Which Android App Store Can be Trusted in China?

Yi Ying Ng[1], Hucheng Zhou[2], Zhiyuan Ji[3], Huan Luo[1], Yuan Dong[1]
[1]*Department of Computer Science and Technology, Tsinghua University*
[2]*Microsoft Research,*
[3]*High Technology Research and Development Center, Chinese Ministry of Science and Technology*
yiying.ng@gmail.com, huzho@microsoft.com, jzy@hrtdc.com, luohuan07@gmail.com, dongyuan@tsinghua.edu.cn

***Abstract*—China has the world's largest Android population with 270 million active users. However, Google Play is only accessible by about 30% of them, and third-party app stores are thus used by 70% of them for daily Android apps (applications) discovery. The trustworthiness of Android app stores in China is still an open question. In this paper, we present a comprehensive study on the trustworthy level of top popular Android app stores in China, by discovering the identicalness and content differences between the APK files hosted in the app stores and the corresponding official APK files. First, we have selected 25 top apps that have the highest installations in China and have the corresponding official ones downloaded from their official websites as oracle; and have collected total 506 APK files across 21 top popular app stores (20 top third-party stores as well as Google Play). Afterwards, APK identical checking and APK difference analysis are conducted against the corresponding official versions. Next, assessment is applied to rank the severity of APK files. All the apps are classified into 3 severity levels, ranging from `safe` (identical and higher level), `warning` (lower version or modifications on resource-related files) to `critical` (modifications on permission file and/or application codes). Finally, the severity levels contribute to the final trustworthy ranking score of the 21 stores. The study indicates that about only 26.09% of level APK files are safe, 37.74% of them are at warning level, and 36.17% of them are surprisingly at critical level. We have also found out that 10 (about 2%) APK files are modified and re-signed by unknown third-parties. In addition, the average trustworthy ranking score (47.37 over 100) has also highlighted that the trustworthy level of the Android app stores in China is relatively low. In conclusion, we suggest Android users to download APK files from its corresponding official websites or use the highest ranked third-party app stores; and we appeal app stores to ensure all hosting APK files are trustworthy enough to provide a "safe-to-download" environment.**

***Keywords*-Android, app store, APK, trustworthy, severity ranking**

I. Introduction

Smart phone is gaining its popularity with no sign of slowing down. Android [1], an open source mobile operating system first contributed by Google, is evolved as the top shipped smartphone operating system with about 79% of market share in 2013 [2]. In fact, the number of Android apps had exceeded one million with total of 1,019,909 in 1st Jan, 2014 [3]. However, Google still provides the Android developers with the flexibility of having open distribution options [4], which means that the Android developers are free to distribute their developed apps in any intended approaches, including publishing into third-party app stores. This freedom raises the growing trend of third-party app stores that supports the demand of personalized and localized apps discovery and recommendations. Especially in China where more than 70% of 270 million of daily active Android smart phones are estimated to be lack of Google services [5] and they are thus replaced by third-party alternatives, such freedom then opened up a big skeptical for the trustworthiness of various APK (Android Package) files that are distributed across third-party Android app stores in China.

Android developers have incentives to distribute their APK files via almost all of the popular app stores, in order to promote their apps and to reach for potential users as much as possible. The more app stores they chosed, the higher the potential returns would be. However, more update efforts are requires accordingly, which could be painful. Consider that one developer has already submitted one app to 10 third-party app stores. An update even with minor change should be uploaded to all of the app stores one by one that at certain time windows the app is not updated in some app stores. There would be some app stores without receiving the update accidentally. As a result, users would get the outdated apps, which is acceptable that if the update is a bug fix. Even all third-party app stores are not overlooked and are up-to-date, it is still worth questioning whether or not the app store would secretly modify the app or others would upload the app with the same name.

According to the report [6], the distribution volume of mobile applications in China had exceeded 180 billions since 3rd quarter, 2013, and the distribution volume via app stores are covering more than 80%. Since third-party app stores are gaining popularity among mobile users, the trustworthiness of the APK files from these top popular third-party app stores do be worthy of further examination. Our studies try to answer the questions: all of the downloadable APK files from the popular Android app stores, are they the same and trustworthy? Is it safe to download from these Android app stores? Is there any side-effect to download from these Android app stores? The answers are meaningful to disclose the actual situation or risk that Android mobile users in China are facing.

We present our study to rank the trustworthy level of 20 popular Android app stores in China, as well as Google Play [7] - the official centralized Android apps store. We select 25 apps with highest downloading rates in China as our sample targets. We have downloaded the identified 25 APK files from the selected 21 Android app stores; only one APK file is selected if an app store has more than one apps with the same name, and it is possible that some apps are missed in certain app stores. There are 506 APK files in total. We also downloaded the corresponding official versions from the official websites, which are considered to be the oracle for our study. For instance, Facebook app can be installed through Google Play directly, while it is also downloadable from Facebook official website with manual installation. All the APK files for the study are collected between 14th - 15 Jan, 2014.

To support our study, we develop a tool, `TrasRank`, to perform APK `identical checking` against the corresponding official version once a APK file is downloaded. TrasRank compares the APK package name, version code and SHA256 checksum between the APK files. Afterwards, TrasRank conducts APK `difference analysis` to identify the differences with the official version. followed by `trustworthy alignment` to assess the severity level for trustworthy ranking. All the apps are classified into 3 severity levels, ranging from `safe` , `warning` to `critical`. It is safe only if it is identical, higher version or zero-modification (SHA256-mismatched APK files may turn out have the same content); it is at warning level if it has lower version or some resource-related files are modified but does not affect the functionality; while it is critical if modifications on the permission file and/or application codes are changed.

The study indicates that about only 26.09% of level APK files are safe, 37.74% of them are at warning level, and 36.17% of them are surprisingly at critical level. In fact, 5.46% of critical labelled APK files are due to difference in signature and 94.54% of critical labelled APK files are due to modification on application codes (classes.dex), library files (.so) and/or permission file (AndroidManifest.xml). We have also found out that 10 (about 2%) APK files are modified and re-signed by unknown third-parties. In addition, the average trustworthy ranking score (47.37 over 100) has also highlighted that the trustworthy level of the Android app stores in China is relatively low. These facts have tremendously signalled the risk of downloading APK files from third-party app stores without proper attention.

The rest of this paper is organized as follows: We first present the methodology of TrasRank in Section II, followed by detailed analysis in Section III. Finally we discuss related works (Section IV), draw conclusions and discuss future work (Section V).

Table I
SELECTED 20 POPULAR ANDROID APP STORES IN CHINA

Group	App Store
Pre-installed	HiMarket[16]
Android Device Manufacturer	Samsung Apps [17], LeStore [18]
Mobile Operator	Mobile Market [19], WoStore [20], TianYi [21]
Search Engine Company	Baidu [22], Sogou [23]
E-commerce provider	Taobao [24]
Pure Store	91 [25], 360 [26], Wandoujia [27], YingYongBao [28], Anzhi [29], Gfan [30], Appchina [31], Mumayi [32], DangLe [33], UC [34], CNMO [35]

II. TRANSRANK EVALUATION METHOD AND DESIGN

A. Evaluation Setup

First, we have identified the 25 top ranking apps from the “Hot Download” section in Baidu [8], which is the dominant search company in China, which owns 1.3 billion of daily active users [5]. According to Statista [9], 25 is the average number of installed apps per smartphone. The displayed total downloads for these identified influential apps are ranging from 10 million to 5 billion.

In term of Android app stores selection, we have included Google Play and the top most influential representatives from 6 different groups (Table I), including default pre-installed third-party app stores and those which are maintained by Android mobile device manufacturer, mobile operator, search engine pioneer, B2C pioneer and popular app stores in China which claimed [10], [11], [12], [13], [14], [15] to be fulfilling at least 1 of the following criteria:

- High Registered User: Ranging from 40 million to 11.4 billion of registered users.
- High Visit Rates: Ranging from 5 to 20 million for daily visit.
- High Download Rates: Ranging from 1 billion to 5 billion of monthly downloads.
- High Total App Collections: 1 million of app collections.

To ensure the validity of our study results, we emphasize on accuracy throughout our data collection processes. To enhance accuracy, we are using the app names as the keywords to search across 21 Android app stores since it is similar to how typical users perform their searches in app stores. Whenever there are multiple similar returned results for the keywords, we chose by prioritizing official label (claim to be the apps which corresponding to its official copy), followed by similarity of the apps name and the keywords used, last updated date and number of downloads.

In the following sections, we discuss on the design and implementations of TrasRank which is written with Java programming language. The main features of TrasRank are consist of APK Identical Check, APK Difference Check and

Trustworthy Alignment which were conducted sequentially to analyse the collected 506 APK files from 21 sampled Android app stores. APK Identical Check is designed to determine if 2 APK files are identical or not while APK Difference Check seeks to identify the content differences between 2 comparing APK files. At the end, Trustworthy Alignment is carried out to align these APK files to their corresponding severity label.

B. APK Identical Check

APK Identical Check is designed with the aim to uncover the identical level, version variation level (updated or outdated) and false-returned level (returning invalid or inaccurate apps by using app name as search keywords) of APK files which are hosting on different app stores compared to the official APK files. This identicalness check is done by comparing the package name, version code and SHA256-checksum values as SHA256 is one of the strongest hash functions [36]; and can be used to check for the identicalness of file. In result, all the APK files will be assigned with a label accordingly based on the comparison result, including I (identical), M (SHA256-mismatched), V (lower version), W (higher version), N (package name mismatched due to false-returned) and E (corrupted file).

To complete the functionality of APK Identical Check, we have utilized 2 tools, which are:

- Android Asset Packaging Tool (aapt) for package information retrieval, including package name and package version
- Java MessageDigest class for APK file SHA256-checksum value retrieval

For version checking, we are checking upon the package version code because the version name of the APK file is not firm enough to determine the actual version of the APK files; it is just a string that represents the release version of the application code but the system does not use this value for any internal purpose [37]. Version code of the APK files on the other hand is used by applications to check for upgrade or downgrade relationship.

C. APK Difference Check

APK Difference Check is designed with the aim to identify the differences in content of 2 comparing APK files. To achieve this goal, we rely on the information which stores inside the APK manifest file. Since APK file is a JAR file by nature, there will be a default manifest file storing under META-INF/MANIFEST.MF; which is automatically generated whenever a JAR file is created. This manifest file stores information about all the other files that are packaged in the archive (the APK file) [38]. Besides, APK Difference Check is designed to compare on the digital signature of APK files. For any difference in digital signature simply means that it was changed by unknown third-party. Similarly, we labelled the APK content differences into 6 categories, including:

- C: Modified content on non-resources related content, such as AndroidManifest.xml, classes.dex and content under /lib folder. We consider this difference category is relatively severe compared to the rest as it may affect how the app is behaving
- D: File removed
- F: Unable to locate MANIFEST.MF file
- R: Modified content on resources-related files, such as images, videos and content which stores under the folder of res/, assets/ and resources.arsc file
- S: Signature changed
- X: Newly added content

We used the same way as reading a JAR file to retrieve the content of manifest file (META-INF/MANIFEST.MF). In order to read the APK certificate (META-INF/CERT.RSA), we used X509Certificate class in Java. The outcome of APK Difference Check is capable to uncover what are the differences in term of contents between 2 APK files. And, if there is a modification, it is managed to tell which are the files.

D. Trustworthy Alignment and Trustworthy Ranking

Trustworthy Alignment is designed to align the collected APK files from the 21 sampled Android app stores in China into corresponding severity label to support the final generation of Trustworthy Ranking based on the analysis results from APK Identical Check and APK Difference Check. We propose 3 severity levels with 9 labels by considering their potential threat level to user devices, which are arranged in an ascending order as follow:

1) Safe:

- Identical: These APK files are same as the official APK files and therefore are perfectly safe to install.
- Higher version: Some developers prefer to update their apps in the third-party app stores before their official websites. Since these APK files are signed by the same developer, they are safe to install too.
- Zero-modification APK files: These APK files were first labelled as SHA256-mismatched. However, there is no modification was found in term of content after applying APK Difference Check; finally mapped as zero-modification as they do not have any visible modifications which carry impact to the current apps. Hence, they are safe to install too.

2) Warning:

- Corrupted: These APK files are corrupted may either due to poor server communication or they are corrupted by nature. Since corrupted APK file cannot even being installed into user device, its threat is the least within this category.
- Modifications on resource-related content: For APK files which contain modifications on resource-related

content, it is risky in the sense that it may be used to convey different message after replacing the data files and graphics in the APK file.

- Lower version: For lower version labelled APK files which signed by the same developer may potentially contain bug fixes in the later version. Hence, it is still risky to install outdated APK files.
- False-returned: For false-returned APK files, it is due to improper management in handling user search queries, apps and package information. In fact, it is fairly unacceptable because these downloaded APK files are different from what users are expecting.

3) Critical:

- Modifications on critical files: Modifications on permission file, application codes and library files are doubtlessly a risk as changes in these files may totally change how the app works.
- Signature changed: The most risky APK files are those which are labelled with signature change. Definitely, these APK files are not the app which users are searching and they are mostly camouflaged APK files.

If the APK file falls under multiple severity labels, they are labelled to their highest severity label which further determine their final severity level (safe, warning or critical). Based on these results, we apply a simple calculation to obtain the Trustworthy Ranking Score for the selected 21 Android app stores in China.

The Ranking Score ($R(s)$) for app store s is calculated through (1), where Positive Score ($P(s)$) is calculated through (2) while Negative Score ($Ne(s)$) is calculated through (3).

$$R(s) = \frac{P(s) - Ne(s) + min}{max + min} \times 100 \tag{1}$$

where $min = 100$, $max = 100$. It means for the best case where all the APK files are all perfectly safe to be installed in app store s, the highest $R(s)$ will be 100. On the other hand, for the worst case where all the APK files have their digital signature changed (re-signed by unknown third-party), the lowest $R(s)$ will be -100.

$$P(s) = \frac{I(s) + W(s) + Z(s)}{T(s)} \times 100 \tag{2}$$

where $I(s)$ is the total number of APK files which are identical (I) labelled for app store s; $W(s)$ is the total number of APK files which are higher version (W) labelled for app store s; $Z(s)$ is the total number of APK files which are zero-modification (Z) labelled for app store s; $T(s)$ is the total number of downloaded APK files for app store s.

$$Ne(s) = Wa(s) + Cr(s) \tag{3}$$

where $Wa(s)$ is the total negative score from Warning severity level for app store s which is calculated through (4); $Cr(s)$ is the total negative score from Critical severity level for app store s through (5).

$$Wa(s) = \frac{E(s) \times j + Re(s) \times k + (V(s) + N(s)) \times l}{T(s)} \tag{4}$$

where $E(s)$ is the total number of APK files which are corrupted (E); $Re(s)$ is the total number of APK files which are labelled with modifications on resource-related files; $V(s)$ is the total number of APK files which are lower version (V) labelled; $N(s)$ is the total number of APK files which are package name mismatched (N) labelled due to false-returned; $T(s)$ is the total number of downloaded APK files for app store s. As different severity label owns different significance, we assign different values to different severity labels where $j = 10$, $k = 20$, $l = 30$.

$$Cr(s) = \frac{C(s) \times m + S(s) \times n}{T(s)} \tag{5}$$

where $C(s)$ is the total number of APK files which contain modifications on critical files (permission file, application code and library files); $S(s)$ is the total number of APK files which are labelled with signature changed (S); $T(s)$ is the total number of downloaded APK files for app store s. We assign different values to different severity labels to represent its significance where $m = 60$, $n = 100$.

III. Analysis And Discussions

Under this section, we present our evaluation results based on TrasRank analysis outcomes after applying APK Identical Check and APK Difference Check to our sampled APK files. In details, the selected 21 Android app store in China are Google Play [7], HiMarket [16], Samsung [17], Lenovo [18], Mobile Market [19], WoStore [20], TianYi [21], Baidu [22], Sogou [23], Taobao [24], 91 [25], 360 [26], Wandoujia [27], YingYongBao [28], Anzhi [29], Gfan [30], Appchina [31], Mumayi [32], DangLe [33], UC [34] and CNMO [35]. The 531 APK files are collected between 14th - 15th Jan, 2014 and the distribution of collected APK files across 21 Android app stores in China is listed in Table II.

A. APK Identical Check Analysis

From the APK Identical Check result, we are managed to conclude that the average identicalness of APK files across 21 Android app stores in China is really low (6.92%). And, the occurrence of having identical APK files comparing to the official APK files is only 80.95% which implies that there are app stores contain not even a single identical APK file. APK Identical Check evaluates identicalness by checking through APK package name (N), version (W: higher; V: low) and SHA256-checksum of the APK file (M). So, if the comparing APK file yielded the exact same value for the above criteria and it is not corrupted (E); it is then considered as identical (I) to the official APK file. The overall statistics for the APK Identical Check is presented

Table II
Distribution of Collected APK Files across 20 Android App Store in China and Google Play

App Store	ID	Total	App Store	ID	Total
360	360	25	91	91	25
Anzhi	anzhi	24	Appchina	appchina	25
Baidu	baidu	24	CNMO	cnmo	25
DangLe	dcn	24	TianYi	dianxin	25
GFan	gfan	25	Google Play	google	23
HiMarket	hiapk	25	LeStore	lenovo	24
WoStore	liantong	24	Mumayi	mumayi	25
YingYongBao	qq	25	Samsung	samsung	19
Sogou	sogou	23	TaoBao	taobao	25
UR	uc	25	Wandoujia	wandoujia	25
Mobile Market	yidong	21			

in Table III and we have come out with the followings conclusions:

- Version (V): The analysis data shows that there is 13.04% of the collected APK files are not the latest version. As users may not be aware of the exact version of the APK files, downloading an outdated APK file is very risky as there are possibilities where identified bugs are only fixed in the later version.
- Version (W): It is interesting for us to find out that there is 15.02% of the APK files are higher version compared to their official APK file. Since the analysis data has shown that the percentage of having at least 1 higher version of APK files is 100%, it simply means the app update behaviour in China is less consistent and reliable. The apps which fall under this category are including cn.kuwo.player [39], com.sina.weibo [40], com.youkuphone [41], tv.pps.mobile [42]. In short, it has drawn a fact that version management across all types of app stores, including the official hosting APK files are under a messy untraceable condition.
- Package Name (N): APK files are labelled under this category is due to a false-returned, which is referring to the APK files which have a different package name compared to the official ones. We found that there is an average 1.58% of APK files fell under this category. Anyhow, we have performed a close check for the 8 false-returned apps and realize that it is fortunately enough to see 5 of the APK files were being labelled so is because of the limitation of using keyword search. For instance, we used "UC Browser" as our keywords to search for the app but it happened that there are several variations of UC Browser apps and we have chosen another app instead of the intended one.

  Besides, it may also due to that the Android developer prefers to create different package name for the same app for different app stores. For instance, iQiYi official package name is com.qiyi.video but in Google Play its package name is com.qiyi.video.market while in Samsung app store its package name is com.qiyi.video.Samsung. The details can be viewed in Table IV. However, the remaining 3 APK files are camouflaged files and are signed by unknown third-party! This has proved that it is rather risky to download from third-party app stores without proper knowledge to detect the package information.
- Corrupted (E): It may due to download failure which caused by network failure or broken APK files. 3.75% of APK files are labelled under this category.
- SHA256-Mismatched (M): Among these evaluation criteria, majority (59.68%) does not pass while comparing the SHA256-checksum. This has indicated that these APK files own the same package name and version but they are not the same as the official APK files which may contain modifications. In order to answer to the doubt of where are the differences between these APK files and the official APK files, therefore we have APK Difference Check and the analysis data is discussed in the following section.

Table III
APK Identical Check Analysis Result

Label	Percentage (%)	Occurrence Percentage (%)
Identical (I)	6.92	80.95
SHA256 (M)	59.68	100
Lower (V)	13.04	61.9
Higher (W)	15.02	100
Name (N)	1.58	23.81
Error (E)	3.75	14.29

Occurrence Percentage (%) refers to the occurrence of the corresponding label across 21 sampled Android app stores.

B. APK Difference Check Analysis

TrasRank APK Difference Check labelled the content difference with 6 labels, which are unable to locate manifest file (F), newly added file (X), file removed (D), modifications on resources-related files (R), modifications on critical files (C) and changed of signature (S). From APK Difference Check analysed result, there is only 280 APK files contain differences in content as opposed to the labelled 302 SHA256-mismatched (M) APK files in APK Identical Check. For the 22 zero-modifications, they own the same number of files and each file is having the same SHA-1 digest in their manifest file compared to the official APK files. This has proved that SHA256-checksum is not firm enough to determine the dissimilarity between APK files.

After conducted file compare (fc command in Windows) command, we noticed the differences between these zero-modification APK files and the official APK files are due to encoding issue as the comparison results yielded unknown

Table IV
CAUSE ANALYSIS FOR PACKAGE NAME MISMATCHED IN APK IDENTICAL CHECK

Official Package Name	App Store	Downloaded Package Name	Cause
com.qiyi.video	google	com.qiyi.video .market	Different package name for different app stores
com.qiyi.video	samsung	com.qiyi.video .Samsung	Different package name for different app stores
com.qzone	taobao	com.qzonele	Camouflaged app, improper signature CN=Unknown, OU=Unknown, O=Unknown, L=Unknown, ST=U...
com.taobao .taobao	dianxin	cn.zhui .client546668	Camouflaged app, improper signature CN=Zhui.CN, OU=Byban Ltd., O=Byban Ltd., L=Shangha...
com.UCMobile	samsung	com.UCMobile .intl	Limitation of app name search
com.UCMobile	taobao	com.UCMobile .ac	Limitation of app name search
com.youdao.dict	yidong	com.xing.you .dao	Camouflaged app, improper signature CN=huangfaxing, OU=huangfaxing, O=huangfaxing, L=z...
com.youku .phone	samsung	com.youku .phone.samsung .market	Different package name for different app stores

characters and symbols. The affected apps are included Sogou Input Method (com.sohu.inputmethod.sogou) and WeChat (com.tencent.mm) which both are heavily dealing with texts. Since these zero-modification APK files do not contain any visible modification which brings any impact to the apps, they are then further mapped back to zero-modification label.

In fact, we applied APK Difference Check onto all the APK files from the Android app stores to check on their digital signatures. However, in term of content differences, we focus on the APK files which are labelled with SHA256-mismatched in APK Identical Check because it is making more sense to compare the contents if 2 APK files are having the same package name and version. To summarize from Table V, the majority fall under resources-related modifications (82.14%), which are referring to the changes are made under the folder of assets/, res/ and resources.arsc. These folders are used to store resource files, such as graphic, video and data (i.e. JSON, XML) in general cases. Even though these resources may not have direct influence to the application code level, it is still worth attention to. It is because if images are replaced in the apps, it may be conveying different messages, including violence. Second majority of difference group for SHA256-mismatched APK files is on critical files (non-resource related modifications) and these changes are including AndroidManifest.xml (which stores the permission details), .so files (library files) and classes.dex (application code); and it is covering 63.93% of total SHA256-mismatched labelled APK files. Lastly, there is 20% of the APK files are having new content in the comparing APK files. For the APK files which fell under the above mentioned difference groups, they are potentially outdated apps, if they are signed by the same authorized developer.

Table V
APK DIFFERENCE CHECK (SHA256-MISMATCHED ANALYSIS)

Label	Total Apps	Percentage (%)
Critical (C)	179	63.93
Removed (D)	42	15
Resource (R)	230	82.14
Signature (S)	6	2.14
New (X)	56	20

No APK file was unable to locate its MANIFEST.MF file (F).

Table VI
APK FILES WITH SIGNATURE CHANGED

Package Name	App Store	Label (APK Identical Check)
com.baidu.appsearch	mumayi [32]	SHA256-mismatched (M)
com.chinamworld.main	mumayi [32]	SHA256-mismatched (M)
com.moji.mjweather	mumayi [32]	SHA256-mismatched (M)
com.qiyi.video	mumayi [32]	SHA256-mismatched (M)
com.tencent.mm	mumayi [32]	SHA256-mismatched (M)
com.UCMobile	mumayi [32]	SHA256-mismatched (M)
com.sina.weibo	mumayi [32]	Higher version (W)

However, it is especially risky to have APK files which are signed by unknown third parties. After applying APK Difference Check on the sampled APK files from app stores, unfortunately, we notice that apart from the previous mentioned 3 camouflaging apps (Table IV), there are another 7 APK files containing different certificate public key. The details have shown that they are signed by unknown third-party with the signature of 'EMAILADDRESS=android@android.com, CN=Android, OU=A..'. The affected APK files are being listed in Table VI.

Up to this point, the evidence is firm enough to conclude to the statement that there is almost no Android app store in China is risk-free for APK file download and the users are suffering from downloading outdated apps, non-targeting apps (due to false-returned), buggy apps and altered camouflaging apps. From the developer perspectives, apart from influencing his reputation due to outdated, buggy or altered apps; the involved update effort is especially painful too.

C. Trustworthy Alignment and Trustworthy Ranking of Android App Stores in China

Since the version inconsistency is an obvious problem for now across various Android app stores in China, to support our ranking calculation, we have made 3 assumptions:

- APK files from official websites are valid and trustworthy
- APK files which are labelled with identical (I), higher version (W) and zero-modification (Z) are valid and trustworthy
- APK files which are labelled with SHA256-mismatched (M), lower version (V), package name mismatched (N) and corrupted (E) are not trustworthy

By using our analysis result from TrasRank Trustworthy Alignment which aligned every collected APK file from the sampled 20 Android app stores in China and Google Play to its corresponding severity label; we manage to propose a trustworthy ranking for these app stores as in Table VII through a relatively simple calculation. Almost all of the Android app stores are open for APK submission from developers while 1 of them do not provide any information for open submission, which is China Unicom [20]. And, majority of these app stores solely rely on the developers to update their APK files, except for wandoujia [27] which contains internal system to crawl latest APK files from other external sources. Besides, we noticed that TaoBao [24] allows APK files sharing from general users who are not the developer and/or owner of the APK files. Nevertheless, manual APK auditing processes may take up 1 to 7 days. These are the potential reasons that causes the version inconsistency across these Android app stores.

To recall, Trustworthy Alignment labelled the APK files into 3 severity levels with 9 severity labels:

- Safe: Identical, higher version and zero-modification
- Warning: Corrupted APK file, modifications on resource-related content, lower version and false-returned
- Critical: Modifications on critical files and signature

Based on the result after applying Trustworthy Alignment, the highest Ranking Score is only 54.4 out of 100 and the average Ranking Score is only 47.37 out of 100; implies that trustworthy level of top 21 Android app stores in China is fairly low. From Table VII, it shows that not all app stores are managed to download the complete set of the top 25 apps. Only 15 app stores are managed to download Baidu Mobile Assistant *(com.baidu.appsearch)* which has more than 1 billions of accumulated downloads [8] to prove its popularity. Thus, it can be concluded that almost no app store in China owns a complete set of Android apps in the market. For instance, developers may not want to join competitors app stores for distributing their apps. And, even Google Play does not contain all the popular apps in China and the trustworthy score is not high too. In short, it is rather risky to download these apps form third-party app stores in China without proper attention and knowledge. The summary of app distributions across 3 severity levels for 21 Android app stores in China together with their Ranking Score can be found in Fig 1.

Table VII
TRUSTWORTHY RANKING OF 21 ANDROID APP STORES IN CHINA WITH APK COLLECTIONS AND UPDATE MECHANISM SUMMARY

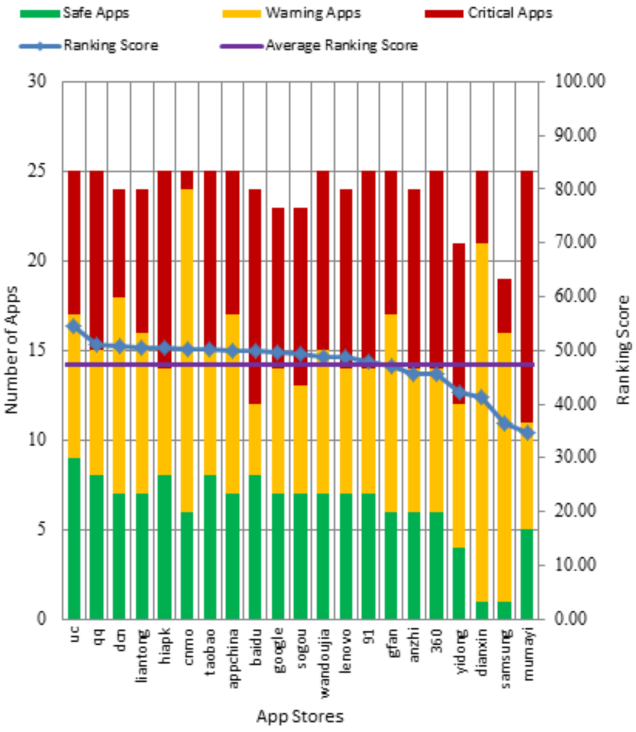
Ra-nk	Store ID	Score	No. App	Safe	Warning				Critical		Upd.
					E	R	V	N	C	S	
1	uc	54.40	25	9	0	4	4	0	8	0	-
2	qq	51.20	25	8	0	7	0	0	10	0	-
3	dcn	50.83	24	7	0	3	8	0	6	0	-
4	liantong	50.42	24	7	1	5	3	0	8	0	-
5	hiapk	50.40	25	8	0	6	0	0	11	0	-
6	cnmo	50.20	25	6	0	1	17	0	1	0	-
6	taobao	50.20	25	8	0	6	0	1	9	1	<5
8	appchina	50.00	25	7	0	8	2	0	8	0	-
9	baidu	50.00	24	8	0	4	0	0	12	0	<5
10	google	49.57	23	7	0	3	3	1	9	0	-
11	sogou	49.35	23	7	0	5	1	0	10	0	-
12	wandoujia	48.80	25	7	0	8	0	0	10	0	
13	lenovo	48.75	24	7	0	5	2	0	10	0	<2
14	91	48.00	25	7	0	7	0	0	11	0	-
15	gfan	47.20	25	6	0	7	4	0	8	0	<2
16	anzhi	45.63	24	6	0	3	5	0	10	0	1-3
17	360	45.60	25	6	0	8	0	0	11	0	<2
18	yidong	42.14	21	4	1	7	0	0	8	1	-
19	dianxin	41.20	25	1	17	0	3	0	3	1	-
20	samsung	36.32	19	1	0	1	11	3	3	0	<7
21	mumayi	34.60	25	5	0	3	3	0	7	7	<2

E: Total corrupted labelled APK files.
R: Total modifications on resource-related labelled APK files.
V: Total lower version labelled APK files.
N: Total package name mismatched labelled APK files (false-returned).
C: Total modifications on critical files labelled APK files.
S: Total signature changed labelled APK files.
Upd.: Update days.

IV. RELATED WORKS

Android landscape examination. Prior large scale works have included the examination of the landscape and impact of Android apps plagiarism [43] which involves 17 Android markets. Besides, there are many related works reveal security risks of Android platform. For instance, Comdroid [44] performed an analysis on the potential vulnerability in Android inter-apps communications and [45], [46] examined the information leaks within and between apps. AppFence [47] revised the apps permission granting to access device information and suggesting 2 different approaches to protect user sensitive data. Another related work is DroidMOSS [48], [49] which conducted a study on 6 unofficial Android apps marketplaces and managed to identify repackaged apps. TrasRank on the other hand examined the impact of app distributions across top popular app stores in China and

Figure 1. APK Collections And App Severity Level Distribution of 21 Android App Stores in China



provide answer to the trustworthy level of these popular app stores by providing facts and figures, which was left unanswered before our works.

Malicious apps detection. Kirin [50] presented a security framework that detects potential unsafe Android apps at installation time and block identified applications which contain insecure policy configurations. VirusMeter [51] discussed malware detection through the monitoring of power consumptions. There is also related work [52] that performed Android apps security study which involves 1,100 top free apps that mainly gathers the Android security issues. In fact, 360 [53] in China is placing efforts in providing free mobile security protection and detection where users are allowed to upload files (including APK files) to the server for security scanning. But, no relevant paper was published regarding its implementations. Besides, DroidRanger [54] conducted a large-scale study for malicious apps that experiments with 204,040 apps across 5 Android markets; making a conclusion that current marketplaces are functional and relatively healthy. However, TrasRank does not conclude with the same viewpoint since version inconsistency, modifications on critical files and the issue of having camouflaged apps with signature changes across third-party app stores in China have proved that the trustworthy level is fairly low.

V. Conclusion

In this paper, we assessed the trustworthy level of 25 top downloading apps in 20 popular third-party Android app stores in China and Google Play, which covers almost all of the top app stores in China. From our analyzed result against 506 collected APK files, it can be concluded that the trustworthy level of the sampled Andoird app stores in China is fairly low. Safe labeled APK files are only 26.09%, while warning labeled APK files are 37.74% and critical labeled APK files are 36.17%. We believe our presented app trustworthy assessment tool, `TrasRank`, could be also adopted by app store to improve their trustworthy level. Currently, we cannot answer the specific causes of these trustworthy issues, which could be caused by bad updating policies developers adopted, or ”evil” app store itself who secretly modified the uploaded apps, or even the third-party developers who could upload the apps with the same name to cheat the app stores and customers. We treat them as future work.

It is always risky to install unsafe APK files as the device is exposed to dangerous security and privacy threats. However, by considering the limitation of keyword searches in app stores where multiple similar APK files with similar icons are returned; it increases the user efforts to identify the intended one. Besides, they are also exposed to the risk of downloading APK files which are incompatible to the user Android devices and harmful camouflaging apps. Even though it is noticeable that all these app stores are trying to maintain a clean and safe-to-download environment, the efforts and times involved to verify and audit are still painful and expensive. In fact, this work is contributing with analysis facts and figures to firmly conclude that the current trustworthy level of Android app stores in China is rather low, messy version management and hardly to claim any Android app stores in China as “safe-to-download”.

For current moment, we suggest the Android users to download APK files from the official websites or choose to download from the highest ranked Android app stores in China. On the other hand, web masters should make sure that all the published APK files in their official websites are up-to-date. At the same times, we appeal the third-party app stores to ensure all hosting APK files are trustworthy enough to provide a “safe-to-download” environment whereby APK files which are signed by unknown third-party should not be listed in the app stores. In future, we seek to implement a detection tool for suggesting trustworthy download sources for new APK file installation and/or updates. In addition, we expect the detection tool to be capable of identifying the differences between 2 APK files which can be used to classify its harmful level to user device.

Acknowledgment

The authors would like to thank Mr. Sun Chan of Intel for useful discussions about software analysis in this paper. The authors are supported in part by National Natural Science Foundation of China (No. 61170051, No. 61272086), Hi-Tech Research and Development Program of China (No. 2009AA011902, No. 2013AA013702) and National Core-High-Base Project of China (No. 2012ZX01039-004-08). Any opinions, findings, and contributions contained in this document are those of the authors and do not reflect the views of these agencies.

References

[1] Google, “Android,” http://www.android.com/, 2014.

[2] PRNewswire, “Strategy analytics: Android captures 79 percent share of global smartphone shipments in 2013,” http://www.prnewswire.com/news-releases/strategy-analytics-android-captures-79-percent-share-of-global-smartphone-shipments-in-2013-242563381.html, 2013.

[3] AppBrain, “Number of available android applications,” http://www.appbrain.com/stats/number-of-android-apps, 2014.

[4] Google, “Open distribution,” http://developer.android.com/distribute/open.html, 2014.

[5] K. Hong, “Report: China has 270m android users that’s nearly 30% of global android activations to date,” http://thenextweb.com/asia/2013/11/27/report-china-has-270-million-android-users-nearly-30-of-global-android-activations-to-date/, 2013.

[6] A. International, “China mobile application distribution market analysis report for 3rd quarter, 2013,” http://www.pmbaike.com/pmbaike/3194.html, 2013.

[7] Google, “Google play,” http://play.google.com, 2014.

[8] Baidu, “Baidu,” http://as.baidu.com/a/rank?pre=web_am_header, 2014.

[9] Statista.com, “The average smartphone user has installed 26 apps,” http://www.statista.com/topics/1002/mobile-app-usage/chart/1435/top-10-countries-by-app-usage/, 2013.

[10] mAPPn Inc., “About gfan.com,” http://www.mappn.com/about/, 2014.

[11] Baidu Baike, “Appchina,” http://baike.baidu.com/view/6064469.htm, 2014.

[12] Mumayi, “Android app annual award,” http://www.mumayi.com/special/2013/pingxuan/, 2014.

[13] Downjoy, “About downjoy,” http://www.d.cn/about_us.html, 2014.

[14] GameRes, “Uc app store,” http://www.gameres.com/msg_220821.html, 2013.

[15] Cnmo, “About cnmo,” http://www.cnmo.com/webcenter/about.html, 2014.

[16] Hiapk, “Himarket,” http://apk.hiapk.com/, 2014.

[17] Samsung, “Samsung,” http://samsungapps.sina.cn/main/getMain.as, 2014.

[18] Lenovo, “Lenovo,” http://app.lenovo.com/, 2014.

[19] China Mobile, “China mobile,” http://mm.10086.cn/, 2014.

[20] China Unicom, “China unicom,” http://store.10010.com/WoApp/chseWoList/init?categoryTag=L005, 2014.

[21] China Telecom, “China telecom,” http://www.189store.com/index.php?, 2014.

[22] Baidu, “Baidu,” http://as.baidu.com/, 2014.

[23] Sogou, “Sogou,” http://app.sogou.com/, 2014.

[24] TaoBao, “Taobao,” http://app.taobao.com/index.htm?spm=0.0.0.0.sdiwMc, 2014.

[25] 91, “91,” http://apk.91.com/, 2014.

[26] 360, “360,” http://zhushou.360.cn/, 2014.

[27] Wandoujia, “Wandoujia,” http://www.wandoujia.com/apps, 2014.

[28] Tencent, “Tencent,” http://android.myapp.com/, 2014.

[29] AnZhi, “Anzhi,” http://www.anzhi.com/, 2014.

[30] Gfan, “Gfan,” http://www.gfan.com/, 2014.

[31] AppChina, “Appchina,” http://www.appchina.com/, 2014.

[32] Mumayi, “Mumayi,” http://www.mumayi.com/, 2014.

[33] Downjoy, “Downjoy,” http://www.d.cn/, 2014.

[34] uc.cn, “Uc app store,” http://apps.uc.cn/, 2014.

[35] Cnmo, “Cnmo,” http://app.cnmo.com/, 2014.

[36] Moveable-type.co.uk, “Sha-256 cryptographic hash algorithm,” http://www.movable-type.co.uk/scripts/sha256.html, 2014.

[37] Android, “Versioning your application,” http://developer.android.com/tools/publishing/versioning.html, 2014.

[38] Oracle, “Understanding the default manifest,” http://docs.oracle.com/javase/tutorial/deployment/jar/defman.html, 2014.

[39] Kuwo.cn, “Kuwo music box apk download page,” http://shouji.kuwo.cn/, 2014.

[40] Weibo.com, “Weibo apk download page,” http://m.weibo.com/web/cellphone.php, 2014.

[41] Youku.com, “Youku apk download page,” http://mobile.youku.com/index/wireless, 2014.

[42] PPStream Inc., “Pps apk download page,” http://dl.pps.tv/pps_android_download.html, 2014.

[43] C. Gibler, R. Stevens, J. Crussell, H. Chen, H. Zang, and H. Choi, “Adrob: Examining the landscape and impact of android application plagiarism,” in *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys ’13. ACM, 2013, pp. 431–444.

[44] E. Chin, A. P. Felt, K. Greenwood, and D. Wagner, “Analyzing inter-application communication in android,” in *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services*, ser. MobiSys ’11. ACM, 2011, pp. 239–252.

[45] W. Enck, P. Gilbert, B.-G. Chun, L. P. Cox, J. Jung, P. McDaniel, and A. N. Sheth, “Taintdroid: An information-flow tracking system for realtime privacy monitoring on smartphones,” in *Proceedings of the 9th USENIX Conference on Operating Systems Design and Implementation*, ser. OSDI’10. USENIX Association, 2010, pp. 1–6.

[46] M. Dietz, S. Shekhar, Y. Pisetsky, A. Shu, and D. S. Wallach, “Quire: Lightweight provenance for smart phone operating systems,” *CoRR*, vol. abs/1102.2445, 2011.

[47] P. Hornyack, S. Han, J. Jung, S. Schechter, and D. Wetherall, “These aren’t the droids you’re looking for: Retrofitting android to protect data from imperious applications,” in *Proceedings of the 18th ACM Conference on Computer and Communications Security*. ACM, 2011, pp. 639–652.

[48] W. Zhou, Y. Zhou, X. Jiang, and P. Ning, “Detecting repackaged smartphone applications in third-party android marketplaces,” in *Proceedings of the Second ACM Conference on Data and Application Security and Privacy*. ACM, 2012, pp. 317–326.

[49] J. Crussell, C. Gibler, and H. Chen, *Attack of the Clones: Detecting Cloned Applications on Android Market*. Springer-Verlag Berlin Heidelberg, 2012, vol. 7459.

[50] W. Enck, M. Ongtang, and P. Mcdaniel, “Mitigating android software misuse before it happens,” 2008.

[51] L. Liu, G. Yan, X. Zhang, and S. Chen, “Virusmeter: Preventing your cellphone from spies,” in *12th International Symposium, RAID 2009, Saint-Malo, France, September 23-25, 2009. Proceedings*, 2009, pp. 244–264.

[52] W. Enck, D. Octea, P. Mcdaniel, and S. Chaudhuri, “A study of android application security,” in *In Proc. USENIX Security Symposium*, 2011.

[53] 360, “360 mobile security scanning,” http://scan.shouji.360.cn/, 2014.

[54] Y. Zhou, Z. Wang, W. Zhou, and X. Jiang, “Hey, you, get off of my market: Detecting malicious apps in official and alternative android markets,” in *In Proceedings of the 19th Annual Network and Distributed System Security Symposium, NDSS 12*, 2012.