mobile apps, however other studies of computer games found a high

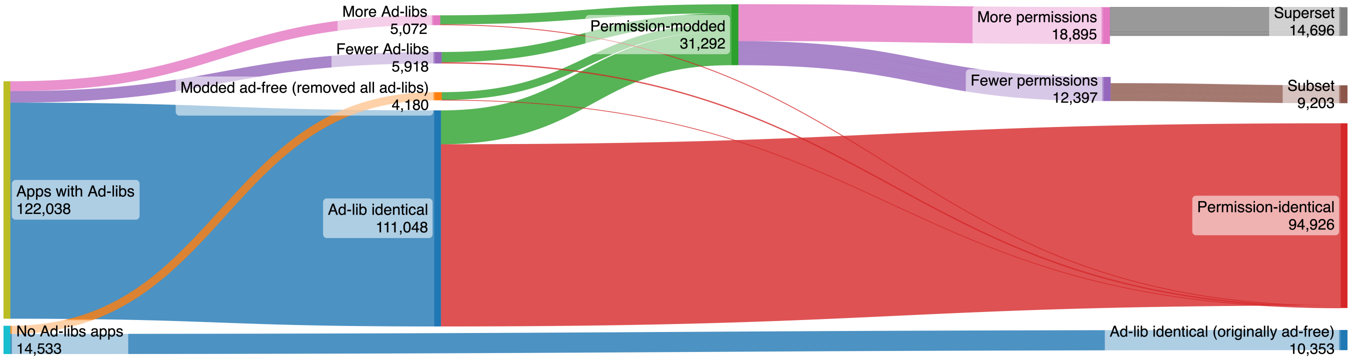

Figure 4: Distribution of permissions and ad libraries.

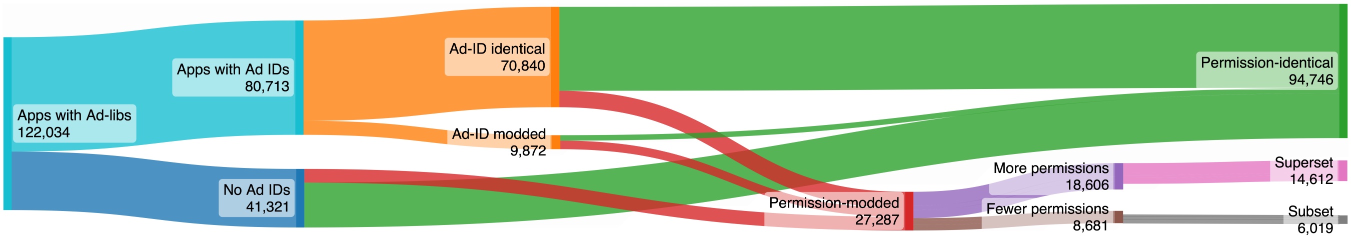

Figure 5: Distribution of permissions and advertiser IDs.

displacement rate of -2.49 for games, meaning each illegal download of a game typically displaces multiple genuine purchases [43]. Furthermore, unlike for music, films, series, and books, where the average pirate tends to increase legal consumption while gradually decreasing illegal consumption, they found the average game pirate tends to increase or at least maintain illegal consumption over time. All economic models and estimations have uncertainties, but this suggests piracy results in user displacement and loss of revenue.

We identified many possible revenue streams for market operators and modders, such as the presence of traditional ads and sponsored posts. Through our analysis we showed a clear correlation between ad-library-modded and permission-modded apps. Previous sections observed code-modded apps are typically permission-modded and some ad-library-modded, while code-identical apps are permission and ad-library-identical. Furthermore, more than 1 in 5 code-modded apps with ad IDs present have had these IDs changed. We identified in previous sections ways in which original developers' and markets' revenue could be disrupted, finding 6 984 pirated apps (2 241 unique) with $2.28 billion in lifetime revenue, and 100k apps with IAPs (see §3.1.5–§3.2). These IAPs are typically offered for free in modded apps. Modded apps may cause the displacement of users from genuine markets and apps, thus affecting developer and market revenue and innovation.

5 SECURITY IMPLICATIONS OF MODDED APPS AND MARKETS

This section answers our final research question: "What are the security implications of installing apps from these markets?" App analysis and VirusTotal malware analysis results are combined to tackle this from the consumers' perspective. However, it is also important to reflect on the impact for the original app developers. In many of these modded apps API keys are still present, meaning the original developers are paying for cloud services or API services, etc. called by the app, e.g. Spotify, TikTok, online games (see §3.2). This has security and economic implications. Also, other users' security might be affected, as modded versions can change what users can or cannot see in social networking apps, for example. This might expose other users' information beyond their preferences or change what a user can store without being detected. It may also affect other users on the Internet, e.g. if the modded app embedded a botnet, which might contribute to attacks on company or governmental internet infrastructure [12, 40].

5.1 VirusTotal analysis methodology

The methodology to utilise VirusTotal results is based on previous approaches, where VirusTotal is queried with the hashes present in the entirety of ModZoo obtaining all existing analysis results. We do not upload any apps to VirusTotal because analysis results obtained after repeated scans have been found by previous studies to be more reliable than new results [45, 46, 56]. Similarly, the recommended threshold of around 10% of antivirus engines (AVs) flagging APKs as malicious is used (see §8). Furthermore, existing [47] and custom tools were used to obtain unified malicious labels. Thus, the size of the VirusTotal results also gives us an idea of how many of the modded and official apps in ModZoo have been previously scanned by users.

We use VirusTotal to get insights into the entire ModZoo dataset. More advanced techniques could be used on a random or selected sample of the dataset, but that is considered future work.

Modded apps are sometimes paired with the non-exact (latest-available) matches when the exact version of the app is unavailable in AndroZoo. We argue it is reasonable to compare modded apps to their non-exact, latest-available matches since the latest-available version should be just as safe if not safer than older versions. AndroZoo has most app versions, and a version number not in AndroZoo typically indicates a heavily modded app, as the AndroZoo authors have mitigations for robust scraping [2].

5.2 Malware, adware, and PUPs

The VirusTotal results cover 103 914 of the modded apps from our ModZoo dataset, as well as 71 670 Google Play apps coming from the AndroZoo dataset. This is because wee use previous analysis results exclusively due to their increased accuracy, as mentioned before. We found almost 9% of code-modded apps and only 0.5% code-identical apps coming from modded app markets were labelled malicious compared to only 0.9% of their currently-available Google Play counterparts, as shown in Figure 6. This section shows users are more vulnerable to malware, adware, potentially unwanted programs (PUPs) and other malicious programs when downloading and installing apps from modded markets.

In total, 167 273 apps were marked as undetected and 8 311 (4.7%) as malicious. Of these, 85.3% came from modded markets and the rest from Google Play (AndroZoo dataset). This translates as 6.82% of modded apps and 1.70% of Google Play apps in ModZoo classified as malicious. However, 8.59% of code-modded apps are malicious, against only 0.51% of code-identical apps.

Furthermore, the risk posed by the use of modded apps goes beyond this, since another important finding of this study is that many of the apps offered in modded markets are no longer offered in Google Play (even if they may still be available in the AndroZoo dataset). We find that 13.36% of the Google Play counterparts are no longer available as of March 2023. Of these, 6.72% are marked as malicious, compared to only 0.93% of those still available. Google Play Protect, Google's built-in malware scanning tool, analyses apps running in devices by default, but sideloaded apps are never dynamically analysed or uploaded to Google for testing unless users send them to Google [5, 22, 23]. Thus, Google Play users would be more protected against malicious apps than modded markets users. This is mainly due to Google Play's dynamic and static analysis of apps as well as their incident response and response to user reports. It should be noted that for hash-identical apps, the risk is obviously identical between Google Play and modded markets at the time they both host the app. Better security protections are present once the app is flagged and removed from Google Play in case it is malicious. These security checks set official markets such as Google Play apart.

Malicious apps are flagged as PUPs such as LuckyPatcher, used to modify Android apps. However, many are flagged with more worrying Trojan-like malware such as Andreed, Triada, RemoteCode, HiddenAds, Kyvu, (LuckyPatcher) IBGV, etc. and more general labels as 'downloader', 'virus', etc. Furthermore, 6 of the 30 most prominent labels are not present in Google Play apps at all, while most others have a significantly bigger presence in modded apps. A complete table of the distribution is included in Appendix B.

5.3 Permissions in code-modded apps

We studied the added permissions in code-modded and malicious code-modded apps, finding many dangerous permissions are added to code-modded apps. Malicious code-modded apps have a higher incidence of these, with 14.6% adding 'SYSTEM_ALERT_WINDOW' which allows creating windows on top of any other app, and 'READ_EXTERNAL_STORAGE', which allows access to other apps' files in the MediaStore. Malicious code-modded apps are twice as likely to request these, although there might be genuine need for some of them in modded apps. The following permissions are more than 4 times more likely to be used in malicious than non-malicious code-modded apps: 'WRITE_SETTINGS' which allows apps to read system settings, 'READ_LOGS' which is not to be used by third-party apps since Log entries can contain private user information, and 'CAMERA'. Other risky permissions such as 'ACCESS_COARSE_LOCATION' and 'ACCESS_FINE_LOCATION' are less common but are 8 times more likely to be added to malicious apps. See Appendix B for more details.

> We have established the security of modded markets is significantly lower than that of Google Play, with 8.6% of code-modded apps and 6.8% of apps hosted in them overall flagged as malicious by VirusTotal, against 0.9% of currently available Google Play apps and 1.7% of all Google Play apps included in this study. Furthermore, we found a high number of dangerous and risky permissions in code-modded apps, especially those classified as malicious.

6 ETHICS

Our institutional ethics committee approved the ethical considerations related to this study. Our ModZoo dataset containing Android apps and their metadata is collected on publicly available modded markets. Apps were only collected for analysis, not for use, and distributed only to other researchers after a thorough approval process, following previous approaches such as AndroZoo's. The majority of apps gathered are still freely available in the 13 markets scraped, for users and researchers to download without any login.

The only exception to the use of modded apps is the case study, and our ethical considerations and method are explained in this paragraph. In order to perform the case study of 28 modded apps we used testing devices and accounts exclusively, using no personal data and a SIM card obtained explicitly for this study. In order to minimise any negative effects on app developers and owners we only used the apps for the minimum amount of time required to test the modded functionalities and assert whether they were present or not. We did not make any modifications of our own to the apps. We only installed the 28 apps, as well as the 'APKMODY' and 'MODDROID' market apps needed to download apps from those two markets. Some apps permit or support interactions with other users online. For example, by sending coins to creators on TikTok

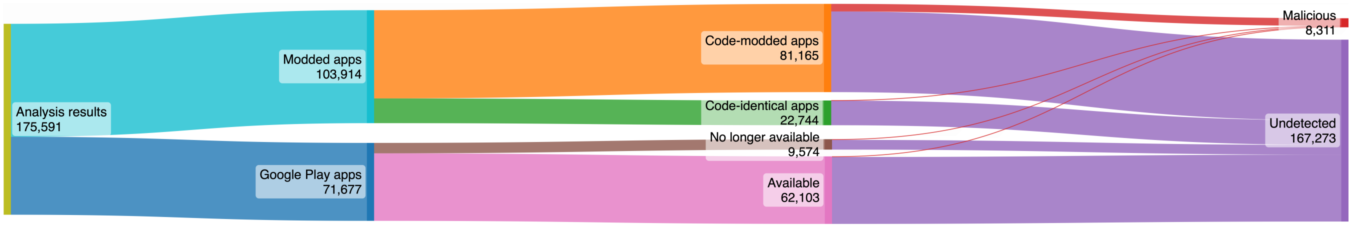

Figure 6: Distribution of malicious apps across Google Play and modded apps.

or playing games against other users online. We did not undertake any activities which we believed could affect other users, including sending messages to other users, looking up other user account details, etc. In order to explore interactions between two users of an online platform where needed, such as Truecaller search, we created two user accounts for this purpose. To not overburden individual modded markets, we downloaded single copies of apps one at a time through their websites or apps, minimising any additional load we may have placed on their service.

We also contacted all market operators for comment using their publicly available contact details obtained from their markets. In our communications we stated our affiliation and purpose for contact.

7 LIMITATIONS

This section briefly considers the limitations of this study.

The ModZoo dataset is potentially biased if the modded markets list is missing important markets. However, we have found that many of the markets towards the bottom of our ranking are no longer active, have very limited catalogues and few users, as discussed in §2.3. We might have missed more such markets, which would have no effect on our analysis results. However, we cross referenced our ranking with the Tranco ranking, confirming we included all the top modded markets as measured by Google Trends and the Tranco ranking in the months leading up to our study.

The analysis of advertising libraries in modded apps is potentially biased, as the accuracy of results depends on that of our ad libraries safelist. Thus, we have revised and expanded it periodically as more apps were analysed. Code obfuscation and code shrinking are techniques available to developers to make apps more secure, difficult to reverse engineer, and storage efficient. However, their use also undermines static analysis in the case of advertising libraries. Advertising IDs, permissions, and other parameters studied are not affected by this limitation. Several analysis tools have been proposed to study the libraries present in obfuscated apps [7, 39, 58, 59], however these require previously downloading all libraries of interest, or are not fast enough for this study considering the scale of the ModZoo dataset. Furthermore, code obfuscation, shrinking, and code optimisation are not enabled by default when using Android Studio [4]. Thus, having considered this limitation and the available solutions it can be argued that using a safelist is an acceptable compromise between accuracy, speed and scale.

Another limitation is the possibility of missing app versions, inherited via the use of AndroZoo [2]. Our results show this as a rare occurrence (see §2.6.1). Our last limitation is the way Google Play reports the IAPs prices as a range, not reporting how many there are or which are subscriptions, unlike other markets. This introduced imprecision in the study of IAPs at scale. IAPs were found not to cover the full price range reported by Google, further showing their inaccuracy.

8 RELATED WORK

Previous market analysis focused on Chinese app markets, analysing over 6 million Android apps and 16 markets [55]. They analysed inter-market similarity, their publishing behaviours, and prevalence of malicious, cloned, and fake apps, finding that Chinese markets performed substantially worse than Google Play. Our study instead focuses on modded apps, and mirrors some of their findings in terms of security and presence of pirated apps. However, we also explore operator and modder motivations and revenue streams. Others studied Android app attribution, and found the lack of metadata in AndroZoo a limitation to study authorship of apps at scale [25]. We found similarly, that metadata for apps no longer hosted in Google Play is lost, and our ModZoo dataset contains the app metadata from each market we scraped.

Other studies focused on Android VPN [28] and firmware over-the-air [8] apps, analysing their security, permissions and presence of malware through VirusTotal. Others compared the presence of trackers and permissions in paid and free games in Google Play, finding free games have 3.4 times more trackers on average and twice the number of dangerous permissions [35]. Another study analysed geoblocking and geographical differences in 26 countries' Google Play markets, finding apps are more often unavailable due to developer-introduced country restrictions than government take-downs [34]. None of them studied modded apps or third-party market security and motivators. Previous research found evidence that malicious apps lasted more than twice as long on Google Play than manufacturer-provided markets [48]. Our findings for modded markets suggest the opposite is true for modded app markets, as their operators lack the motivations that device manufacturers have to keep their platforms free of malicious apps. Our study is also novel in the mapping of third-party (modded apps) with their Google Play counterparts to compare ad libraries, permissions sets, latest available versions, and security implications. Others found repackaged apps are common in official markets, and are aimed at tricking unknowing users to think they are the official apps [31]. We found modded apps are instead usually advertised as modified versions (although not always, see §3.1.2). They found half of the 15k repackaged apps studied contained adware, against our 9% malicious code-modded apps. However, only 4% of them added permissions against 24% of code-modded apps in our study. They did not study ad IDs and their results are not reproducible due to the unavailability of one of the datasets used.

The relationship between malware and permissions declared by apps is explored, previous research separated prominent and trivial permissions [6], created permissions graphs and fuzzy clustering to find outliers [51, 53], and found malware-related permissions based on other datasets [3]. However, these approaches rely on existing datasets or do not publicly share their own. Unlike ours, they do not consider the connections between ad libraries and permissions changes. Furthermore, we share our ModZoo dataset. We also explored permissions added to malicious code-modded apps and found increased use of dangerous permissions. Others have focused on Manifest file features such as intents and context [38, 52], while some have added identification of packages and APIs used [1]. Static analysis is common to these large-scale approaches. Our study combines this with the analysis of the markets, and VirusTotal analysis.

Previous studies have used VirusTotal to analyse apps at scale. Zhu et al. surveyed 115 papers to identify common methodologies, and collected analysis results for a year [60, 61]. Although based on portable executables instead of APKs, they found 'trusted' engines do not perform well compared to the threshold approach consisting of labelling files malicious when flagged by at least N antivirus engines included in VirusTotal results. Most papers use thresholds to classify malicious files and the most popular threshold, $t = 1$, does not perform well [61]. They recommend a small threshold bigger than 1, such as 2 to 15. We have incorporated their insights into our 10% (5–7) threshold. Others worked on the security of third-party Android markets using VirusTotal and a threshold of 6 [9]. They found 5% to be malicious, and 31% had not been analysed before by VirusTotal, thus yielding no results. They analysed a very small sample compared to ours, downloading a total of 9k apps from 9 markets. Furthermore, we found a higher proportion of malicious apps in modded markets, and compared modded apps to their Google Play counterparts. Most approaches use a similar approach with different thresholds [41]. Others used weighted voting, relied on supervised learning, and used future results (after 4 weeks) as ground truth [30]. Others confirmed the increased accuracy of older results [45, 46]. Others created a dataset containing fewer than 10k malware samples [56], our ModZoo dataset contains 8.3k but is in continuous expansion.

Others focused on the misuse of native code libraries in Android apps [57], evaluating their approach on one third-party market using a relatively small sample. Furthermore, it required manual verification for some types of misuse, making it unsuitable at scale. Similarly, others identified harmful libraries in both Android and iOS based on VirusTotal results [11]. Our study links the presence of ad libraries and their changes with changes in ad IDs and permissions.

Previous research has explored the motivations of users to sideload, their knowledge of it, and other aspects [17]. Unlike our study, they do not consider modded apps nor motivations of the maintainers. Their questionnaire is run on a sample of Computer Science students and staff, as well as relevant sideloading and rooting Reddit forums users, thus providing limited data on the real-world occurrence of sideloading.

9 CONCLUSION

This paper presented the results of the first large-scale study into Android markets that offer modded apps. We explored the space through a large-scale technical analysis of 146k modded apps available on the 13 most popular markets. By comparing apps available on these markets to their Google Play counterparts, we demonstrated that the vast majority of apps were modified in one or more ways, including those labelled as unmodified. Furthermore, we have made the resulting dataset with almost 300k apps publicly available.

Currently, modded markets are likely to reduce the income of app developers and official markets due to the widespread and free availability of apps which usually charge on installation or to enable premium features. We found the majority of apps fell into the gaming category, however many other popular apps exist on these markets, including a modified version of TikTok advertised as offering free coins and a modified version of Spotify offering ad-free music without subscription. We also found modded apps included additional ad libraries and permissions, and 22% of modded apps had different ad IDs when compared with the Google Play version, suggesting ad revenue may be diverted away from the original app developer to a third party.

From the perspective of users, modded apps are advertised as offering new, desirable features, which our case studies suggest often, although not always, work. However, there are also significant negative effects. Users should be aware that using these markets supports third-parties unrelated to the genuine developers, in many cases diverting or curtailing the advertising, app purchase and IAPs revenue streams. While the presence of ad-free versions of apps is widely touted, we found fewer than 3% of modded apps had all ads and trackers removed. Approximately 9% of code-modded apps were marked as containing adware, grayware or Trojans by VirusTotal, 10 times the rate found in Google Play versions. Modded markets continue to host malicious apps that had been removed from Google Play for a long time. Furthermore, modded app users might put other users' privacy and security at risk, as modded apps might allow content supposed to be private to be viewable by other users, malware and spyware might make use of added permissions to access other users' private information, etc. We did not explore whether operating system features such as Google Play Protect provide sufficient protection against the increased risk associated with installing modded apps, something we defer to future work.

Developers should be aware of these markets and practices, and given more tools and support to report malicious versions of their apps. Developers, especially smaller ones, will have a hard time reporting misuse of their intellectual property since at present they would need to manually report multiple versions of their apps in more than 400 modded Android markets. We also know from contacting the market operators for comment that many contact forms and emails are answered with automated responses unrelated to the query, and many others do not even arrive at the market operators' inbox.

The question of whether mobile devices should allow the installation of apps outside the official market is under investigation by regulators in the EU [15, 16] and the UK [14]. One area of particular interest is the apparent trade-off between consumer protection and consumer choice. The iOS and Android ecosystems have taken different approaches: the former makes it very hard for the average consumer to install apps from outside the official market, while Android offers official support. Our work suggests regulators should consider what options there might be to counter the negative effects of access to modded markets and sideloading while protecting or enhancing user and app developer choice. The question does not reduce to whether sideloading should be allowed or not. In fact there are a range of options, including allowing sideloading while requiring apps to be tested and signed by an approved tester; requiring the distribution of alternative market apps through the official market in order to offer a pinch-point to support regulation; etc. A confounding factor is that large revenue streams are tied to the status quo where a significant proportion of the purchase price of paid apps and IAPs flow to the official market operator.

ACKNOWLEDGMENTS

We thank Richard Clayton for his help in setting up and maintaining the scraping machines, proxies and storage. We thank Stan (Jiexin) Zhang for his contributions to some of the scrapers. We are grateful to VirusTotal for the academic access to their API. We also thank colleagues and anonymous reviewers for their feedback. This work is supported by Nokia Bell Labs (for LAS and ARB) and the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 949127) (for HSD and AH). The views, opinions and findings in this paper are those of the authors and not necessarily those of our funders.

AVAILABILITY

ModZoo is available to any researcher and applications to access our dataset can be made following the process detailed in [https://www.cambridgecybercrime.uk/datasets.html]. Access requires contacting our institution first similar to previous datasets such as AndroZoo [2] due to copyright concerns. It would be relatively straightforward for malicious actors to start a modded app market with our dataset without this application process.

REFERENCES

[1] Yousra Aafer, Wenliang Du, and Heng Yin. 2013. DroidAPIMiner: Mining API-Level Features for Robust Malware Detection in Android. In *International conference on security and privacy in communication systems*. Springer, 86–103. https://doi.org/10.1007/978-3-319-04283-1_6.

[2] Kevin Allix, Tegawendé F. Bissyandé, Jacques Klein, and Yves Le Traon. 2016. AndroZoo: Collecting Millions of Android Apps for the Research Community. In *Proceedings of the 13th International Conference on Mining Software Repositories* (Austin, Texas) *(MSR '16)*. ACM, New York, NY, USA, 468–471. https://doi.org/10.1145/2901739.2903508 http://doi.acm.org/10.1145/2901739.2903508.

[3] Fahad Alswaina and Khaled Elleithy. 2018. Android Malware Permission-Based Multi-Class Classification Using Extremely Randomized Trees. *IEEE Access* 6 (2018), 76217–76227. https://doi.org/10.1109/ACCESS.2018.2883975.

[4] "Android.com". 2023. Android Studio: Shrink, obfuscate, and optimize your app. https://developer.android.com/studio/build/shrink-code.

[5] "Android.com". 2023. Google Play Protect: 2.5 Billion active devices. https://www.android.com/intl/en_us/play-protect/.

[6] AM Aswini and P Vinod. 2014. Droid permission miner: Mining prominent permissions for Android malware analysis. In *The Fifth International Conference on the Applications of Digital Information and Web Technologies (ICADIWT 2014)*. IEEE, 81–86. https://doi.org/10.1109/ICADIWT.2014.6814679.

[7] Michael Backes, Sven Bugiel, and Erik Derr. 2016. Reliable Third-Party Library Detection in Android and Its Security Applications. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security* (Vienna, Austria) *(CCS '16)*. Association for Computing Machinery, New York, NY, USA, 356–367. https://doi.org/10.1145/2976749.2978333 https://doi.org/10.1145/2976749.2978333.

[8] Eduardo Blázquez, Sergio Pastrana, Álvaro Feal, Julien Gamba, Platon Kotzias, Narseo Vallina-Rodriguez, and Juan Tapiador. 2021. Trouble Over-The-Air: An Analysis of FOTA Apps in the Android Ecosystem. In *2021 IEEE Symposium on Security and Privacy (SP)*. IEEE, 1606–1622. https://doi.org/10.1109/SP40001.2021.00095.

[9] William J. Buchanan, Simone Chiale, and Richard Macfarlane. 2017. A methodology for the security evaluation within third-party Android Marketplaces. *Digital Investigation* 23 (2017), 88–98. https://www.sciencedirect.com/science/article/pii/S1742287617300245 https://doi.org/10.1016/j.diin.2017.10.002.

[10] L. Ceci. 2023. TikTok IAP revenues worldwide 2023. https://www.statista.com/statistics/1377090/tiktok-worldwide-in-app-revenues-quarterly/.

[11] Kai Chen, Xueqiang Wang, Yi Chen, Peng Wang, Yeonjoon Lee, XiaoFeng Wang, Bin Ma, Aohui Wang, Yingjun Zhang, and Wei Zou. 2016. Following Devil's Footprints: Cross-Platform Analysis of Potentially Harmful Libraries on Android and iOS. In *2016 IEEE Symposium on Security and Privacy (SP)*. 357–376. https://doi.org/10.1109/SP.2016.29.

[12] Catalin Cimpanu. 2021. Android devices ensnared in DDoS botnet. https://www.zdnet.com/article/android-devices-ensnared-in-ddos-botnet/.

[13] "Cloudflare.com". 2023. Cloudflare DDoS Protection & Mitigation. https://www.cloudflare.com/en-gb/ddos/.

[14] The Competition and Markets Authority (CMA). 2022. Mobile ecosystems market study final report. https://www.gov.uk/government/publications/mobile-ecosystems-market-study-final-report.

[15] "Europa.eu". 2021. 2020/0374(COD) Digital Markets Act. https://oeil.secure.europarl.europa.eu/oeil/popups/ficheprocedure.do?reference=2020/0374(COD).

[16] "Europa.eu". 2022. Deal on Digital Markets Act: EU rules to ensure fair competition and more choice for users. https://www.europarl.europa.eu/news/en/press-room/20220315IPR25504.

[17] Craig Goodwin and Sandra Woolley. 2022. Sideloading: An Exploration of Drivers and Motivations. In *35th International BCS Human-Computer Interaction Conference 35*. 1–6. http://doi.org/10.14236/ewic/HCI2022.37.

[18] "Google.com". 2022. Changes to Google Play's billing system for users in Russia and Belarus. https://support.google.com/googleplay/android-developer/answer/11950272.

[19] "Google.com". 2022. Rest of the world. https://support.google.com/googleplay/android-developer/answer/12201481.

[20] "Google.com". 2022. Supported locations for distribution to Google Play users. https://support.google.com/googleplay/android-developer/answer/10532353.

[21] "Google.com". 2023. Learn about refunds on Google Play. https://support.google.com/googleplay/answer/2479637.

[22] "Google.com". 2023. On-device protections. https://developers.google.com/android/play-protect/client-protections.

[23] "Google.com". 2023. Use Google Play Protect to help keep your apps safe and your data private. https://support.google.com/googleplay/answer/2812853.

[24] Marie Charlotte Götting. 2023. Spotify's revenues from 2012 to 2022 by segment. https://www.statista.com/statistics/245125/revenue-distribution-of-spotify-by-segment/.

[25] Kaspar Hageman, Álvaro Feal, Julien Gamba, Aniketh Girish, Jakob Bleier, Martina Lindorfer, Juan Tapiador, and Narseo Vallina-Rodriguez. 2023. Mixed Signals: Analyzing Software Attribution Challenges in the Android Ecosystem. *IEEE Transactions on Software Engineering* 49, 4 (2023), 2964–2979. https://doi.org/10.1109/TSE.2023.3236582

[26] Simon Hill. 2014. Freemium apps: necessary evil or plain greedy? *Android Authority* (21 05 2014). https://www.androidauthority.com/freemium-model-good-bad-thing-384124/.

[27] iBotPeaches. 2023. Apktool. https://github.com/iBotPeaches/Apktool.

[28] Muhammad Ikram, Narseo Vallina-Rodriguez, Suranga Seneviratne, Mohamed Ali Kaafar, and Vern Paxson. 2016. An Analysis of the Privacy and Security Risks of Android VPN Permission-enabled Apps. In *Proceedings of the 2016 internet measurement conference*. 349–364. https://doi.org/10.1145/2987443.2987471.

[29] Mansoor Iqbal. 2023. Spotify revenue and Usage Statistics (2023). https://www.businessofapps.com/data/spotify-statistics/.

[30] Alex Kantchelian, Michael Carl Tschantz, Sadia Afroz, Brad Miller, Vaishaal Shankar, Rekha Bachwani, Anthony D. Joseph, and J. D. Tygar. 2015. Better Malware Ground Truth: Techniques for Weighting Anti-Virus Vendor Labels. In *Proceedings of the 8th ACM Workshop on Artificial Intelligence and Security* (Denver, Colorado, USA) *(AISec '15)*. Association for Computing Machinery, New York, NY, USA, 45–56. https://doi.org/10.1145/2808769.2808780 https://doi.org/10.1145/2808769.2808780.

[31] Kobra Khanmohammadi, Neda Ebrahimi, Abdelwahab Hamou-Lhadj, and Raphaël Khoury. 2019. Empirical study of android repackaged applications. *Empirical Software Engineering* 24 (2019), 3587–3629. https://doi.org/10.1007/s10664-019-09760-3.

[32] John Koetsier. 2017. App developers losing $3-4 billion annually thanks to 14 billion pirated apps. https://www.forbes.com/sites/johnkoetsier/2017/07/24/app-developers-losing-3-4-billion-annually-thanks-to-14-billion-pirated-apps/.

[33] John Koetsier. 2023. TikTok earned $205 million more than Facebook, Twitter, snap and Instagram combined on in-app purchases in 2023. https://www.forbes.com/sites/johnkoetsier/2023/03/01/tiktok-earned-205-million-more-than-facebook-twitter-snap-and-instagram-combined-on-in-app-purchases-in-2023/.

[34] Renuka Kumar, Apurva Virkud, Ram Sundara Raman, Atul Prakash, and Roya Ensafi. 2022. A Large-scale Investigation into Geodifferences in Mobile Apps. In *31st USENIX Security Symposium (USENIX Security 22)*. USENIX Association, Boston, MA, 1203–1220. https://www.usenix.org/conference/usenixsecurity22/presentation/kumar.

[35] Pierre Laperdrix, Naif Mehanna, Antonin Durey, and Walter Rudametkin. 2022. The Price to Play: A Privacy Analysis of Free and Paid Games in the Android Ecosystem. In *Proceedings of the ACM Web Conference 2022* (Virtual Event, Lyon, France) *(WWW '22)*. Association for Computing Machinery, New York, NY, USA, 3440–3449. https://doi.org/10.1145/3485447.3512279 https://doi.org/10.1145/3485447.3512279.

[36] Victor Le Pochat, Tom Van Goethem, Samaneh Tajalizadehkhoob, Maciej Korczyński, and Wouter Joosen. 2019. Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation. In *Proceedings of the 26th Annual Network and Distributed System Security Symposium*. Internet Society, 1–15. https://doi.org/10.14722/ndss.2019.23386.

[37] Rimantas Leonavičius. 2021. How to access Google Play app store while in China. https://cybernews.com/resources/how-to-access-google-play-app-store-while-in-china/.

[38] Xiang Li, Jianyi Liu, Yanyu Huo, Ru Zhang, and Yuangang Yao. 2016. An Android malware detection method based on AndroidManifest file. In *2016 4th International Conference on Cloud Computing and Intelligence Systems (CCIS)*. IEEE, 239–243. https://doi.org/10.1109/CCIS.2016.7790261.

[39] Ziang Ma, Haoyu Wang, Yao Guo, and Xiangqun Chen. 2016. LibRadar: Fast and Accurate Detection of Third-Party Libraries in Android Apps. In *Proceedings of the 38th International Conference on Software Engineering Companion* (Austin, Texas) *(ICSE '16)*. Association for Computing Machinery, New York, NY, USA, 653–656. https://doi.org/10.1145/2889160.2889178 https://doi.org/10.1145/2889160.2889178.

[40] Chandraveer Mathur. 2023. A new Android botnet trojan is out for your banking data. https://www.androidpolice.com/android-botnet-trojan-steal-banking-data/.

[41] Peng Peng, Limin Yang, Linhai Song, and Gang Wang. 2019. Opening the Blackbox of VirusTotal: Analyzing Online Phishing Scan Engines. In *Proceedings of the Internet Measurement Conference* (Amsterdam, Netherlands) *(IMC '19)*. Association for Computing Machinery, New York, NY, USA, 478–485. https://doi.org/10.1145/3355369.3355585 https://doi.org/10.1145/3355369.3355585.

[42] Sarah Perez. 2023. Spotify's third-party billing option has now reached over 140 global markets. https://techcrunch.com/2023/01/31/spotifys-third-party-billing-option-has-now-reached-over-140-global-markets/.

[43] Joost Poort, João Quintais, Martin A van der Ende, Anastasia Yagafarova, and Mathijs Hageraats. 2018. Global Online Piracy Study. *Amsterdam Law School Research Paper* 2018-21 (2018). https://doi.org/10.2139/ssrn.3224323.

[44] Rafael Rob and Joel Waldfogel. 2006. Piracy on the high C's: Music downloading, sales displacement, and social welfare in a sample of college students. *The Journal of Law and Economics* 49, 1 (2006), 29–62. https://doi.org/10.3386/w10874.

[45] Aleieldin Salem. 2021. Towards Accurate Labeling of Android Apps for Reliable Malware Detection. In *Proceedings of the Eleventh ACM Conference on Data and Application Security and Privacy* (Virtual Event, USA) *(CODASPY '21)*. Association for Computing Machinery, New York, NY, USA, 269–280. https://doi.org/10.1145/3422337.3447849

[46] Aleieldin Salem, Sebastian Banescu, and Alexander Pretschner. 2021. Maat: Automatically Analyzing VirusTotal for Accurate Labeling and Effective Malware Detection. *ACM Trans. Priv. Secur.* 24, 4, Article 25 (jul 2021), 35 pages. https://doi.org/10.1145/3465361

[47] Marcos Sebastián, Richard Rivera, Platon Kotzias, and Juan Caballero. 2016. AVclass: A Tool for Massive Malware Labeling. In *Research in Attacks, Intrusions, and Defenses*, Fabian Monrose, Marc Dacier, Gregory Blanc, and Joaquin Garcia-Alfaro (Eds.). Springer International Publishing, Cham, 230–253. https://software.imdea.org/~juanca/papers/avclass_raid16.pdf.

[48] Yun Shen, Pierre-Antoine Vervier, and Gianluca Stringhini. 2022. A Large-scale Temporal Measurement of Android Malicious Apps: Persistence, Migration, and Lessons Learned. In *31st USENIX Security Symposium (USENIX Security 22)*. 1167–1184. https://www.usenix.org/conference/usenixsecurity22/presentation/shen-yun.

[49] Michael D Smith and Rahul Telang. 2012. Assessing the Academic Literature Regarding the Impact of Media Piracy on Sales. *SSRN Electronic Journal* (2012). https://dx.doi.org/10.2139/ssrn.2132153.

[50] Kristin Snyder. 2023. The secret to TikTok's success with in-app purchases. https://dot.la/tiktok-revenue-2659494404.html.

[51] Karina Sokolova, Charles Perez, and Marc Lemercier. 2017. Android application classification and anomaly detection with graph-based permission patterns. *Decision Support Systems* 93 (2017), 62–76. https://doi.org/10.1016/j.dss.2016.09.006.

[52] Guillermo Suarez-Tangil, Santanu Kumar Dash, Mansour Ahmadi, Johannes Kinder, Giorgio Giacinto, and Lorenzo Cavallaro. 2017. DroidSieve: Fast and Accurate Classification of Obfuscated Android Malware. In *Proceedings of the Seventh ACM on Conference on Data and Application Security and Privacy*. 309–320. https://doi.org/10.1145/3029806.3029825.

[53] Altyeb Altaher Taha and Sharaf Jameel Malebary. 2021. Hybrid Classification of Android Malware Based on Fuzzy Clustering and the Gradient Boosting Machine. *Neural Computing and Applications* 33, 12 (jun 2021), 6721–6732. https://doi.org/10.1007/s00521-020-05450-0 https://doi.org/10.1007/s00521-020-05450-0.

[54] "Wallhax.com". 2021. What are ESP cheats? how ESP hacks work in multiplayer games! https://wallhax.com/what-are-esp-cheats/.

[55] Haoyu Wang, Zhe Liu, Jingyue Liang, Narseo Vallina-Rodriguez, Yao Guo, Li Li, Juan Tapiador, Jingcun Cao, and Guoai Xu. 2018. Beyond Google Play: A Large-Scale Comparative Study of Chinese Android App Markets. In *Proceedings of the Internet Measurement Conference 2018 (IMC '18)*. Association for Computing Machinery, 293–307. https://doi.org/10.1145/3278532.3278558.

[56] Haoyu Wang, Junjun Si, Hao Li, and Yao Guo. 2019. RmvDroid: Towards A Reliable Android Malware Dataset with App Metadata. In *2019 IEEE/ACM 16th International Conference on Mining Software Repositories (MSR)*. 404–408. https://doi.org/10.1109/MSR.2019.00067 https://doi.org/10.1109/MSR.2019.00067.

[57] Qing Wang, Juanru Li, Yuanyuan Zhang, Hui Wang, Yikun Hu, Bodong Li, and Dawu Gu. 2018. NativeSpeaker: Identifying Crypto Misuses in Android Native Code Libraries. In *International Conference on Information Security and Cryptology*. Springer, 301–320. https://doi.org/10.1007/978-3-319-75160-3_19.

[58] Yan Wang, Haowei Wu, Hailong Zhang, and Atanas Rountev. 2018. ORLIS: Obfuscation-Resilient Library Detection for Android. In *Proceedings of the 5th International Conference on Mobile Software Engineering and Systems* (Gothenburg, Sweden) *(MOBILESoft '18)*. Association for Computing Machinery, New York, NY, USA, 13–23. https://doi.org/10.1145/3197231.3197248 https://doi.org/10.1145/3197231.3197248.

[59] Jiexin Zhang, Alastair R. Beresford, and Stephan A. Kollmann. 2019. LibID: Reliable Identification of Obfuscated Third-Party Android Libraries. In *Proceedings of the 28th ACM SIGSOFT International Symposium on Software Testing and Analysis* (Beijing, China) *(ISSTA 2019)*. Association for Computing Machinery, New York, NY, USA, 55–65. https://doi.org/10.1145/3293882.3330563 https://doi.org/10.1145/3293882.3330563.

[60] Shuofei Zhu, Jianjun Shi, Limin Yang, Boqin Qin, Ziyi Zhang, Linhai Song, and Gang Wang. 2020. Measuring and Modeling the Label Dynamics of Online Anti-Malware Engines. In *29th USENIX Security Symposium (USENIX Security 20)*. USENIX Association, 2361–2378. https://www.usenix.org/conference/usenixsecurity20/presentation/zhu.

[61] Shuofei Zhu, Ziyi Zhang, Limin Yang, Linhai Song, and Gang Wang. 2020. Benchmarking Label Dynamics of VirusTotal Engines. In *Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security* (Virtual Event, USA) *(CCS '20)*. Association for Computing Machinery, New York, NY, USA, 2081–2083. https://doi.org/10.1145/3372297.3420013 https://doi.org/10.1145/3372297.3420013.

A LIST OF KEYWORDS

The full list of keywords to identify the Android markets is presented in Table 2.

B DETAILED ANALYSIS RESULTS

More detailed results of the VirusTotal threat labels are included in Table 3. The occurrence of the rest of labels after 'solid' is 3 or lower, and thus not included.

The top 10 modification descriptions in modded markets app descriptions are included in Table 4 together with their counts.

The 30 permissions most commonly added to malicious code-modded apps are shown in Table 5, together with their occurrence in malicious and all code-modded apps and their protection level category. Google distinguishes different protection level categories for permissions [1]: 'dangerous', 'normal', and 'signature'. Some permissions are classified as "Not for use by third-party applications", we categorise them as very dangerous in Table 5. An example of such

[1] https://developer.android.com/reference/android/Manifest.permission

Table 3: Most common VirusTotal threat labels and their distribution

Threat Label	Total	Modded Apps	Google Play
None	163 788	93 541	70 247
andreed	3 830	3 761	69
grayware	3 586	3 052	534
fyben	2 111	2 078	33
adware	1 011	490	521
downloader	465	413	52
androeed	183	181	2
kyvu	72	19	53
grayware:tool	51	37	14
triada	28	7	21
remotecode	27	24	10
dataeye	24	14	10
hiddenads	23	18	5
luckypatcher	21	21	0
ibgv	18	18	0
spyware	18	16	2
tencentprotect	18	10	8
fleeceware	17	8	9
boogr	14	13	1
wamod	14	14	0
virus	13	13	0
appflood	11	8	3
igexin	10	7	3
virtualapp	7	5	2
mcalprotect	6	3	3
remco	6	3	3
revpn	6	5	1
agentsmith	5	4	1
apkprotector	5	5	0
clicker	5	4	1
miniupnp	5	3	2
subspod	5	0	5
utilcode	5	3	2
browserad	4	4	0
fghg	4	4	0
loead	4	4	0
powerofr	4	3	1
solid	4	3	1

Table 4: List of top 10 modifications present in the modded markets.

Modification	Count
mod money	19 206
unlimited money	7 202
free shopping	3 680
original	3 563
premium unlocked	1 257
full version	1 095
mod menu	1 020
mod: premium	895
mod: unlocked	730
unlimited coins	702
no ads	289

Table 2: List of keywords used to identify the Android markets.

Keyword	Language
Android app stores	English/Chinese/Hindi/Russian
free Android app store	English/Chinese/Hindi/Russian
mod apk	English/Chinese/Hindi/Russian
download premium apk	English/Chinese/Hindi/Russian
download paid apps free	English/Chinese/Hindi/Russian
mod Android	English/Chinese/Hindi/Russian
mod games	English/Chinese/Hindi/Russian
paid apps for free	English/Chinese/Hindi/Russian
premium apps for free	English/Chinese/Hindi/Russian
unlocked android apps	English/Chinese/Hindi/Russian
unlocked android games	English/Chinese/Hindi/Russian
mod apps for free	English/Chinese/Hindi/Russian
YouTube mod	English
Spotify mod	English
Truecaller mod	English

permissions is 'READ_LOGS', which allows apps to "read the low-level system log files", which may contain users' private information. Signature permissions can be "signature|privileged|development", "signature|setup|appop|installer|pre23|development", etc. However, we use 'signature' as the category for readability below. Signature permissions can only be used by apps signed with the same certificate as the app that declared it. Thus, many of these are usually reserved for system apps and similar or apps by the same developer sharing functionality or data. Finally, some categories are not categorised by Google even though they are relatively popular in both Google Play and modded apps. We classify these as 'uncategorised' even though they are probably 'normal'. It is worth noting that although 'normal' permissions do not require user confirmation in-app like 'dangerous' permissions, they are still potentially dangerous as users of modded markets are not presented with accurate information of the 'normal' or 'dangerous' permissions used by apps.

They are all android.permission.{} except 'net.dinglisch.android.tasker.PERMISSION_RUN_TASKS' AND 'com.android.launcher.permission.INSTALL_SHORTCUT', abbreviated 'tasker.PERMISSION_RUN_TASKS' and 'launcher.INSTALL_SHORTCUT', respectively. We classified them as dangerous and normal, respectively, although they are not included in Google's classifications. We also abbreviated 'android.permission.REQUEST_IGNORE_BATTERY_OPTIMIZATIONS' and 'DOWNLOAD_WITHOUT_NOTIFICATION' as 'REQUEST_IGNORE_BATTERY_OPT' and 'DOWNLOAD_WITHOUT_NOTIF' for readability.

Table 5: Top 30 added permissions in malicious code-modded and code-modded apps and their category.

Permission	Category	Malicious code-modded (%)	Code-modded (%)
SYSTEM_ALERT_WINDOW	signature	14.60	8.72
READ_EXTERNAL_STORAGE	dangerous	9.01	4.10
BLUETOOTH_ADMIN	normal	7.39	1.65
BLUETOOTH	normal	7.19	1.62
WRITE_SETTINGS	signature	7.05	1.70
CHANGE_WIFI_STATE	normal	6.68	1.55
FLASHLIGHT	normal	6.61	1.56
USE_FINGERPRINT	normal	6.59	1.57
READ_LOGS	very dangerous	6.50	1.54
REQUEST_IGNORE_BATTERY_OPT	normal	6.50	1.56
READ_SETTINGS	uncategorised	6.50	1.56
tasker.PERMISSION_RUN_TASKS	dangerous	6.50	1.56
CAMERA	dangerous	6.38	1.54
REQUEST_INSTALL_PACKAGES	signature	5.62	1.07
VIBRATE	normal	4.38	0.88
WRITE_EXTERNAL_STORAGE	dangerous	3.82	3.01
ACCESS_WIFI_STATE	normal	3.41	0.86
QUERY_ALL_PACKAGES	normal	2.81	6.66
GET_TASKS	normal	1.96	0.74
READ_PHONE_STATE	dangerous	1.38	0.63
RESTART_PACKAGES	deprecated	1.06	0.13
KILL_BACKGROUND_PROCESSES	normal	1.01	0.11
RECEIVE_BOOT_COMPLETED	normal	0.90	0.15
CHANGE_NETWORK_STATE	normal	0.85	0.10
BATTERY_STATS	signature	0.78	0.09
ACCESS_COARSE_LOCATION	dangerous	0.76	0.09
BROADCAST_STICKY	normal	0.76	0.07
ACCESS_FINE_LOCATION	dangerous	0.67	0.09
launcher.INSTALL_SHORTCUT	normal	0.67	0.09
DOWNLOAD_WITHOUT_NOTIF	normal	0.53	0.18