

Analyzing User Data Privacy in Android Mobile OS

by

Abdul Momen Miah

18201009

MD. Akhtaruzzaman Ashik

18201006

Pranab Chakma

18201092

Shoumya Dipta Baidya

18201032

Musharrat Zaman

18201080

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering

School of Data and Sciences

Brac University

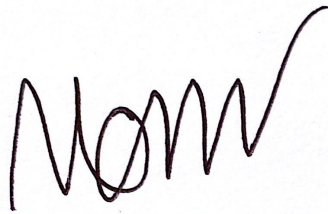
January 2023

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

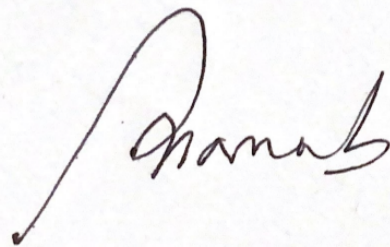
Student's Full Name & Signature:



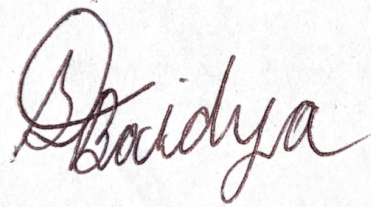
Abdul Momen Miah
18201009



MD. Akhtaruzzaman Ashik
18201006



Pranab Chakma
18201092



Shoumya Dipta Baidya
18201032



Musharrat Zaman
18201080

Approval

The thesis titled “Analyzing User Data Privacy in Android Mobile OS” submitted by

1. Abdul Momen Miah(18201009)
2. MD. Akhtaruzzaman Ashik(18201006)
3. Pranab Chakma(18201092)
4. Shoumya Dipta Baidya(18201032)
5. Musharrat Zaman(18201080)

Of Fall, 2022 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science and Engineering on January 26, 2023.

Examining Committee:

Supervisor:
(Member)



Dr. Amitabha Chakrabarty
Associate Professor
Department of Computer Science and Engineering
Brac University

Program Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam
Professor
Department Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi
Associate Professor
Department Computer Science and Engineering
Brac University

Abstract

Uses of smartphones are increasing rapidly more than anything, and so are the chances of risking our personal information. We constantly install various apps on our mobile phones for different purposes. Data privacy is one of the significant concerns regarding using any mobile app. Users have to allow too many permissions for those apps to use them. Many apps collect too much data from the users, even though they are not required to be functional. These data collections create a massive data breach from the users' end. Also, much information is collected in the background without any concern from the users, only to track users' behavior and provide more personalized advertising. Over 70% of the applications we use, either directly or through third-party libraries, get users' sensitive information. We will analyze the amount of information collected by the apps and the ad companies using AdGuard. Also, we will build a unique data set and analyze the ad/tracking aggressiveness of the ad companies based on the user's Mobile Brand, and OS version and so on.

Keywords: Data Privacy; Android Data Privacy; Ad Tracker; Google Ads; Meta Ads; Ad Block

Acknowledgement

We would first like to express our sincerest gratitude to the Great Allah for providing us with the courage and guidance necessary to complete our thesis without interruption. It is through His blessings that we have been able to achieve this significant milestone in our academic journey.

Dr. Amitabha Chakrabarty, our thesis supervisor, has played a vital role in our research and writing processes. His insightful advice, encouragement, and support were invaluable in facilitating the completion of our thesis. We are deeply appreciative of his mentorship and dedication.

We would also like to extend our heartfelt thanks to Mr. Shahriar Hossain for his constant support and assistance throughout our journey. His invaluable help and willingness to assist us whenever we required it have been instrumental in our success. Without his support, this accomplishment would not have been possible.

Lastly, we would like to express our profound gratitude to our parents for their unwavering love and prayers. Our academic journey has been built upon their support and encouragement. This thesis would not have been achievable without their guidance and support, for which we will always be grateful.

Table of Contents

Declaration	i
Approval	ii
Abstract	iii
Acknowledgment	iv
Table of Contents	v
List of Figures	vii
List of Tables	1
1 Introduction	2
1.1 Research Problem	3
1.2 Research Objectives	4
2 Literature Review	5
2.1 Android App Permission Settings	5
2.2 Existing Strategies for Addressing Tracking Risks	6
2.2.1 End user privacy controls	6
2.2.2 OS Regulations	6
2.2.3 Legal Regulation	6
2.3 Risks of Using Android Third Party Libraries	7
2.4 Private Data Leakage in Mobile Devices	8
2.5 Various Encryption for Enhancing Securities	8
2.6 Privacy-Preserving Apps	9
2.7 Exposing the Data Sharing Practices	9
2.8 Related Works	10
3 Methodology	14
3.1 Pre-Processing Data	16
3.2 Processing Data	16
4 Result & Analysis	30
5 Conclusion	46

List of Figures

3.1	The Flow Chart of the Proposed Generalized Process	15
3.2	Total Web Request vs Total Blocked Web Request for Different Ad Companies	19
3.3	Comparison of Total Web Request and Total Blocked Web Request in Android 7	19
3.4	Comparison of Total Web Request and Total Blocked Web Request in Android 8	20
3.5	Comparison of Total Web Request and Total Blocked Web Request in Android 9	20
3.6	Comparison of Total Web Request and Total Blocked Web Request in Android 10	21
3.7	Comparison of Total Web Request and Total Blocked Web Request in Android 11	21
3.8	Comparison of Total Web Request and Total Blocked Web Request in Android 12	22
3.9	Version VS Blocked Web Request	22
3.10	24 hours Total Web Request and Blocked Web Request Comparison in Honor	23
3.11	24 hours Total Web Request and Blocked Web Request Comparison in HUAWEI	23
3.12	24 hours Total Web Request and Blocked Web Request Comparison in Infinix	24
3.13	24 hours Total Web Request and Blocked Web Request Comparison in Lenovo	24
3.14	24 hours Total Web Request and Blocked Web Request Comparison in Nokia	25
3.15	24 hours Total Web Request and Blocked Web Request Comparison in Nothing	25
3.16	24 hours Total Web Request and Blocked Web Request Comparison in OnePlus	26
3.17	24 hours Total Web Request and Blocked Web Request Comparison in OPPO	26
3.18	24 hours Total Web Request and Blocked Web Request Comparison in POCO	27
3.19	24 hours Total Web Request and Blocked Web Request Comparison in Realme	27
3.20	24 hours Total Web Request and Blocked Web Request Comparison in Redmi	28

3.21	24 hours Total Web Request and Blocked Web Request Comparison in Samsung	28
3.22	24 hours Total Web Request and Blocked Web Request Comparison in Sony	29
3.23	24 hours Total Web Request and Blocked Web Request Comparison in Xiaomi	29
4.1	Version-Wise Blocked Percentage (Android v7, 8 & 9)	31
4.2	Version-Wise Blocked Percentage (Android v10, 11, 12 & 13)	32
4.3	Brand-Wise Blocked Percentage (Honor, HUAWEI, Infinix & Lenovo)	32
4.4	Brand-Wise Blocked Percentage (Nokia, Nothing, OnePlus & OPPO)	32
4.5	Brand-Wise Blocked Percentage (POCO, Realme & Redmi)	33
4.6	Brand-Wise Blocked Percentage (Samsung, Sony & Xiaomi)	33
4.7	Brand-Version Web Request Percentage (Android v7, 8 & 9)	34
4.8	Brand-Version Web Request Percentage (Android v10, 11, 12 & 13) .	35
4.9	24 Hours Analysis	36
4.10	24 Hours Analysis of Meta (Company)	36
4.11	24 Hours Analysis of Facebook Android	37
4.12	24 Hours Analysis of Instagram	37
4.13	24 Hours Analysis of WhatsApp	38
4.14	24 Hours Analysis of Google (Company)	38
4.15	24 Hours Analysis of YouTube	39
4.16	All Web Request VS All Blocked Web Request	39
4.17	Blocked Web Request (First Party VS Third Party)	40
4.18	Third Party Comparison (Google & Meta)	41
4.19	Youtube Data usage in Google (First Party)	42
4.20	YouTube Ads on Google	42
4.21	YouTube Web Request VS Blocked Web Request	43
4.22	Google Web Request VS Blocked Web Request	44
4.23	Meta Blocked Web Request in Category	44
4.24	Meta First Party VS Third Party (Blocked Web Request)	45
4.25	Meta Web Request VS Blocked Web Request in Category	45
4.26	Meta First Party VS Third Party (Web Request)	45

List of Tables

2.1	Summary of Papers Regarding Mobile App Data Privacy	10
4.1	Android Version-Wise Blocked, Google & Meta Blocked Percentage .	30
4.2	Brand-Wise Blocked, Google & Meta Blocked Percentage	31

Chapter 1

Introduction

Data privacy is one of the most concerning topics today. After vast technological development, data privacy became a big topic for people because of their information privacy concerns. Data privacy studies how data will be collected, shared and used. Today without technology, our life is impossible, and the uses of this technology create vast amounts of data. This data contains personal, sensitive, social, and all kinds of data. So, using this data can earn bad and good results for us in terms of using it. Here comes the concept of data privacy and how this data will be collected, used, and shared. Leading advertising companies mainly collect data from their users with or without consent. They use this data to analyze the user's behavior to earn profit for the company. After the United States of America enacted the Fair Information Practice Act, the European Union enacted GDPR (General Data Protection Regulation) in 2016 for this reason [23]. Many other countries also established similar laws for concern with personal data privacy. However, people still question whether their data is safe and do they know which information is taken from them and for what purposes.

According to Statista[1], the most dominant smartphone operating system is Android, with over 2 billion users in 190 countries, holding 71.47%(August 2022). The number of smartphone users in China is expected to exceed 1.3 billion by 2026[6]. The Android App store has about 3.51 million applications[27]. In 2021, the Chinese third-party App Store had over 2.52 million Android applications[19]. Android is now used by the majority of smartphone and tablet makers. Every app store has many apps like games, social media, news, sports, and IM from many big & small companies. The app companies collect the user data after the app is installed on the user's phone, which is a big concern regarding data privacy from a user perspective. The app companies take user data sometimes with user permission for the app functionalities. However, the apps grab data silently from third-party libraries, and the companies sell the collected data to other third-party advertising companies. After installation, the app often wants permissions for user information like phone location, contact list, messages, call logs, and files & media access. Sometimes, apps do not need to use this data. If the user does not give permission, the app stops working, so the user is mostly forced to permit without questioning. It is an example of the invention of the data privacy policy. Also, As the Android OS is open-source, most phone companies make their customized OS. Those customized

OS generally get security patches lately compared to the stock OS; this creates a significant risk[32]. The customized OS breaks so many core functionalities because of their customizations, making vulnerability at the OS level. So in our research, we tried to study some of the android apps. If they take more than necessary information as they need and how they are gaining data, we also compare similar types of apps to see if the app can be run with minimal data.

For the research, we will use AdGuard along with The Black List Project and some infamous filter lists from EasyList, ABP, and Peter Lowe. Using AdGuard, we tracked the foreground & background tracking of the apps by using LocalVPN & HTTPS filtering. The filters from AdBlock Plus (ABP), EasyList & AdGuard helped to identify the ads, trackers & crypto miners from the ocean of logs. Moreover, The Black List Project & Netify determines the advertising companies

1.1 Research Problem

Android is the most used operating system for smartphones and tablets based on Linux, created by Android, Inc., and then acquired by Google [12]. With the increased availability of smartphone devices, the number of mobile applications available in different markets has increased tremendously. As of June 2022, about 2.65 million applications are accessible on Google Play alone [28]. An estimated 1.91 billion smartphone users around the globe use apps every day; these users take advantage of the apps' vast range of functionalities [14]. According to [5], on the Android platform, privacy control is critical. To guarantee safe information storage and delivery, traditional security guards only have one point, such as data encryption. This strategy may not work properly on the receiving end to prevent data or information leakage. Moreover, methods of social engineering that enable users to be duped without suspicion may endanger personal privacy. According to [16], most applications leak users' private information, yet consumers have no idea where or how the information is utilized. Many mobile applications use third-party libraries, implying that over 70% of apps would share users' private information with third parties. Each time a user installs an app on Android is presented with a list of permissions the app needs to function correctly. A permission is a unique string of text that Android or third-party developers may define [8].

According to [7], phone call lists, messages, call logs, browsing history, and GPS location are the sensitive user data that every mobile device stores. There is currently no appropriate means for customers to know how different apps access their private personal information or to block this access method. Although applications must request permission to access data (after installing the app), the customer is not notified of the nature of the access.

According to [23], consumers are worried about the processing, storing, and using of their personal information. In addition, they are concerned about mobile device vulnerabilities., which might result in data breaches, hacking, and data theft. All of these difficulties may hinder mobile application use among consumers. Because the mobile application market is one of the largest in the IT industry, a low adoption rate is detrimental to organizations. In 2022, market revenue totaled \$430.90 billion, and it is anticipated that this value will climb to \$614.40 billion by 2026 [29]. In

particular, the following questions will be answered: How much data is sent for what purposes is an app making? How many attempts and permissions are required for an app to remain functional, and how many attempts is one app making for just tracking and giving more precise/targeted ads? How many attempts are made by the big ad companies in the third-party apps? Furthermore, how are the attempts different based on user age, gender, location, Mobile Brand, and OS version? This research will answer the above question using AdGuard, Netify, The Black List Project, widely popular Adblock Plus (ABP), EasyList filters.

1.2 Research Objectives

This research focuses on developing & analyzing a detection system for intrusions for Android to minimize the tracking level and irrelevant data sharing of Android apps. Android OS is open-source and mostly customized by manufacturer phone companies. It creates many drawbacks, and a system-level blocking approach can help the users protect their valuable data. This can also affect the companies to respect the ad-targeting approach using users' information. The objectives of this research are:

1. To deeply understand how Android apps track users.
2. To deeply understand how the tracking techniques work.
3. To develop a model for detecting a universal tracking detection and tracking protection system to enhance user privacy.
4. To evaluate the model.
5. To compare the aggressiveness of big ad companies and their subsidiaries for tracking user's behavior and providing precise ads.

Very little work has been done regarding these related topics. This work mainly focuses on how the ad-tracking system differs in manufacturer brand, OS versions, etc. The related papers did not show that amount of priority in these criteria. That is why this paper aims to focus on these things that are responsible for ad-companies tracking behavior.

Chapter 2

Literature Review

The tracking behavior of Android apps is increasing rapidly. According to [26], in a study conducted by Gioacchino Tangari, 88.0% (n = 18472) of mHealth applications had code that may gather user information. 3.9 percent of applications (n = 616) communicated user data in their traffic. mHealth apps and other categorical apps also track & collect data in both foreground and background. According to [22], GDPR could not help stop ad companies from grabbing users' valuable information. It has just changed the tracking strategy of the companies to avoid legal issues.

2.1 Android App Permission Settings

According to [20], the Android system is currently a mobile terminal operating system with a market share of more than 80%; mobile apps running on Android face more security concerns owing to the open-source nature of its platform and the variety of its application market. To provide users better services, these apps typically access the user's mobile data, such as the permission to read the user's contact list, call logs, messages, geographical location, camera permission, storage & media permission, and sometimes even ask for modifying system setting permissions. Some of these permissions will solicit user authorization when the app is launched, while others will not; the default setting is permissive. In addition, if a user does not agree to grant access to a portion of the app, they cannot utilize any app feature. Thus, some users had to grant all rights to use the app. Currently, the issue of excessive app claims is pervasive. After a thorough examination of many categorized applications, it has been determined that privacy-related options may be located not only in the "privacy settings" part but also in the "Notifications," "General," and "About" sections. Users may only close them using the application permission management of the Android system settings if no closing option is provided.

The author's [11] analysis reveals that apps overextend their access to the personal access to personal information of mobile users since. Most consumers are unaware that their privacy is at risk through the phone and must therefore be secured. Given the proper knowledge, a high proportion of smartphone users will adjust the permissions originally provided to applications on their devices, according to the relevant poll.

The findings of this research [11] indicate the length of time mobile applications need data access. Specifically, most applications need a large amount of data stored on mobile devices, but just a handful run without any special access demand. The most popular kind of access is pictures, media, and file access, which enables users access to all data stored on their mobile devices. WiFi connection data is the next commonly desired kind of access, which might provide information about all wireless devices run by WiFi, geolocation, and other sensitive pieces of information for tracking. Another significant conclusion is that consumers evaluate mobile apps without considering the quantity of data entry required by each application.

In [8], the authors revealed how during the installation of an app, the user is given a limited amount of time and options to install the app; if they do not provide permission as requested, the installation of the app will be canceled. Therefore the user must provide permission for the application to be installed. It violates the user's right to privacy. As Android gives a Third Party app with an enhanced API, the app may get any data without the user's knowledge. There is no dynamic method for modifying the permission settings of apps. Thus we must create a mechanism that gives users this capability. Static permission management is the foundation of the Android permission system.

2.2 Existing Strategies for Addressing Tracking Risks

2.2.1 End user privacy controls

According to [15], even though online browsers have traditionally allowed users to block tracking using the browser's default settings or Third Party plugins, tracking occurs in every sector whether it is web or mobile applications. In contrast, no major smartphone platform operating system allows users to prevent or otherwise regulate Third Party app surveillance.

2.2.2 OS Regulations

Due to the introduction and proliferation of trackers and the absence of widespread implementation of effective end-user tracker restrictions, platform developers for Mobile OS have taken a variety of safeguards to mitigate the dangers.

2.2.3 Legal Regulation

Self-regulatory efforts coexist with many country-specific legal prohibitions with vast degrees of implementation. Europe's data protection laws are perhaps the most strict and forward-looking of all of them. According to [22], they analyzed the Android application ecosystem, which continues to be the leading ecosystem for smartphone apps. To examine how the tracking environment has altered with the adoption of the GDPR in 2018, they analyze roughly 2 million Android applications from the app market of the UK before and after its implementation in 2018. They obtained the data set from a thesis work on app tracking published prior to the implementation of GDPR. This allowed them to get app tracking data prior to

GDPR. They evaluate the app in two ways. Dynamic method versus static method. The dynamic analysis examines the behavior of applications at runtime by running them on a real smartphone OS. The majority of work in this area focuses on evaluating network traffic generated by applications. Here, an app from the Google Play Store UK was used. Without running them, static analysis dissects the behavior of applications. To examine tracking in the app using static analysis, they used four ways: app detection and download, detection of tracking, resolution of companies, and market concentration analysis. They stole an app from the UK Google Play Store. Then, for tracking detection, an automatic scan of *.dex files (matching to developed application code) was performed to classify all URLs (strings beginning with http:// or https://). Then, all URLs relating to hosts that appeared in at least 0.1% of applications (in 2017 or 2020) were cross-referenced to confirm that they corresponded to trackers. The prior study’s definition of a tracker was used. Results indicate that the root tracker corporation is still monitoring apps via a subsidiary company, and the number of apps containing tracker code has risen.

2.3 Risks of Using Android Third Party Libraries

Third-party libraries are heavily used in Android applications. These libraries share rights with their hosting applications, which are easily authorized and may expose users’ private information without their concern. They[16] examined 150 popular apps and gathered 1,909 privacy-related call chains. To produce Android-device identification, Third Party libraries most of the time need access to device data, according to privacy regulations. In addition, Third Party libraries will need the location data and network connection, which may lead to unanticipated concerns of privacy leakage. In addition, it has been discovered that when an app is operating, the hosting application and Third Party libraries run in the same process, share the same rights, and have indistinct borders. Several studies have looked into Android advertising libraries using static analysis. The studies reveal that many in-app advertising libraries collect personal information without declaring it in their documentation [16], and that this practice may be on the rise [22]. The author [3] presented an automated method for locating and repositioning missing permission questions in areas where Third Party libraries may abuse permissions. Only a few studies have used dynamic analysis to uncover potential risks [4]. Brahmastra[4] is an automated tool for examining the possible vulnerability of Third Party libraries integrated with smartphone applications, which goes above the barrier of GUI-based testing tools. MAdFraud [2] detects fraudulent ad clicks from host programs using dynamic analysis. FlowDroid and TrustDroid were used for static privacy leakage detection, but TaintDroid and NDroid are dynamic tools that need modification to the Android system. AppFence is a TaintDroid app that can prevent unwanted data transfer. AppIntent is a set of dynamic and static analytic tools for determining whether personal data collection aligns with the user’s intent. The framework also includes the Third Party library detection tool LibD [10], and it uses an Android application’s internal code dependencies to find and categorize library candidates. Based on feature hashing, the tool performs better than most earlier methods, which categorized library candidates based on similarity comparison in that it can manage code with obscured package and method names. [8] Also suggested, The LBE Privacy Guard is a background service that keeps track of how applications behave.

Suppose an app tries to access sensitive information like their location, cell phone number, and the Internet without permission. In that case, they are notified and given the option to decline or authorize the request. The PMP (Privacy Management Platform) educates the user about the various Privacy Settings accessible and how they impact the services. So that individuals can easily make educated decisions. They described an algorithm based on these apps and frameworks in which the system would distribute information to apps based on user authorization. The authors [8] proposed system's access manager would stop an app API from attempting to access data without user consent. The main goal was to develop a system that offers the user control over the data privacy of his Android phone.

They investigate the Third Party libraries' privacy leakage characteristics within Android applications using a case study [16] and establish four types of data leakage routes for Android apps. The authors then provide a privacy-preserving analysis methodology that allows for fine-grained and real-time analysis. They differentiate between the hosting app and the app's Third Party libraries. The Xposed architecture is used to construct their instrument. Xposed is a framework for Third Party plugin development. The Android privacy-related APIs can be hooked using the Xposed framework. The proposed technology can address the question of which application component obtains and leaks sensitive data.

2.4 Private Data Leakage in Mobile Devices

The authors [17] provide a framework for analyzing smartphone use and mobile network traffic data to do extensive privacy leakage detection and privacy inference mining on vast amounts of real-world data. The authors [17] use mobile traffic data to create a training data set and train a privacy detection algorithm. Furthermore, they identified private usage patterns using machine learning algorithms. They discovered that certain apps broadcast passwords in plain text, which leads to more privacy categories leaking in Android than in iOS; that GPS location is the most exposed privacy in both systems [7]; and that usage patterns are linked to the price of the mobile device. Network communications must be converted to text to make processing easier. They extracted the HTTP GET / POST data from such packets, converted it to JSON using Bro and Scrapy, and noticed data leaks through PII mining and password use. Our data produced by incomplete applications and browsers include a significant proportion of passwords transferred in plain text. After deduplication, 221 plain text credentials were removed because they were insufficiently robust. They also discovered that data escapes via users' location information. It may occur in two ways: the gathering of location information and clustering of location information.

2.5 Various Encryption for Enhancing Securities

People use mobile phones (especially Android) for various purposes, including personal ones. Consequently, users transmit text messages and other confidential and private information via mobile phones. However, by doing so, they compromise their privacy, as most applications read private user data and send it to Third Party libraries for analysis. Therefore, to deal with this issue, the authors [12] created

an app that allows users to exchange encrypted text or other information (photos, addresses) and choose the algorithm for encryption from four options: AES, Triple DES, RSA, and Blowfish. They also maintained the Google-implemented security system, which included Android application manifest, Sandboxing, and Google bouncer.

2.6 Privacy-Preserving Apps

Although consuming permissions aggressively might be done to provide greater functionality, this feature may be viewed as a privacy issue because other functionally equivalent applications do not require this permission [13]. This relates to the app's aggressive permission consumption. If less than ten percent of functionally comparable applications in a group utilize certain permission, it is considered aggressive. For this reason, they refer to an aggressively utilized permission as unnecessary permission for that program. They selected 1400 applications that used unnecessary permissions at random. In addition, static analysis was performed on the .apk files obtained from the Google Play Store. There were a total of 28,000 participants in the study. For static analysis, they look for APIs used by these applications. Every app has an embedded library that calls an API function. These Android API functions are protected by system authorization. Some app libraries demand additional permissions, although other applications of the same sort do not. They determined that improved apps with comparable features might replace up to 43.5 percent of apps using real-world data from over 28,000 users.

2.7 Exposing the Data Sharing Practices

As mentioned in [14], Examining Data Controller Indicators in a real situation of privacy-related decision-making was the main goal of this study. They wanted to see if DCIs would induce people to make different decisions than they would otherwise. Second, they wanted to know if their DCI decisions would be based on unique lines of reasoning or if they would consider other factors. Finally, they want to know if users prefer DCIs and which forms they find most useful. To access the Internet, a smartphone was set up with proxy software and a mitmproxy server. After that, the app was downloaded onto the phone, and a 10-minute tour was finished to use every function at least once. This method was created to ensure that the app generated and collected a representative sample of traffic statistics. The raw log files were translated into higher-level descriptions using a dual data processing pipeline. Initially, raw log files were processed by data detector to determine the types of data being sent. Phone ID, Phone attributes, Location, and Personal factors were used to accomplish this. The primary research involved a series of app selection tasks where participants had to choose between two functionally equivalent apps. Participants were encouraged to choose one of the two programs using a think-aloud approach based solely on the information supplied in each interface. After making a decision, participants were asked to describe their choice and rate their confidence in it on a Likert scale. This study has a total of 32 participants. Unsurprisingly, a higher diversity of information offered by interface conditions leads to a broader range of decision-making processes. However, the elements evaluated in the control

and information-rich conditions differed significantly; in the information-rich situations, the parameters that were most often reviewed in the control conditions (such as app name and data categories) were examined much less frequently. Other elements were initially evaluated, such as the number of data destinations and purposes of use, as well as characteristics of the destination firms, such as their reputation, dependability, and origin. The PDCI was particularly interested in whether an app provided data to businesses that already had it or increased their exposure (Personalized Data Controller Indicator). These findings show that these characteristics were more influential in app selection than in platform authorization interfaces.

2.8 Related Works

This part aims to review previous relevant work in Android Apps Data Privacy critically.

While Third Party libraries give a variety of features, they also raise security and privacy issues. The host applications and Third Party libraries run in the same process and have the same permissions. The ability of Third Party libraries to adhere to privacy rules is beyond the control of application developers. According to [16], many apps with ad libraries gather private information without notifying the use of privacy permissions in their privacy policy statement, and this tendency is increasing. Livshits et al. suggested a technique for automatically finding and putting missing permission prompts in areas where Third Party libraries may potentially misuse rights.

To find and categorize library possibilities, the study integrated LibD, a Third Party library discovery tool, with the internal code dependencies of an Android app. The tool is based on feature hashing, unlike most earlier approaches that categorize discovered library candidates based on similarity comparisons. It can better handle code with obfuscated package and method names.

In [13], of the 1,400 applications in our sample, 358 (25.6 percent) utilized one or more unnecessary permissions that embedded libraries might freely abuse. 72 percent of these 358 applications had libraries that could access one permission, whereas 28 percent contained libraries that could access two or more rights. The ability of a library to access numerous superfluous permissions reduced monotonically as the number of permissions increased.

Table 2.1: Summary of Papers Regarding Mobile App Data Privacy

Ref	Task	Algorithm/Tool used	Data set size	Accuracy Obtained
[5]	On Android apps, information loss evaluation, privacy leak detecting, and privacy risk assessment	An analysis framework called AppLeak	NA	NA

Continuation of Table 4.2				
Ref	Task	Algorithm/Tool used	Data set size	Accuracy Obtained
[7]	Access to Sensitive Information in Android Apps is being monitored	Modified the Android OS, Hidden Markov Models(HMMs)	single Android device. Four mobile apps, MDM	NA
[8]	Use a new type of algorithm in which access manager block app API if app try to access any info without permission	Suggested new algorithm with 6 steps	NA	NA
[11]	The excessive permissions used by free applications	NA	529	NA
[12]	Transmite data in encrypted forms in android phones	AES, Triple DES, RSA, Blowfish	NA	NA
[13]	Analyzing risk from aggressive permission usage from apps	Static analysis on app's embedded libraries	1400 apps were used and a total of 28000 user	NA
[14]	Expose the data sharing activities of apps	mitmproxy server, DCI(Data Controller Indicator),PDCI(Personalized Data Controller Indicator)	Lab Environment; Total 32 participant,50 apps	Not a real time survey
[15]	Finding the distribution of Third Party trackers	APKTool, python library tldextract	959,000 applications from the Google Play Stores in the USA and the UK	Gini inequality coefficient= 0.44
[16]	Investigating Third Party libraries' privacy leaking behavior inside mobile applications	The Xposed framework combining static and dynamic methods, LibD	1909 call chains connected to privacy information,150 applications using a Third Party library	NA

Continuation of Table 4.2				
Ref	Task	Algorithm/Tool used	Data set size	Accuracy Obtained
[17]	Analyzing how and in what scale private data leaks from mobile phones	Preprocess data (using Wire-shark, Bro), Webview, Recon approach, DSSCAN	480 types of android phones based on price range, analysis on 221 users, based on the passwords they use	NA
[20]	Investigates the users' attention to personal privacy data, then analyzes users' understanding	Survey	A total of 42 categories of Mobile applications from the top 50 apps in the app list, 20 apps were selected	NA
[21]	Examining the privacy practices of well-known COVID-19 and social media applications	Self-Developed Automation Tool	46	NA
[22]	App download then track detection and company market concentration analysis	Scan *.dex file searching http string, open source software gplaycli	1,000,750 apps from UK play store	NA
[23]	Made 9 hypothesis based on data collecting behavior then took percentage value of these 9 behaviors from many incident	2 (FIPs versus NO FIPs) X 2 (AUTO vs non-AUTO) between subject factorial design	NA	dULS & dG < 95%, SRMR = 0.021 (< 0.08), NFI = 0.956 (> 0.90), dULS = 0.046 (HI95 % = 0.052), and dG = 0.256 (HI95 % = 0.687)

Continuation of Table 4.2				
Ref	Task	Algorithm/Tool used	Data set size	Accuracy Obtained
[26]	Analysing Inconsistent privacy practices in mHealth apps	Crawler	20991	97%

Chapter 3

Methodology

The AdGuard app provides a depth log of the web request along with other details such as the unique device id. For the research, the AdGuard log files is customized with 10 filters from 10 different companies, which are "EasyList", "EasyPrivacy", "Peter Lowe's Blocklist", "Adblock Warning Removal List", "NoCoin Filter List", "AdGuard Mobile Ads Filter", "AdGuard Annoyances Filter", "AdGuard Base Filter", "AdGuard Tracking Protection Filter", "AdGuard Social Media filter". These filters are responsible for different classifications, such as ads, trackers, malware, and crypto miners. Figure 3.1 shows the high-level view of the processes.

The research consists of three major stages:

1. Input Data Pre-Processing: this stage is concerned with formatting the raw data in a way so that it can be processed.
2. Processing: this stage is concerned with de-cluttering the data and making it categorical so that it can be analyzed.
3. Predictions: this stage is concerned with using ML and some other mechanisms to detect the tracking patterns of the ad companies.

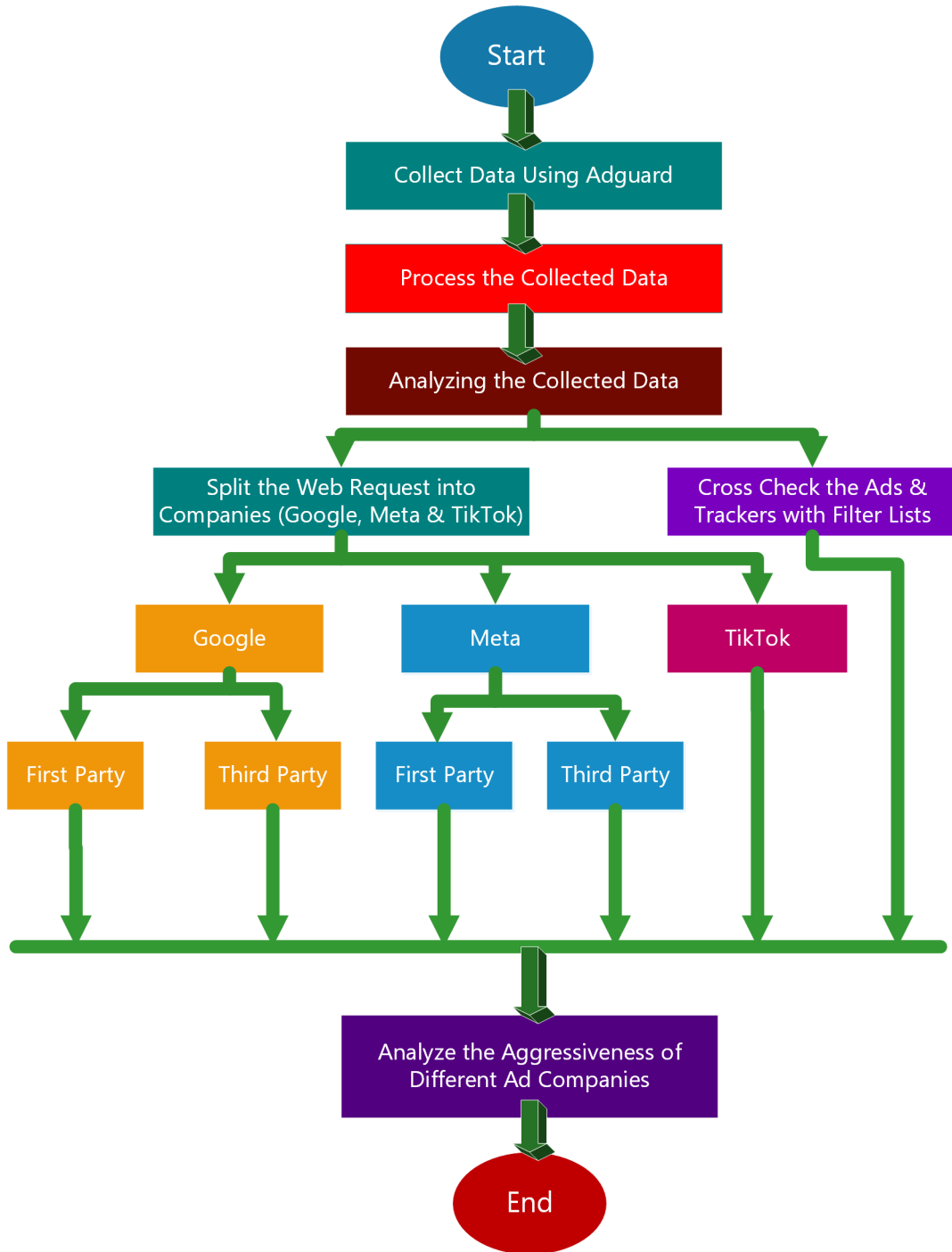


Figure 3.1: The Flow Chart of the Proposed Generalized Process

3.1 Pre-Processing Data

The extracted AdGuard log file consists of raw data, including device model, android version, web request time, and web request (ads, trackers, standard request, device system request, blocked web request, and other system information). Each person provided 72 hours of data for this phase, and this data was collected from all over the Bangladesh. To avoid data loss, each person provided several log files after 12 or 24 hours. To simplify processing and ensure proper timing and no overlap, those log files were merged.

3.2 Processing Data

After merging, several python scripts were used to de-clutter the merged file. Firstly, python scripts were used to filter out the unwanted portion, including AdGuard settings, AdGuard startup logs, selected filter lists, and other unnecessary data. Then, from the filtered requests, blocked requests were identified by another script. Cross-checked were made several times, with the filter lists to avoid mistakes. From 72 hours, convenient 24 hours data were taken for the analysis.

Later, the block requests were categorized into four categories: Google, Meta, TikTok(ByteDance), and Others. While Google & Meta dominate the ad markets, TikTok grabbed a large portion within a concise period with its only one app. The whole scenario is shown briefly in Figure 3.2 which includes Total web requests; Total blocked web requests, Total web requests, and Total blocked web requests from Google, Meta, TikTok(ByteDance), and others. Then, Figure 3.3 to Figure 3.8 shows the analysis of how the ad and tracking request varies in different versions of Android. The 24 hours data were divided into hourly, to simplify the analysis and make the analyzing process more efficient (Figure 3.9). Figure 3.10 to Figure 3.23 shows a brief overview of the total and blocked requests between different manufacturers/brands. From the data set, fourteen brands: Honor, HUAWEI, Infinix, Lenovo, Nokia, Sony, OnePlus, OPPO, POCO, Realme, Redmi, Samsung, Nothing and Xiaomi were found. Lastly, some other independent projects like The Block List Project, Netify presets are used to detect company-wise ad server/domain & IP address detection. Which helped to determine the aggressiveness of the ad-companies tracking behavior and also showed how the mobile manufacturing companies are responsible for ads and tracking.

We have used several python scripts to process our data. Here are some basic structure of python scripts that were used during the data processing time.

Listing 3.1 is used to identify web requests.

```
1 #This part is for analyzing total webrequest which includes blocked
   webrequest
2 count = 0
3 with open(filename) as fname:
4     for line in fname.readlines():
5         if 'proxy-server-pool' and 'DEBUG' in line:
6             count += 1
7 print('Total Webrequest:', count)
```

Listing 3.1: Web Request

Listing 3.2 is used to identify browser web requests and non-browser web requests.

```
1 #This part is for analyzing total webrequest which includes blocked
  webrequest and differentiation between browser and non-browser
  webrequest
2 countWOBrowser = 0
3 countBrowser = 0
4 with open(filename, 'r') as fname:
5     filecontent=fname.readlines()
6     with open('BrowserList.txt') as fname2:
7         filecontent2=fname2.readlines()
8         for line in filecontent:
9             count = 0
10            for line2 in filecontent2:
11                if not re.findall(line2.strip(), line, flags=re.IGNORECASE)
:
12                count += 1
13            if count == 127 :
14                countWOBrowser += 1
15        for line in filecontent:
16            for line2 in filecontent2:
17                if re.findall(line2.strip(), line, flags=re.IGNORECASE):
18                    countBrowser += 1
19                break
20 print('Total without Browser:', countWOBrowser)
21 print('Total Just Browser:', countBrowser)
```

Listing 3.2: Browser and Non-Browser Web Request

Listing 3.3 is used to identify browser web requests from Google developed apps.

```
1 count = 0
2 with open(filename) as fname:
3     with open(new_filename, 'w') as writefile:
4         with open(new_filename2, 'w') as writefile2:
5             for line in fname.readlines():
6                 if 'com.google' in line:
7                     count += 1
8                     continue
9                 if 'android.chrome' in line:
10                    count += 1
11                    continue
12                if 'app.revanced.android' in line:
13                    count += 1
14                else:
15 print('Total Webrequest of Google (1st Party):', count)
```

Listing 3.3: Google (1st Party) Web Request

Listing 3.4 is used to identify browser web requests from Meta developed apps.

```
1 #This part is for analyzing total webrequest of Meta (1st Party) which
  includes blocked webrequest from non-browser webrequest
2 count = 0
3 with open(filename) as fname:
4     for line in fname.readlines():
5         if 'com.facebook' in line:
6             count += 1
7             continue
8         if 'com.instagram' in line:
```

```

9         count += 1
10        continue
11        if 'com.whatsapp' in line:
12            count += 1
13        else:
14    print('Total Webrequest of Meta (1st Party):', count)

```

Listing 3.4: Meta (1st Party) Web Request

Listing 3.5 is used to identify Meta's Graph web requests.

```

1 #This part is for analyzing total webrequest of Meta's Graph which
  includes blocked webrequest from non-browser webrequest
2 count=0
3 with open(filename) as fname:
4     filecontent=fname.readlines()
5     text="graph.facebook.com"
6     for line in filecontent:
7         x = re.findall(' "[^"]*" ', line)
8         if re.findall(text, *x):
9             count += 1
10    print('Total Webrequest of Meta Graph:', count)

```

Listing 3.5: Meta Graph

Listing 3.6 is used to identify ads.

```

1 #Ads Count
2 count = 0
3 with open('/content/ads.txt') as fname:
4     filecontent=fname.readlines()
5     with open(filename) as fname2:
6         filecontent2=fname2.readlines()
7         for line in filecontent:
8             for line2 in filecontent2:
9                 x = re.search("rule=(.+?) ", line2)
10                y = (x.group(1) + "\n")
11                if y in line:
12                    count += 1
13    print(count)

```

Listing 3.6: Ads

Web Request & Blocked Request of All Ad Companies

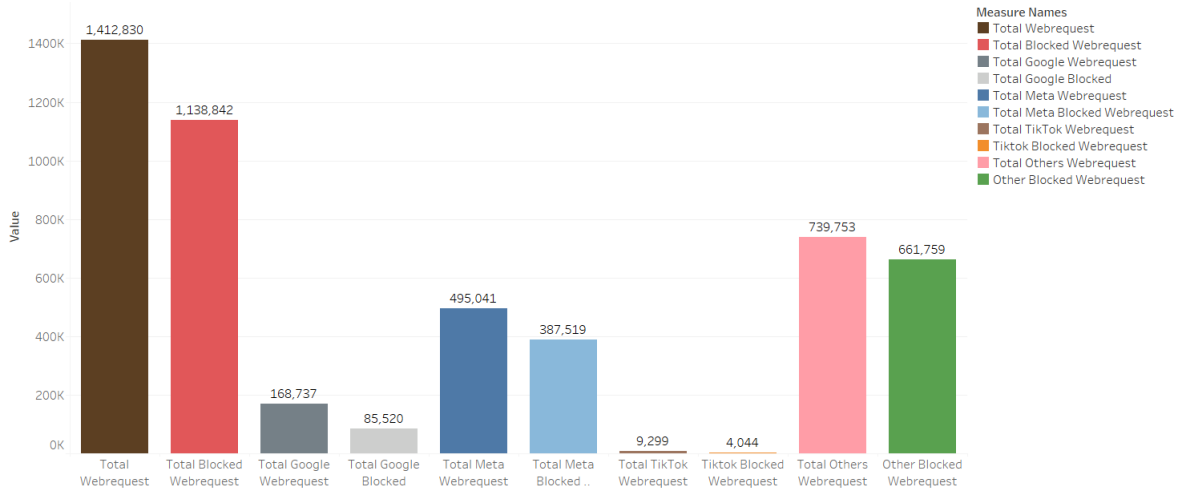


Figure 3.2: Total Web Request vs Total Blocked Web Request for Different Ad Companies

Web Request and Blocked Request (Version 7)

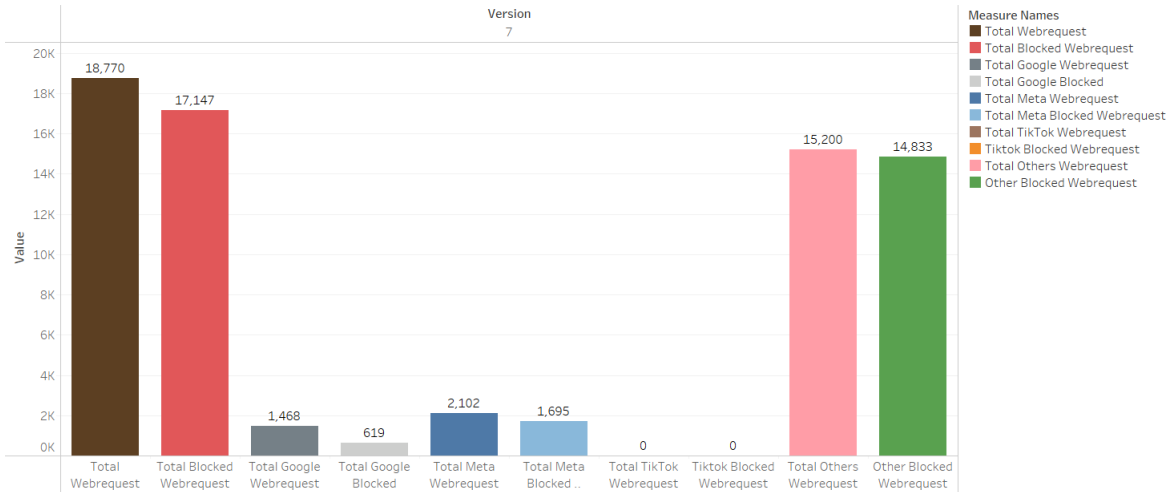


Figure 3.3: Comparison of Total Web Request and Total Blocked Web Request in Android 7

Web Request and Blocked Request (Version 8)

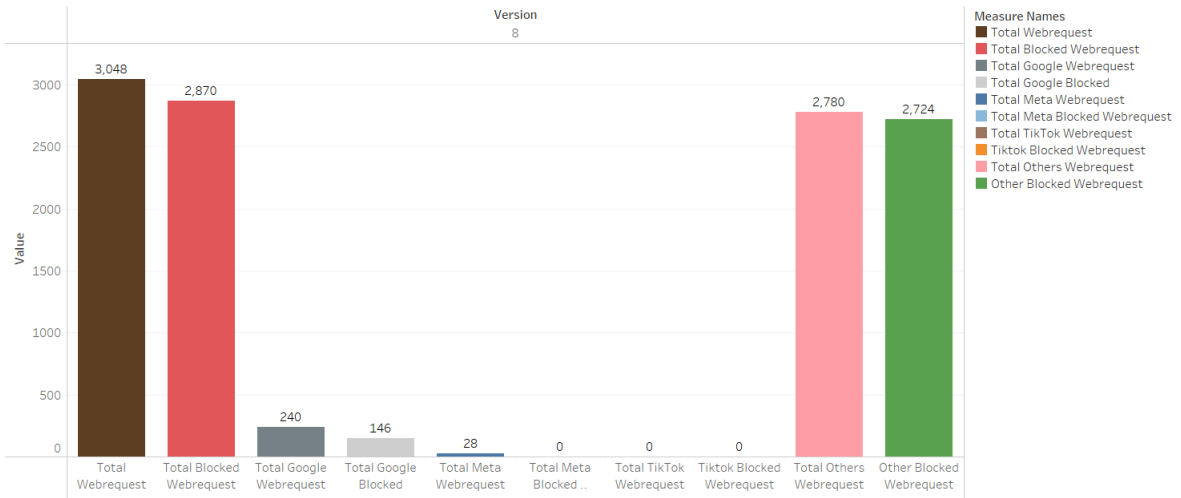


Figure 3.4: Comparison of Total Web Request and Total Blocked Web Request in Android 8

Web Request and Blocked Request (Version 9)

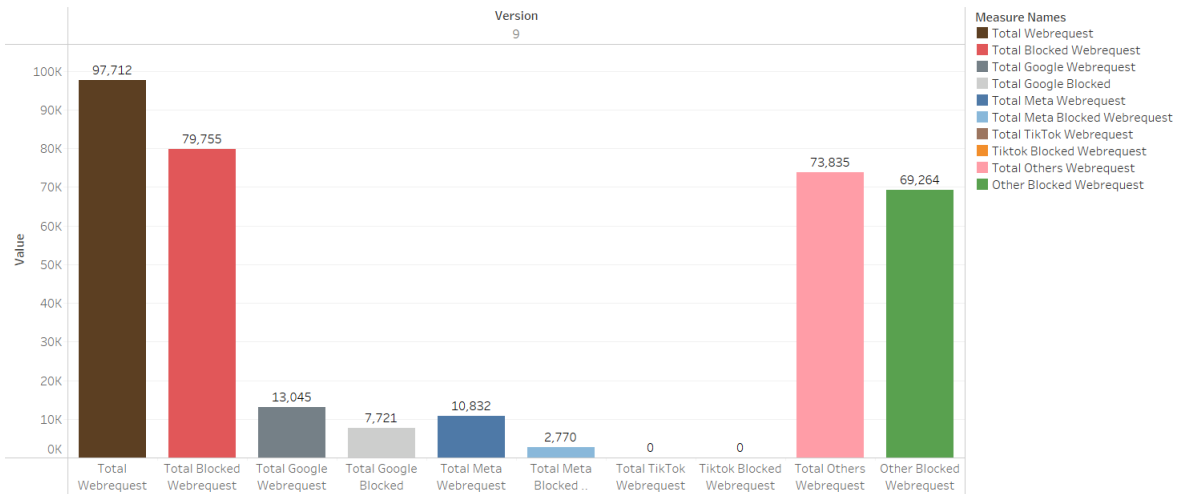


Figure 3.5: Comparison of Total Web Request and Total Blocked Web Request in Android 9

Web Request and Blocked Request (Version 10)

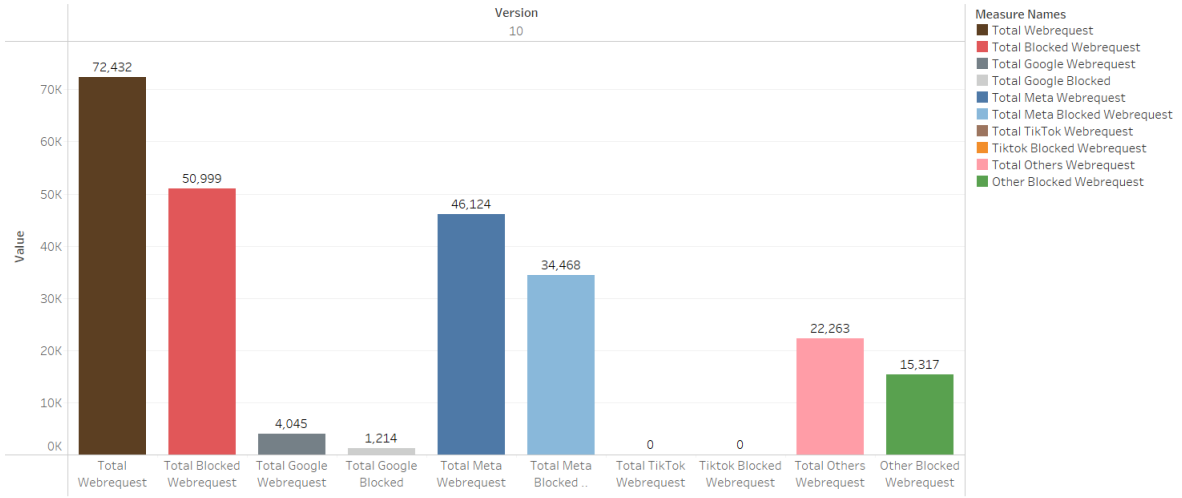


Figure 3.6: Comparison of Total Web Request and Total Blocked Web Request in Android 10

Web Request and Blocked Request (Version 11)

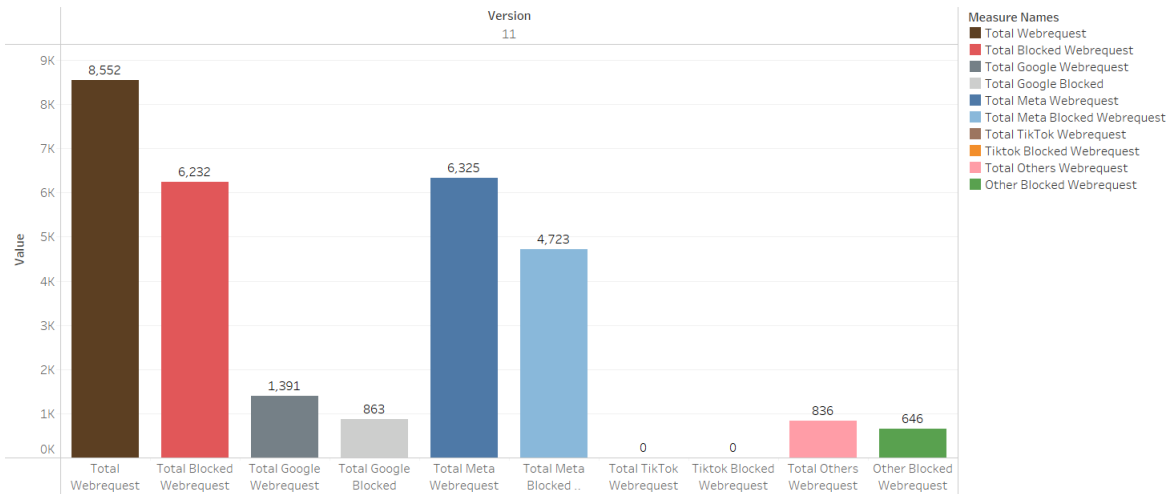


Figure 3.7: Comparison of Total Web Request and Total Blocked Web Request in Android 11

Web Request and Blocked Request (Version 12)

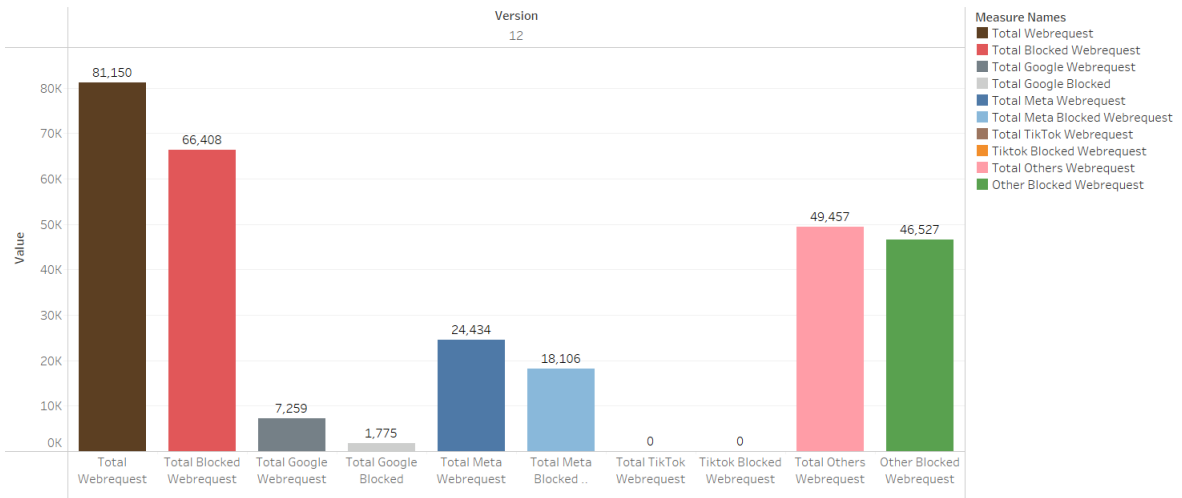


Figure 3.8: Comparison of Total Web Request and Total Blocked Web Request in Android 12

Versions VS Blocked Webrequest

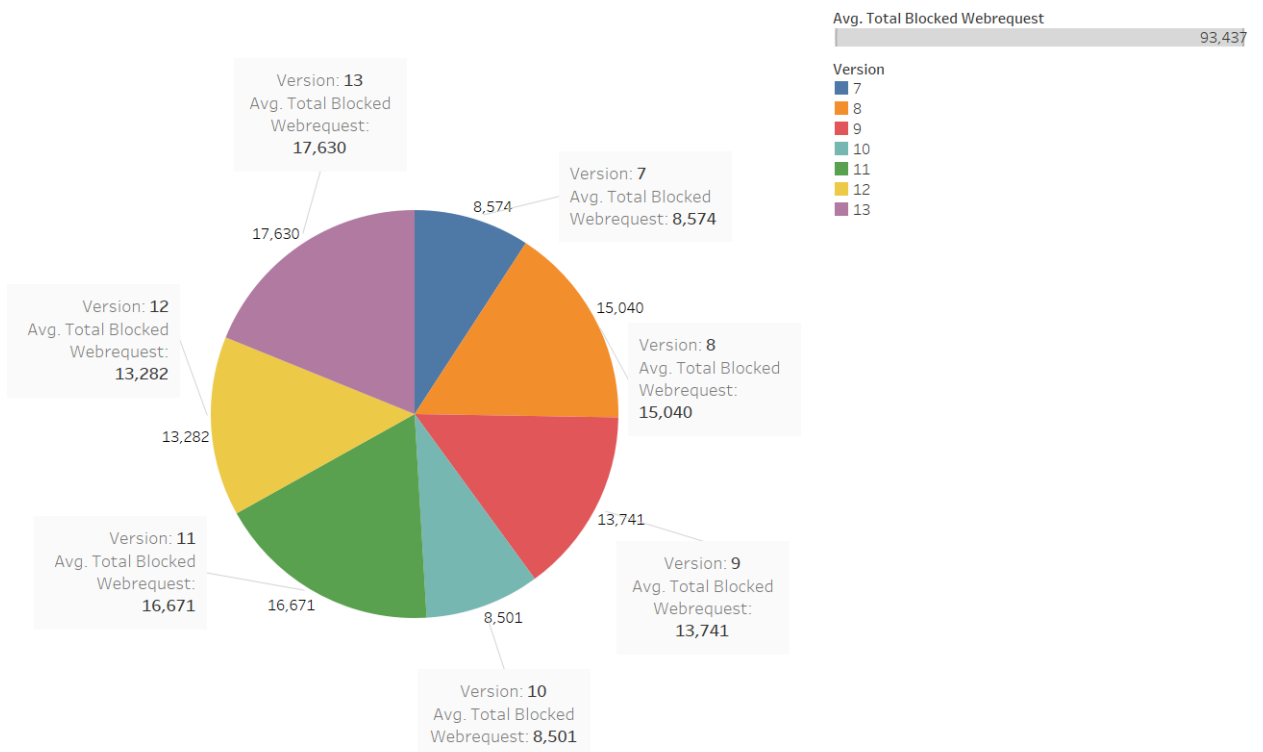


Figure 3.9: Version VS Blocked Web Request

Web Request and Blocked Request(HONOR)

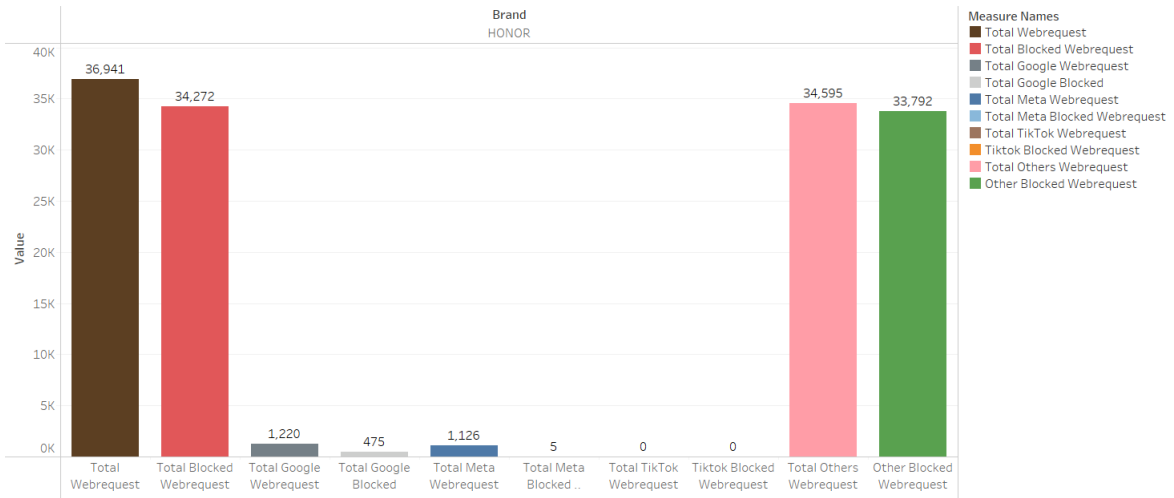


Figure 3.10: 24 hours Total Web Request and Blocked Web Request Comparison in Honor

Web Request and Blocked Request(HUAWEI)

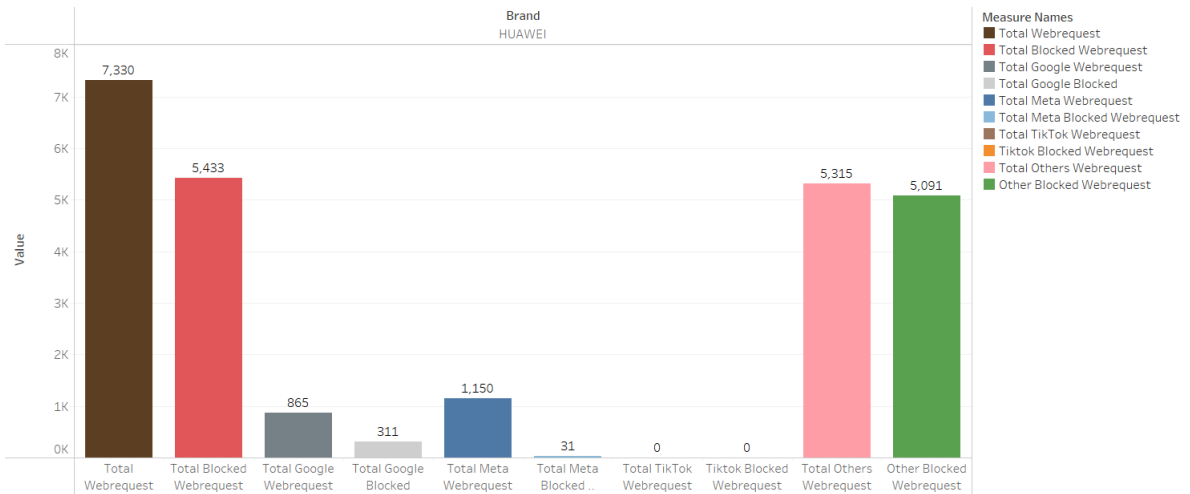


Figure 3.11: 24 hours Total Web Request and Blocked Web Request Comparison in HUAWEI

Web Request and Blocked Request(Infinix)

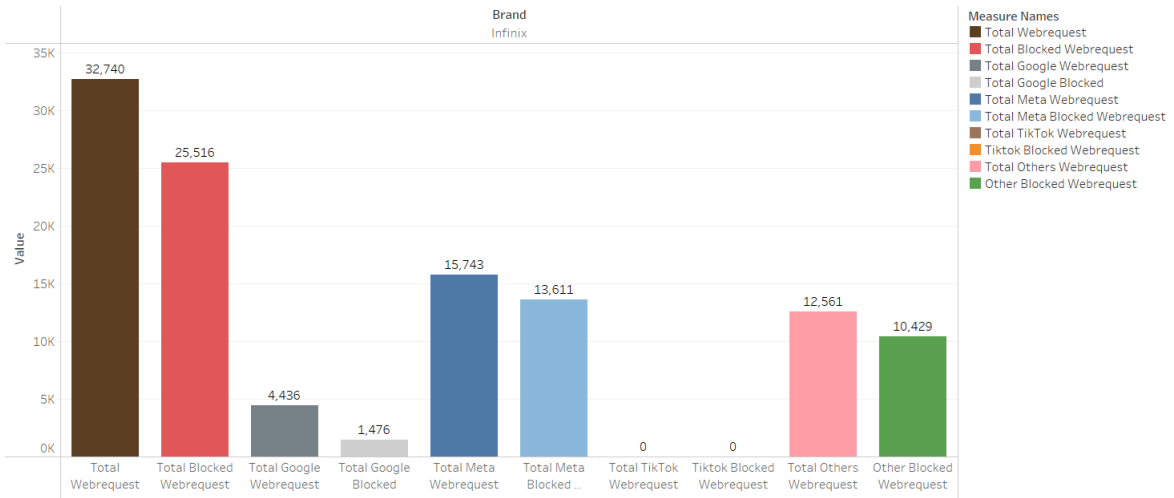


Figure 3.12: 24 hours Total Web Request and Blocked Web Request Comparison in Infinix

Web Request and Blocked Request(Lenovo)

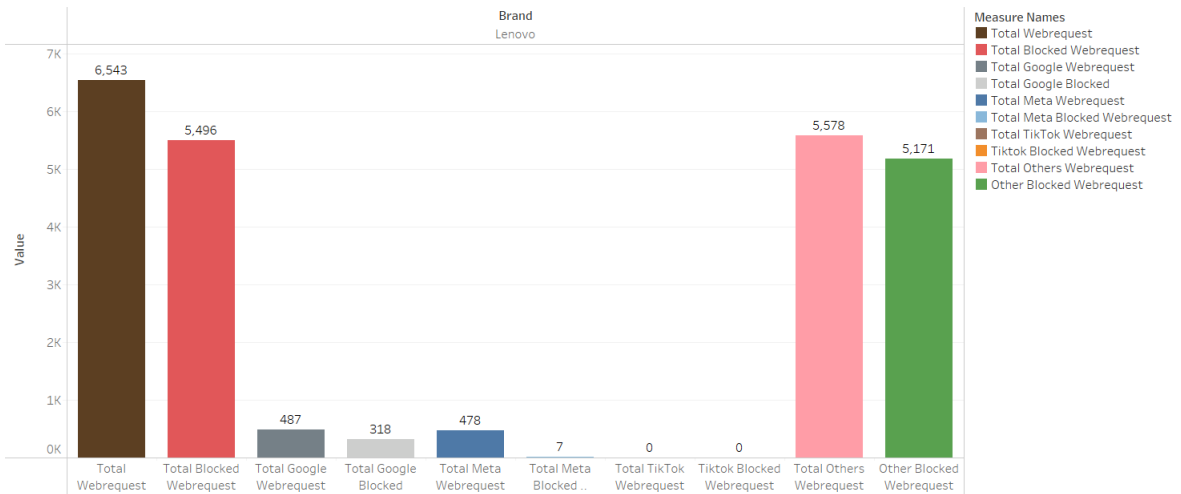


Figure 3.13: 24 hours Total Web Request and Blocked Web Request Comparison in Lenovo

Web Request and Blocked Request (Nokia)

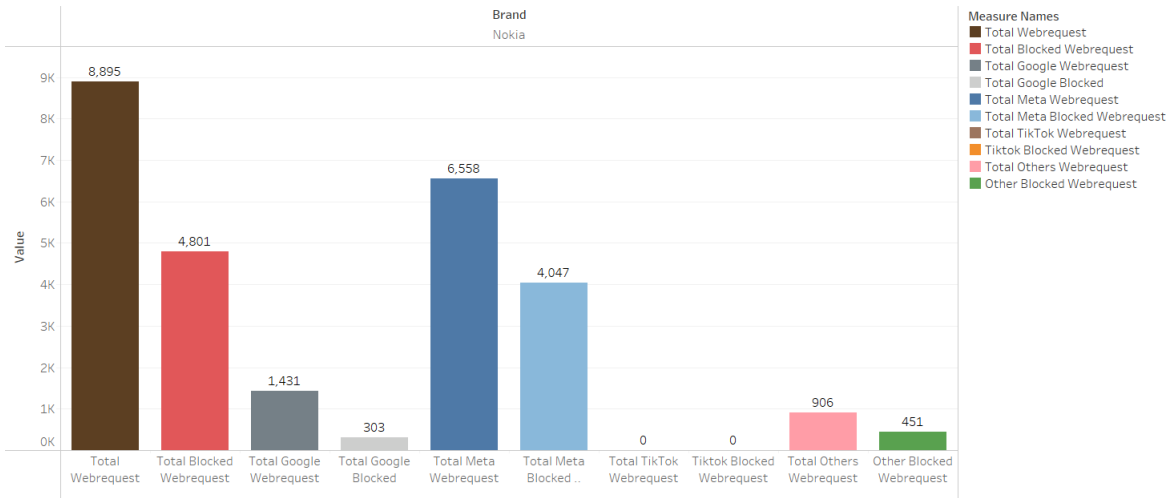


Figure 3.14: 24 hours Total Web Request and Blocked Web Request Comparison in Nokia

Web Request and Blocked Request(Nothing)

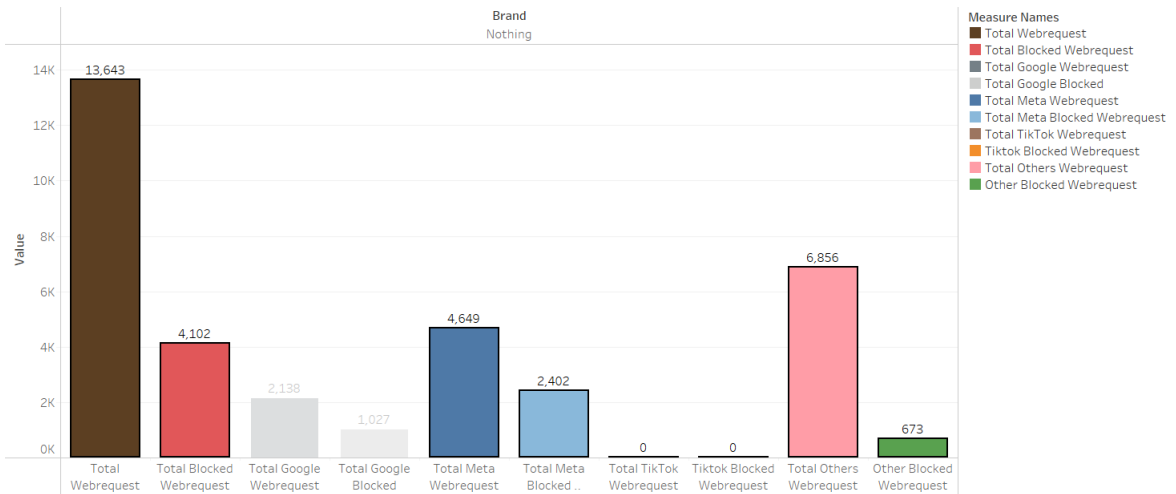


Figure 3.15: 24 hours Total Web Request and Blocked Web Request Comparison in Nothing

Web Request and Blocked Request(OnePlus)

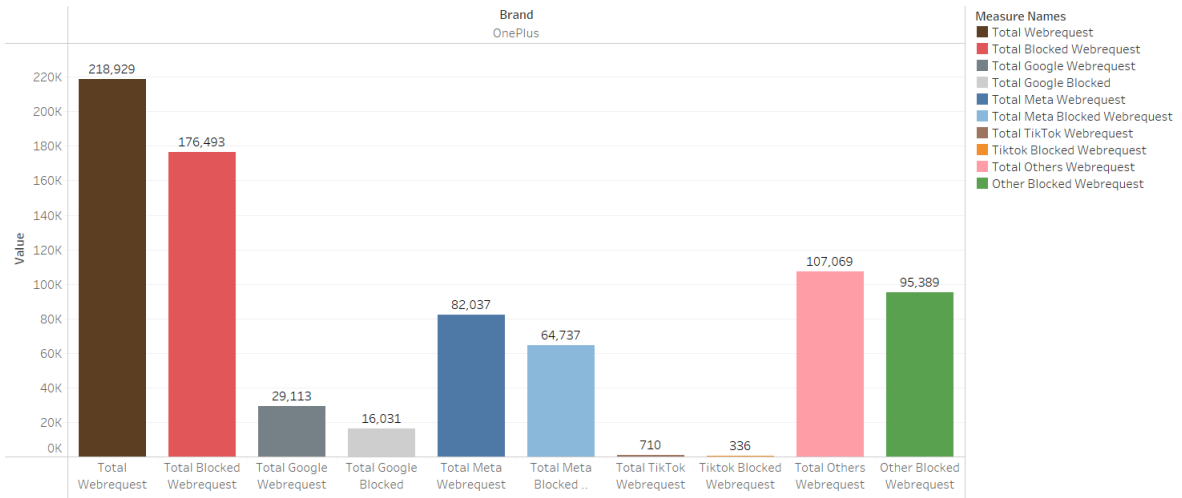


Figure 3.16: 24 hours Total Web Request and Blocked Web Request Comparison in OnePlus

Web Request and Blocked Request(OPPO)

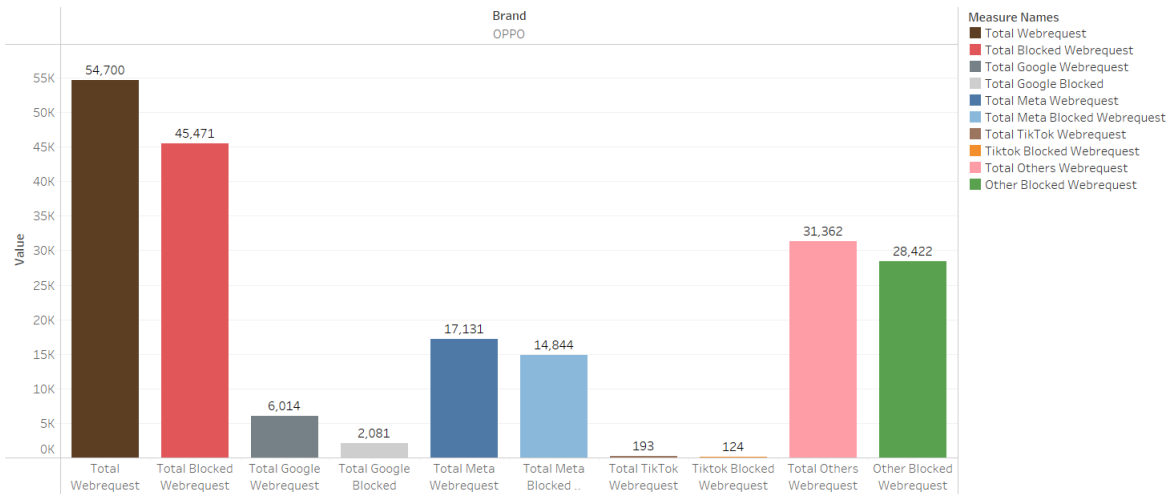


Figure 3.17: 24 hours Total Web Request and Blocked Web Request Comparison in OPPO

Web Request and Blocked Request(POCO)

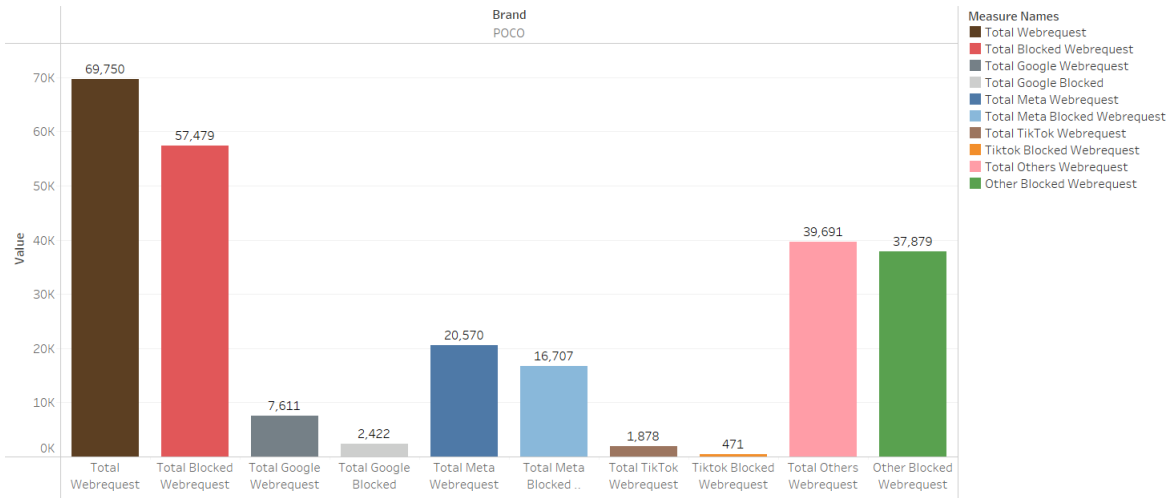


Figure 3.18: 24 hours Total Web Request and Blocked Web Request Comparison in POCO

Web Request and Blocked Request(Realme)

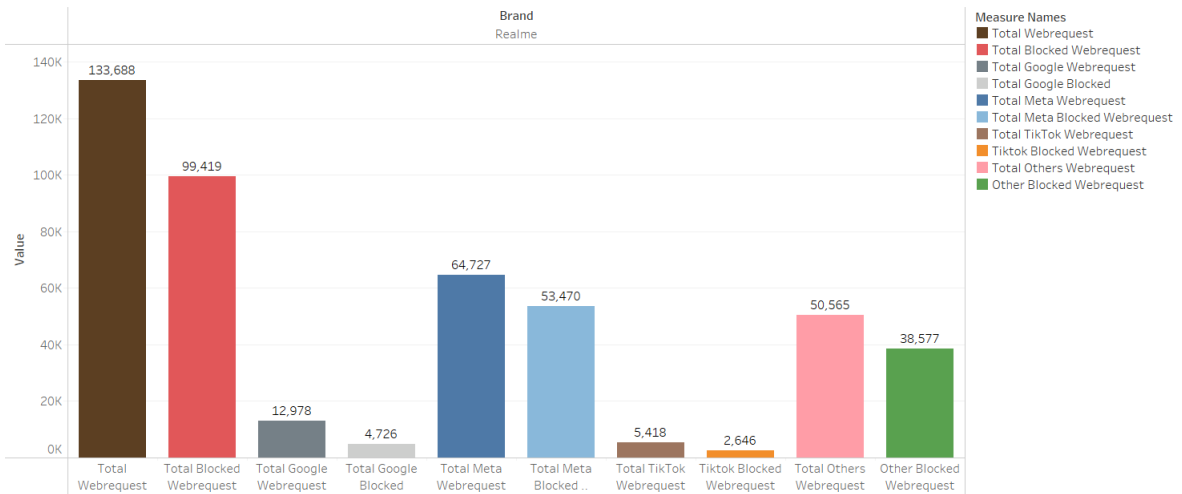


Figure 3.19: 24 hours Total Web Request and Blocked Web Request Comparison in Realme

Web Request and Blocked Request(Redmi)

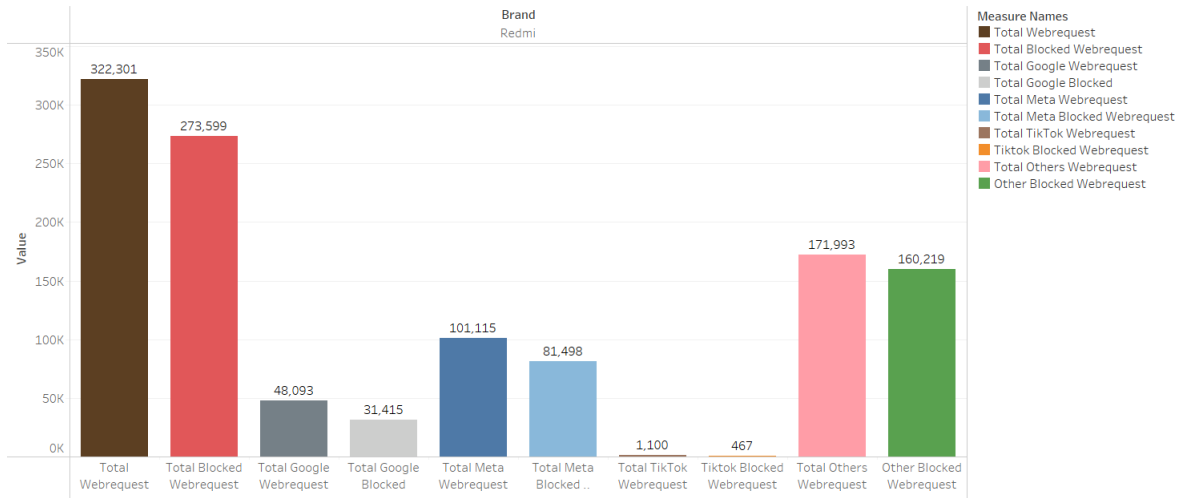


Figure 3.20: 24 hours Total Web Request and Blocked Web Request Comparison in Redmi

Web Request and Blocked Request(Samsung)

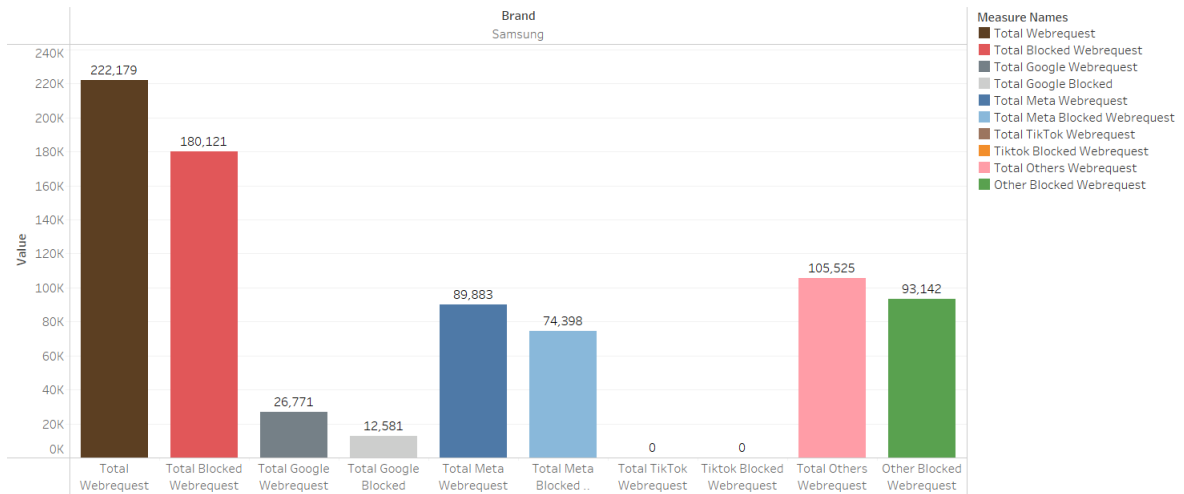


Figure 3.21: 24 hours Total Web Request and Blocked Web Request Comparison in Samsung

Web Request and Blocked Request(Sony)

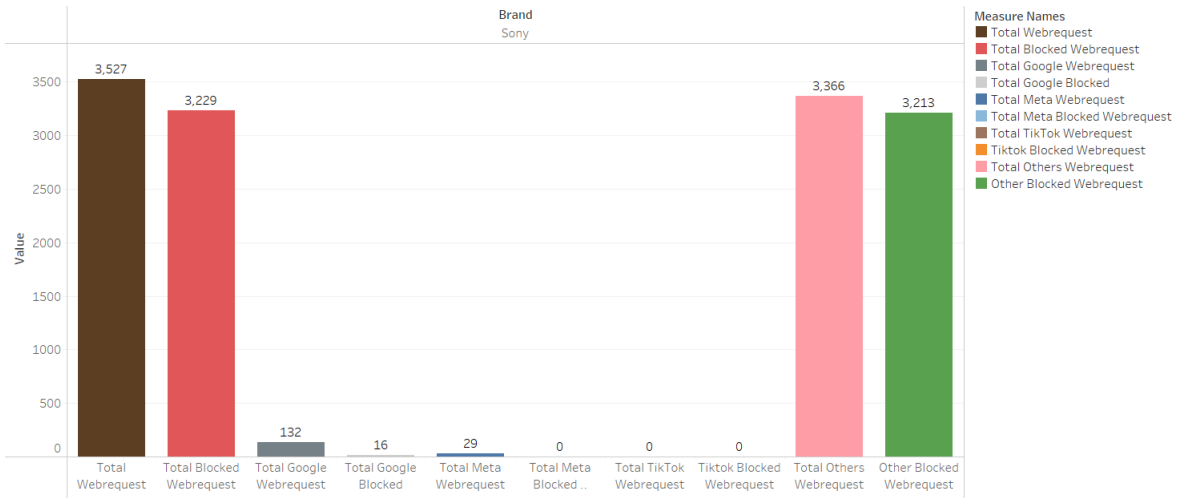


Figure 3.22: 24 hours Total Web Request and Blocked Web Request Comparison in Sony

Web Request and Blocked Request(Xiaomi)

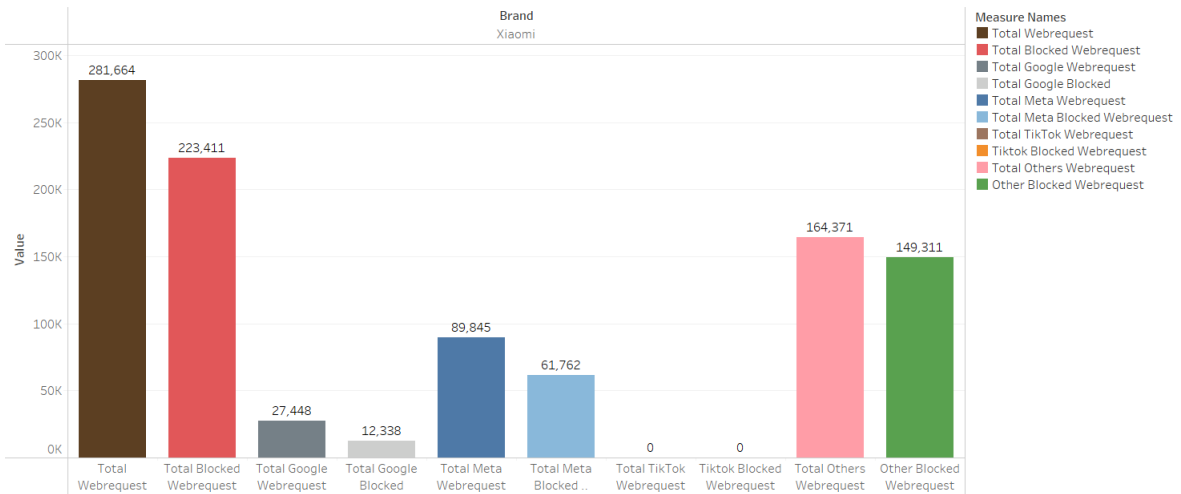


Figure 3.23: 24 hours Total Web Request and Blocked Web Request Comparison in Xiaomi

Chapter 4

Result & Analysis

Analysis from our created dataset, a total of 82.05% of requests were blocked, and the calculation method was Equation 4.1. Then, the percentage of First Party Google Ads was identified from the entire blocked list (Equation 4.2) which is 5.79%. Moreover, for First Party Meta Ads, using formula (Equation 4.3) which is 26.52%. On the other hand, when it comes to TikTok(ByteDance), because of getting very small amount of devices where TikTok is installed, the ads & tracking calculations were further excluded from the calculations.

$$\text{Blocked Ratio} = \frac{\text{Total Number of Blocked Web Requests}}{\text{Total Number of Web Requests}} * 100 \quad (4.1)$$

$$\text{Google Ads Ratio(Blocked)} = \frac{\text{Total Number of Blocked Google Ads}}{\text{Total Number of Blocked Web Requests}} * 100 \quad (4.2)$$

$$\text{Meta Ads Ratio(Blocked)} = \frac{\text{Total Number of Blocked Meta Ads}}{\text{Total Number of Blocked Web Requests}} * 100 \quad (4.3)$$

Table 4.1: Android Version-Wise Blocked, Google & Meta Blocked Percentage

Version	Total Block %	Google Block %	Meta Block %
7	91.35	3.61	9.89
8	86.60	2.81	0.64
9	84.37	6.96	2.26
10	69.63	3.89	67.37
11	82.15	7.13	34.17
12	78.22	10.36	44.77

In the data table 4.1, Figure 4.1 & Figure 4.2 briefly shows the blocked percentage comparison by Android version. Version 7 has the most blocked web requests,

91.35% and version 10 has the lowest blocked web requests which is 69.63% out of 100%, respectively.

In the analysis shown in Table 4.2, Honor has the highest percentage of blocked (ad & tracker) web requests which are 92.77%, and Xiaomi’s sub-brand Redmi has the 3rd highest percentage of blocked (ad & tracker) requests is 84.89%. On the other hand, Nothing Phone has the 30.07% of blocked web requests with almost the same amount of apps. Analysis of the web requests shows that Redmi phones are highly bloated with so many Xiaomi apps. Those apps continuously track users’ behavior and are responsible for Redmi’s high percentages compared to other brands.

Table 4.2: Brand-Wise Blocked, Google & Meta Blocked Percentage

Brand	Total Blocked %	Google Blocked %	Meta Blocked %
Honor	92.77	1.39	0.07
HUAWEI	74.12	5.72	0.57
Infinix	77.94	5.78	53.34
Lenovo	84.00	5.79	0.13
Nokia	53.97	6.31	84.29
Nothing	30.07	25.04	58.56
OnePlus	80.62	9.08	36.68
OPPO	83.13	4.58	32.64
POCO	82.41	4.21	29.07
Realme	74.37	4.75	53.78
Redmi	84.89	11.48	29.79
Samsung	81.07	6.98	41.30
Sony	91.55	0.50	0.00
Xiaomi	79.32	5.52	27.65

OS Version Wise Blocked Web Request Percentage (Version 7,8,9)

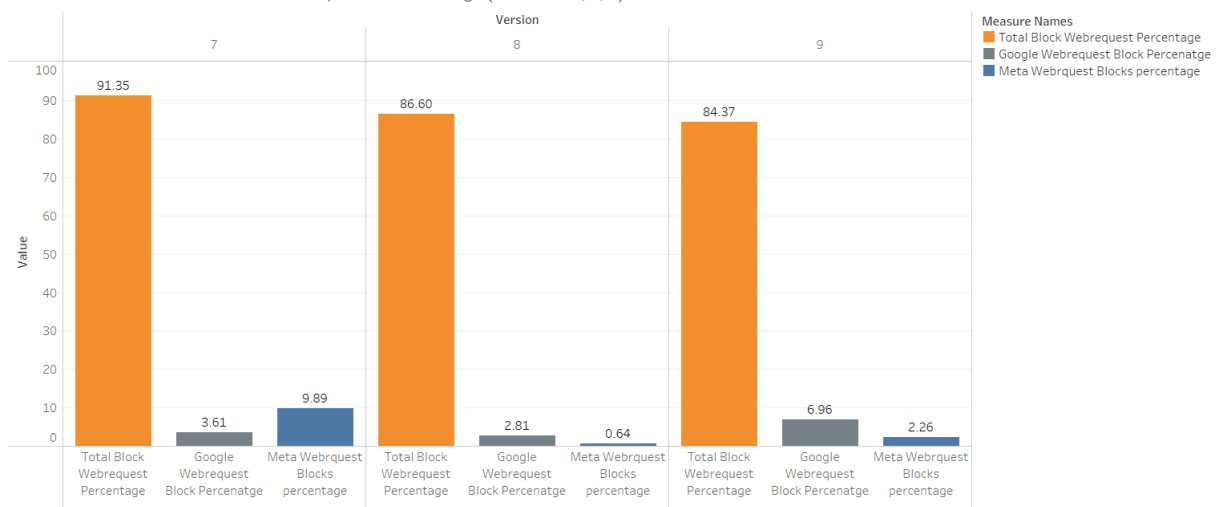


Figure 4.1: Version-Wise Blocked Percentage (Android v7, 8 & 9)

OS Version Wise Blocked Web Request Percentage (Version 10,11,12,13)

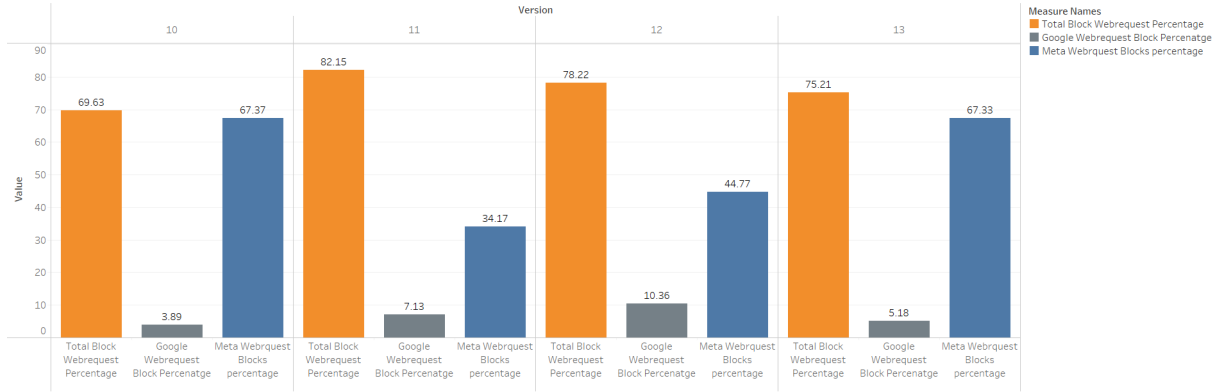


Figure 4.2: Version-Wise Blocked Percentage (Android v10, 11, 12 & 13)

Brand Wise Blocked Web Request Percentage

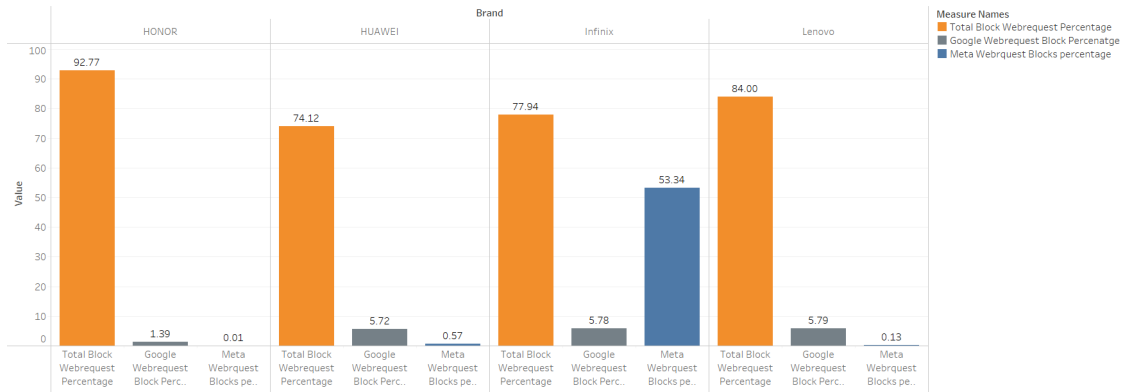


Figure 4.3: Brand-Wise Blocked Percentage (Honor, HUAWEI, Infinix & Lenovo)

Brand Wise Blocked Web Request Percentage

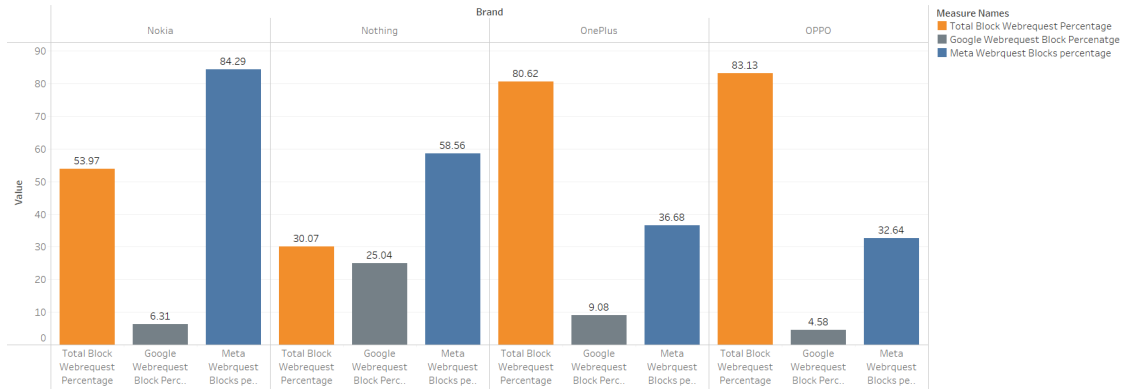


Figure 4.4: Brand-Wise Blocked Percentage (Nokia, Nothing, OnePlus & OPPO)

Brand Wise Blocked Web Request Percentage

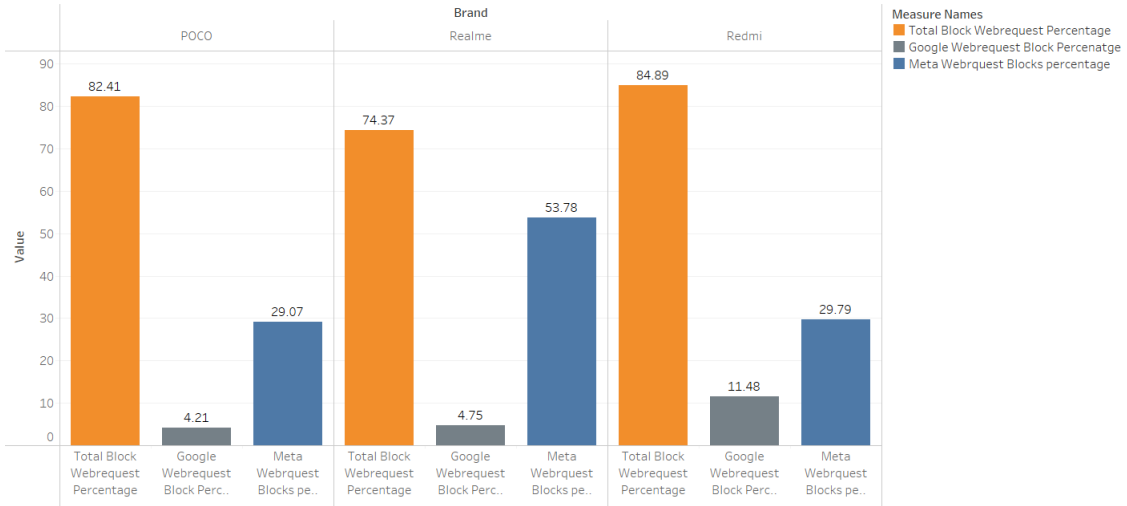


Figure 4.5: Brand-Wise Blocked Percentage (POCO, Realme & Redmi)

Brand Wise Blocked Web Request Percentage

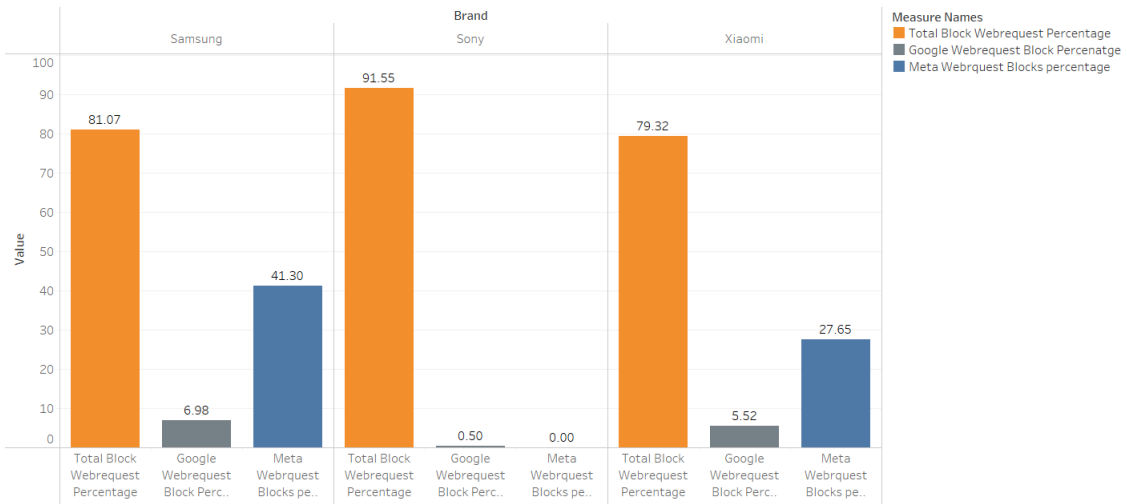


Figure 4.6: Brand-Wise Blocked Percentage (Samsung, Sony & Xiaomi)

While analyzing the brand & version, we have found that the amount of ads & trackers varies heavily even with the same amount of applications from brand to brand. Sometimes manufacturing brands like Xiaomi's Redmi are responsible for the tracking. Some brands also allow access to Third Party companies like Facebook, Spotify, and so on to track users' data [18]. Figure 4.7 & 4.8 shows an overview of the tracking percentages on different android versions of different manufacturers/brands.

Brand - Version Blocked Web Request Percentage (Version 7,8,9)



Figure 4.7: Brand-Version Web Request Percentage (Android v7, 8 & 9)

Brand - Version Blocked Web Request Percentage (Version 10,11,12,13)

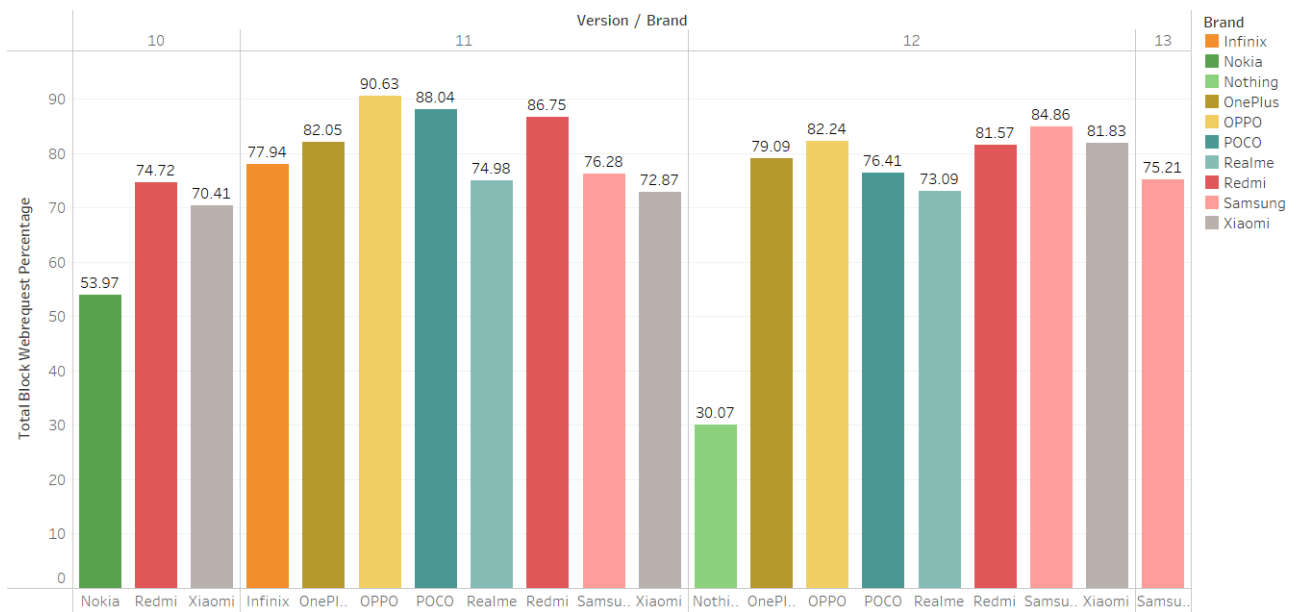


Figure 4.8: Brand-Version Web Request Percentage (Android v10, 11, 12 & 13)

The brand & version wise blocked data clearly shows why the manufacturing brand is important. Even in the same OS versions, different results are found for different brands.

24 Hours Analysis

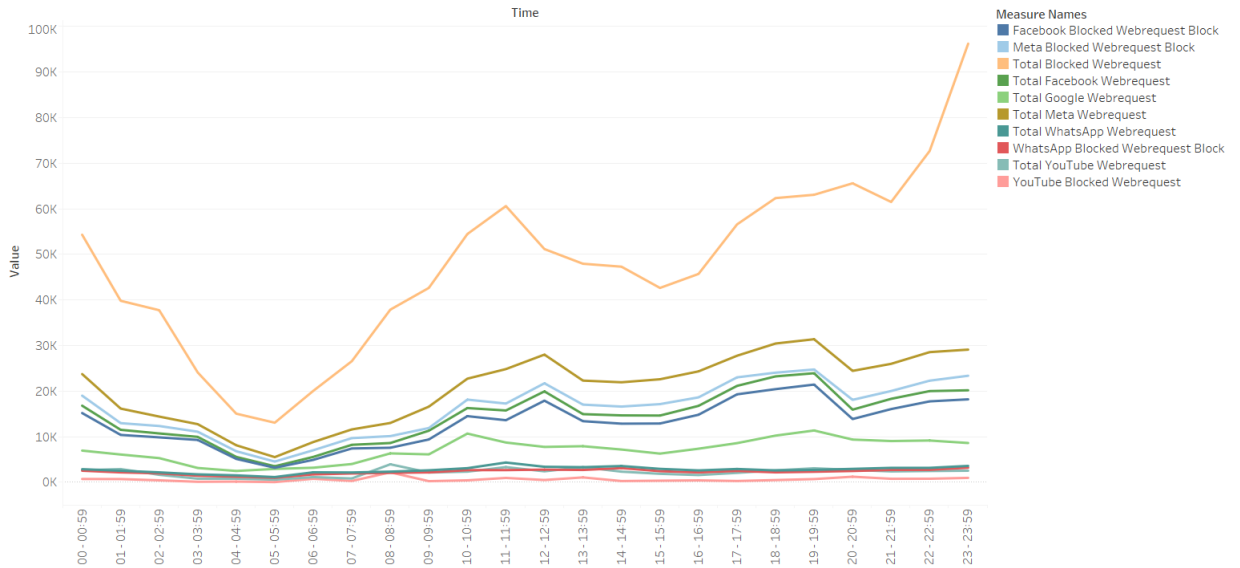


Figure 4.9: 24 Hours Analysis

Figure 4.9 shows the complete overview of the analysis. It shows the simplified view of the whole data in a single image. The categorical data shows how a user is being tracked every hour simultaneously by different ad companies. The tracking behavior remains almost the same even when the user is sleeping/not using the phone. The Figure 4.10 to Figure 4.12 shows the comparison more precisely.

24 Hours Graph(Meta)

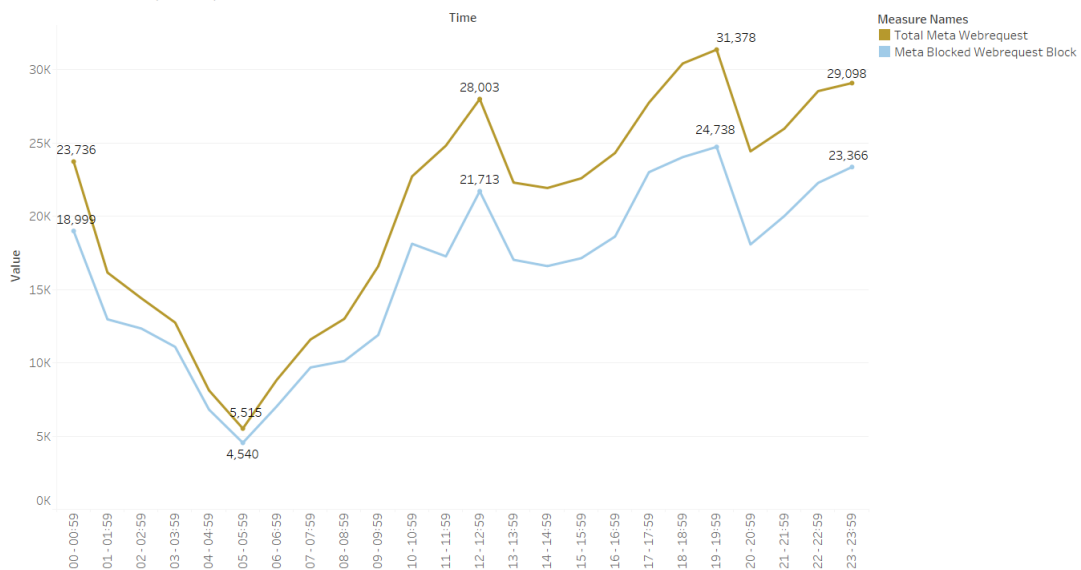


Figure 4.10: 24 Hours Analysis of Meta (Company)

The graph shows that Meta tracks its users even when the user is not using his/her devices. The amount of number of tracks varies from around 50 to 1000 times every single hour. Facebook tracks users behavior heavily through cookies and other tracking technologies on its website and mobile app [33]. This data is collected and used to personalize and improve the users experience, as well as for targeted advertising. Facebook also tracks users across the web through its social media plug-ins and pixel tracking. The data collected can include browsing history, search history, location data, and device information [33].

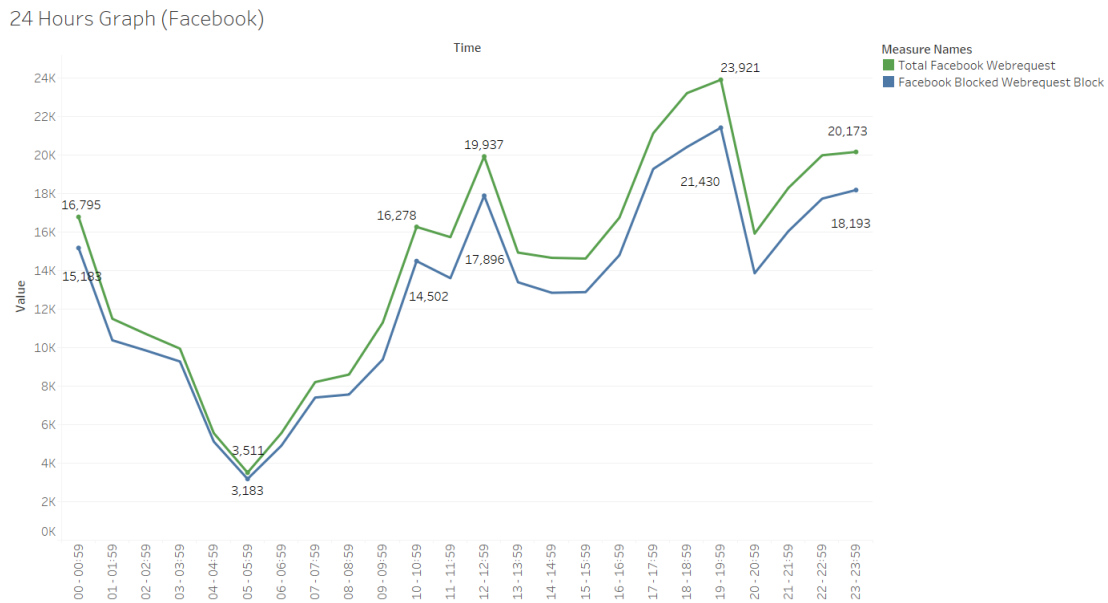


Figure 4.11: 24 Hours Analysis of Facebook Android

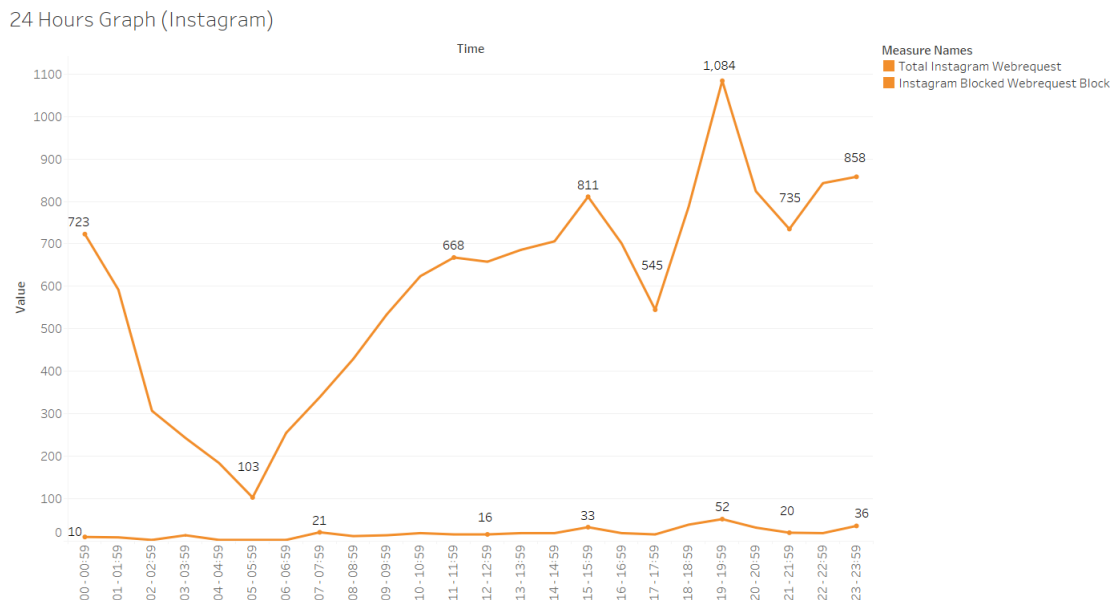


Figure 4.12: 24 Hours Analysis of Instagram

Though Facebook’s parent company Meta claims that WhatsApp is completely safe and advertisement free, our research has found that, like every other platform, WhatsApp also tracks user activity very heavily [24]. WhatsApp tracks user behavior to some extent, similar to other mobile apps and websites. WhatsApp uses tracking technologies like cookies and analytics software to collect data on how users interact with the app, such as their device information, location data, and browsing history [25]. This data is used to personalize the user experience, improve the app’s performance, and help troubleshoot issues. WhatsApp also shares the data with parent company Meta which is infamous for its repulsive practice of its users’ data. Figure 4.13 shows the total number of web requests & blocked web requests.

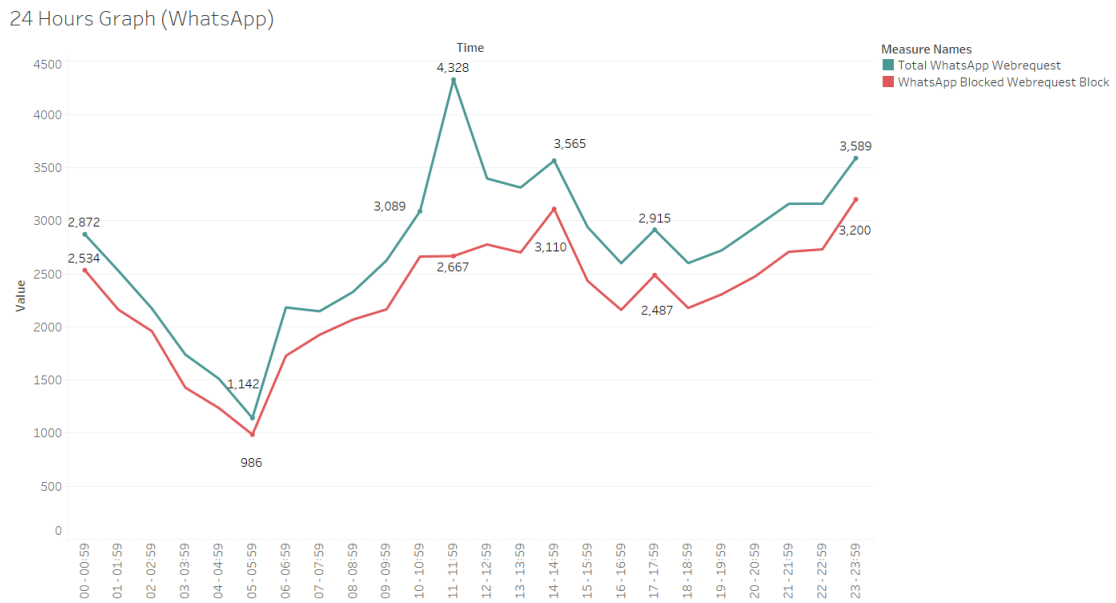


Figure 4.13: 24 Hours Analysis of WhatsApp

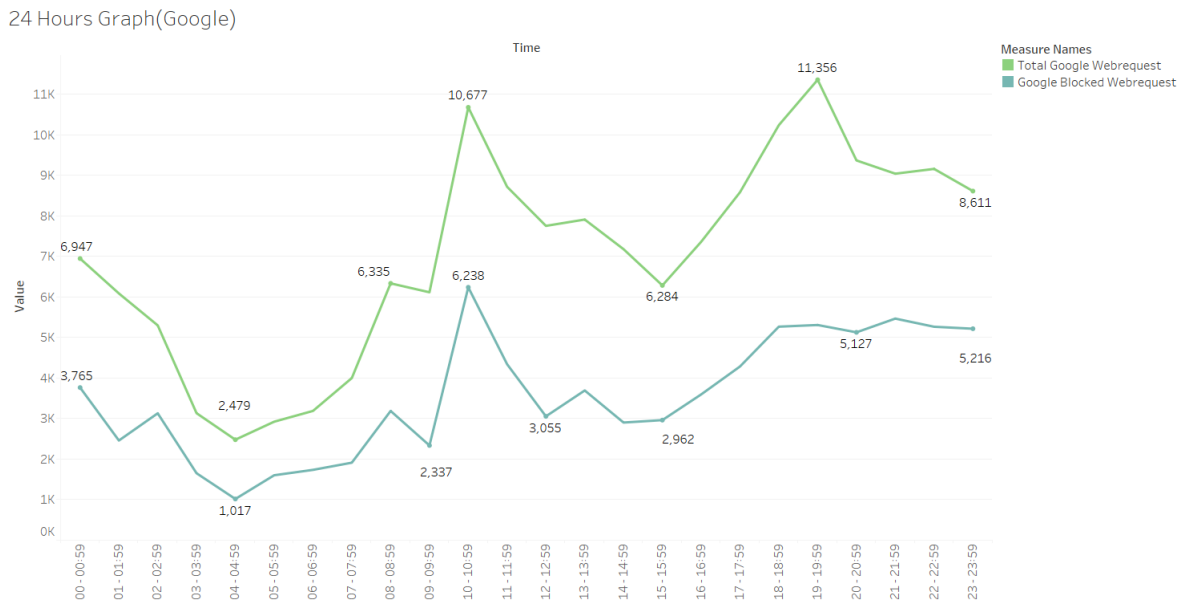


Figure 4.14: 24 Hours Analysis of Google (Company)

24 Hours Graph (Youtube)

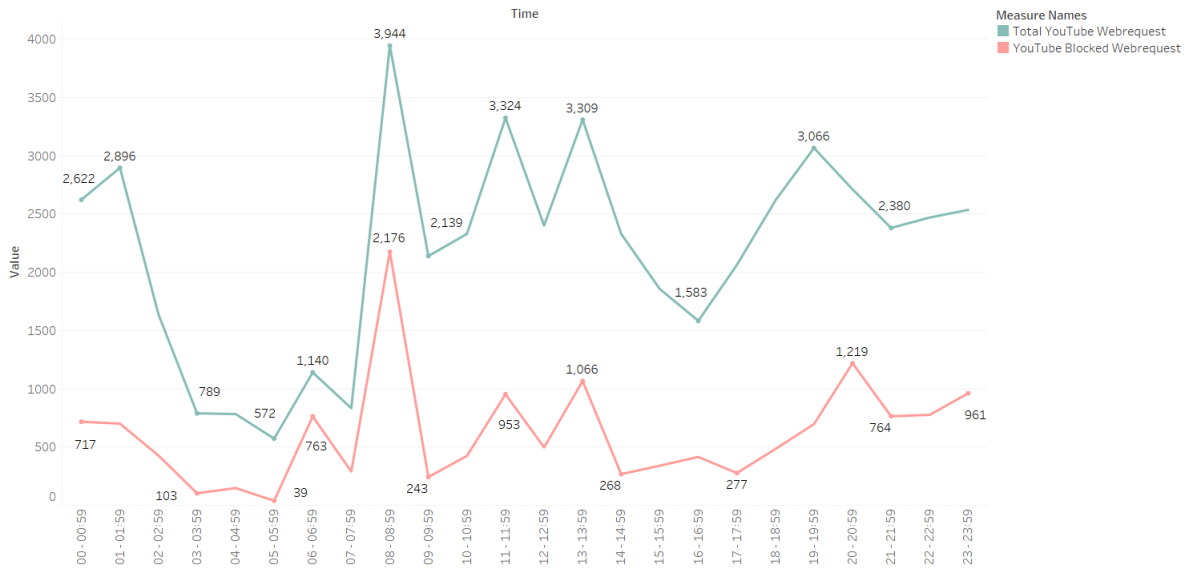


Figure 4.15: 24 Hours Analysis of YouTube

All Webrequest VS All Blocked Webrequest

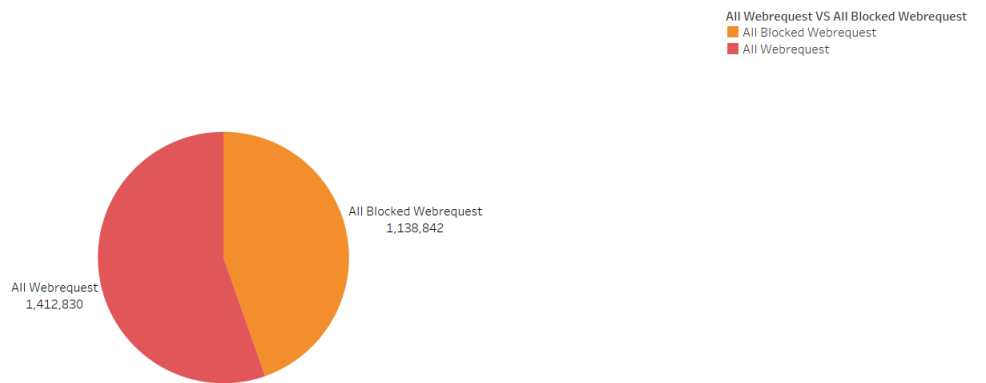


Figure 4.16: All Web Request VS All Blocked Web Request

First Party refers to entities that are directly associated with the website or application being visited, such as the website owner or operator. Third Party refers to entities that are not directly associated with the website or application being visited, such as advertisers or analytics providers [31]. For example, when a user visits a website, the website itself is considered a First Party while any ads or tracking scripts on the site from other companies would be considered Third Parties. While analyzing the data of Google & Meta, we have found that, Google is widely dependent on Third Parties to push ads and track user, but Meta is solely dependent on its First Party data. This means, it is dependent on its own Facebook, Instagram, WhatsApp and so on data to track the users. Though, Meta also collects data from various websites and apps using its ‘Graph’ [9].

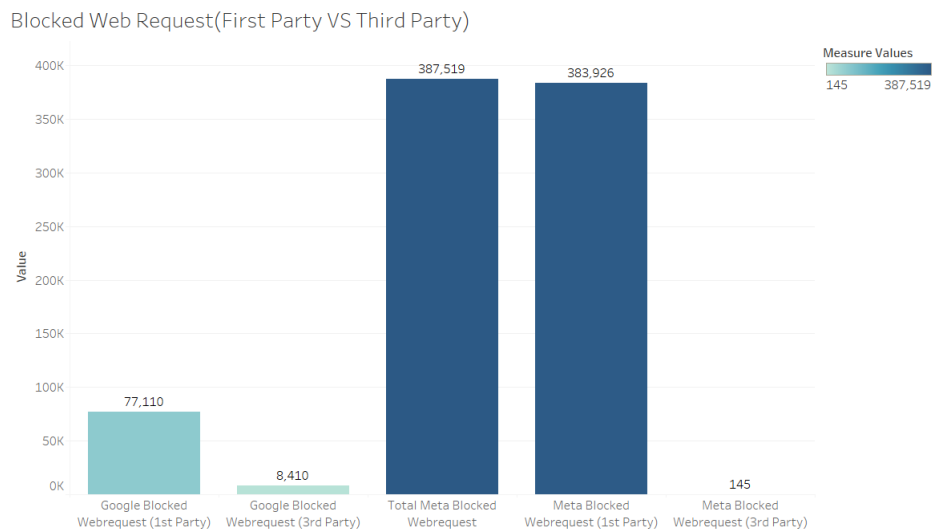


Figure 4.17: Blocked Web Request (First Party VS Third Party)

Third Party Request Comparision (Google& Meta)

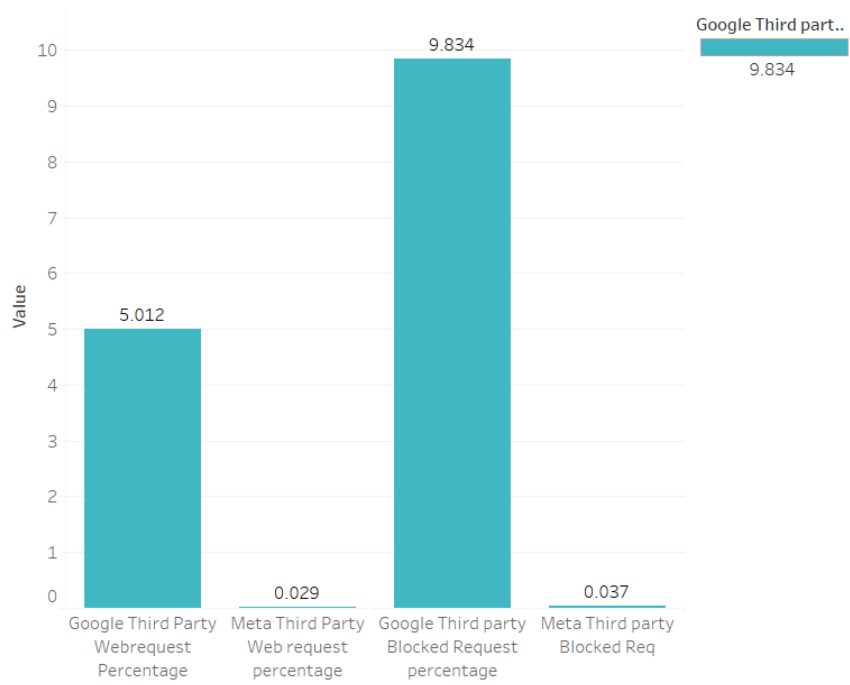


Figure 4.18: Third Party Comparison (Google & Meta)

YouTube is a freemium service, though its main revenue comes from ads [30]. While analyzing the data, we have found that more than 24.61% of web requests of Google come from YouTube only, as the service is widely available and popular now. The blocked percentage is also very high which is around 16.05% of total number blocked Google web requests.

Google First Party Blocked VS Youtube Blocked

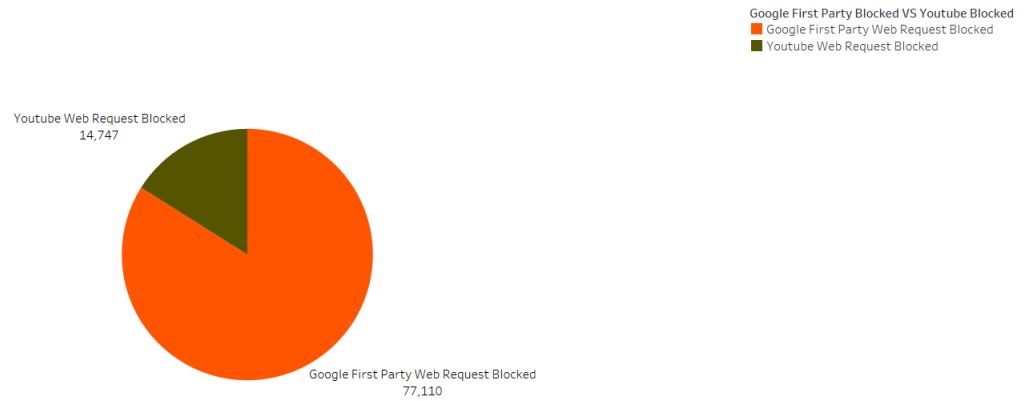


Figure 4.19: Youtube Data usage in Google (First Party)

Google First Party Webrequest VS Youtube Webrequest



Figure 4.20: YouTube Ads on Google

Youtube Web Request VS Blocked Web Request

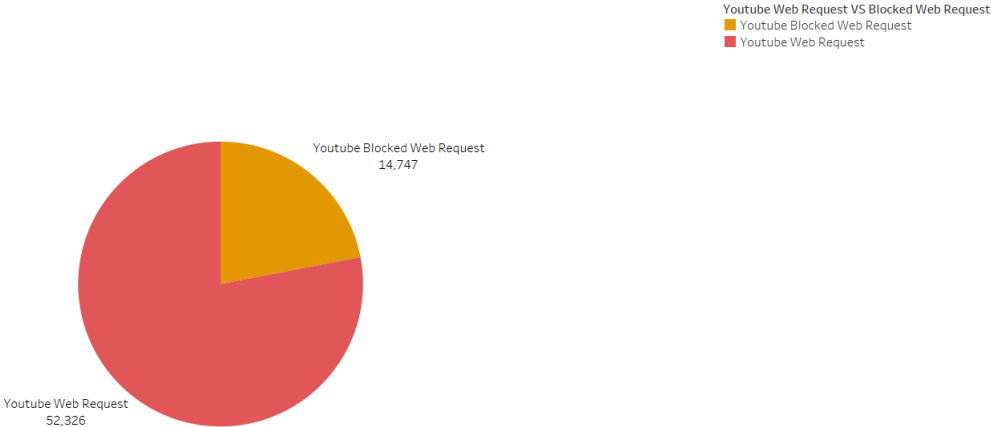


Figure 4.21: YouTube Web Request VS Blocked Web Request

Figure 4.22 shows that the total number of web requests and blocked web requests. The ratio is around 3:1. That means, around 33.64% web request comes from Google just for serving ads/tracking users.

Google Webrequest VS Blocked Webrequest

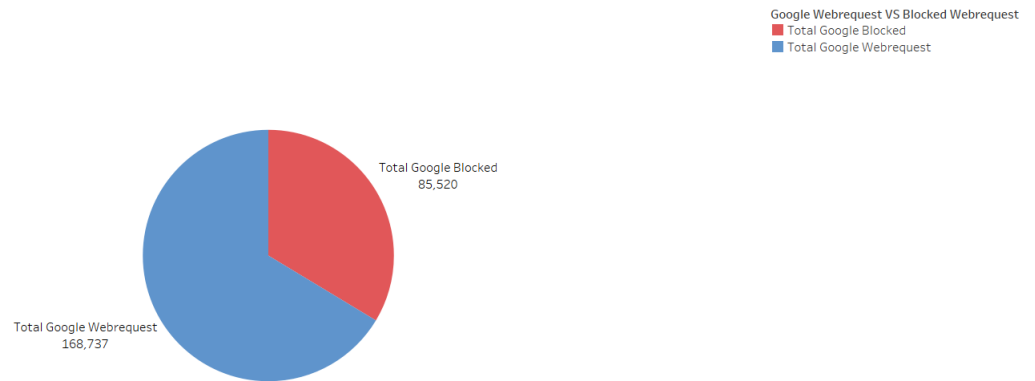


Figure 4.22: Google Web Request VS Blocked Web Request

From the dataset, we have found that Meta 84.85% blocked web requests (which is mainly for tracking users/serving precise ads) comes from Facebook Android app, 15.03% comes from WhatsApp & 0.12% comes from Instagram. The Figure 4.23 to 4.26 shows the in-depth analysis of that.

Meta Block Request

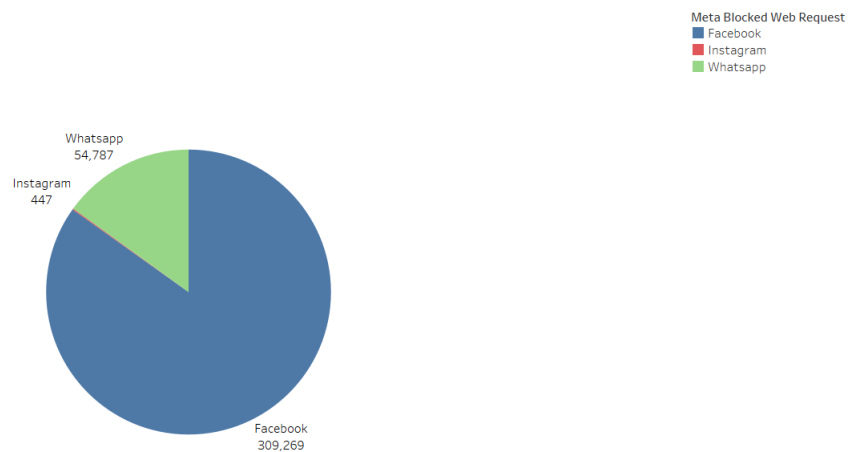


Figure 4.23: Meta Blocked Web Request in Category

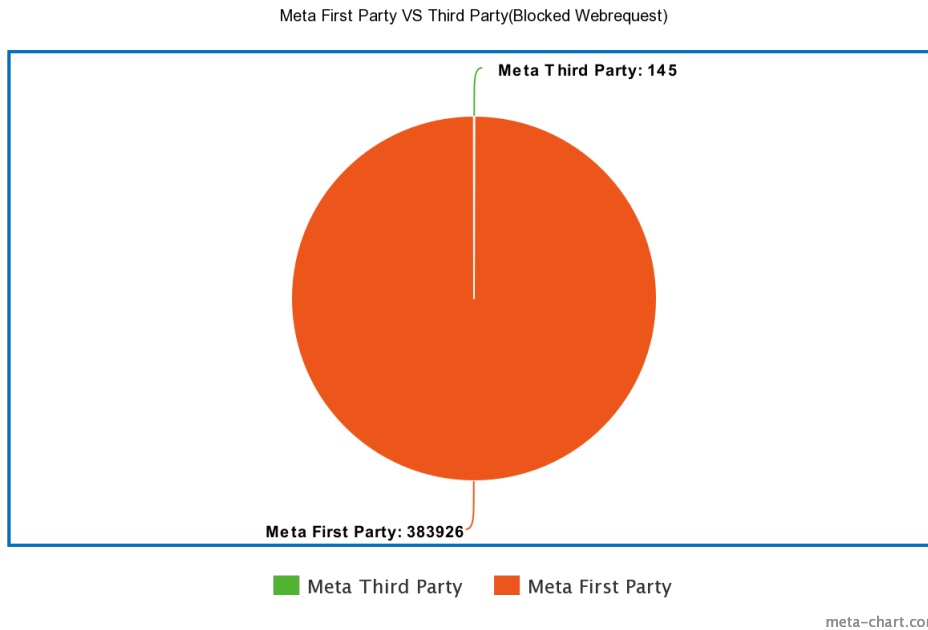


Figure 4.24: Meta First Party VS Third Party (Blocked Web Request)

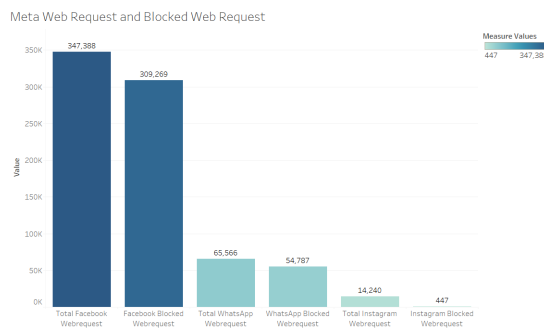


Figure 4.25: Meta Web Request VS Blocked Web Request in Category

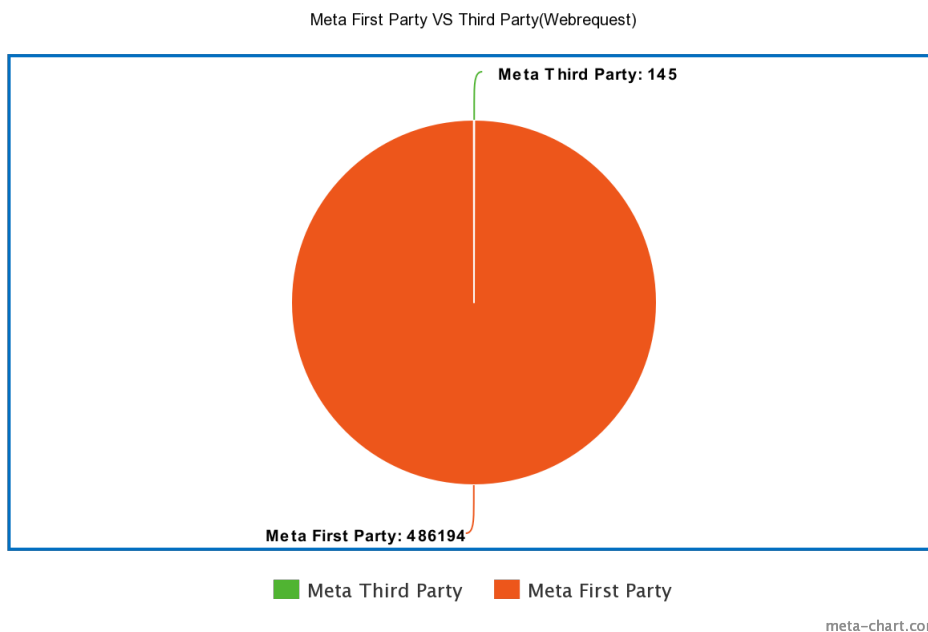


Figure 4.26: Meta First Party VS Third Party (Web Request)

Chapter 5

Conclusion

Data privacy in Android is becoming a significant problem because of Third Party tracking and excessive app permissions. Despite the efforts made by the EU (GDPR) and the USA, the amount of tracking is rising dramatically daily. This study aims to show the tracking and ad request results so users can easily understand how they are affected by their mobile brand and versions regarding data privacy.

The study has many drawbacks in the data processing. In the future, the study aims to make a behavior pattern on tracking and ad requests using machine learning algorithms to enhance users' privacy.

Future Plan

We will develop an android app that will be minimal and lightweight, and by default, it will block malicious ads & trackers. As some apps do not work properly without ads, there will be dynamic options to grant permission for showing ads for a single time. Also, the app will try to analyze the installed apps module to understand which one is to block and which one is to show. The data can be used in the benchmarking process. Our future plan also includes some motivations like the data can also help to improve the DND (Do Not Disturb) services, physical activity detection, and so on.

Bibliography

- [1] 2012. [Online]. Available: <https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009/>.
- [2] P. Pearce, A. Felt, G. Nunez, and D. Wagner, "Addroid: Privilege separation for applications and advertisers in android," *ASIACCS 2012 - 7th ACM Symposium on Information, Computer and Communications Security*, May 2012. DOI: 10.1145/2414456.2414498.
- [3] B. Livshits and J. Jung, "Automatic mediation of privacy-sensitive resource access in smartphone applications," Aug. 2013, pp. 113–130.
- [4] R. Bhoraskar, S. Han, J. Jeon, *et al.*, "Brahmastra: Driving apps to test the security of third-party components," Aug. 2014.
- [5] K.-H. Yeh, N.-W. Lo, and C.-Y. Fan, "An analysis framework for information loss and privacy leakage on android applications," in *2014 IEEE 3rd Global Conference on Consumer Electronics (GCCE)*, 2014, pp. 216–218. DOI: 10.1109/GCCE.2014.7031192.
- [6] 2015. [Online]. Available: <https://www.statista.com/statistics/467160/forecast-of-smartphone-users-in-china/>.
- [7] R. Roshandel and R. Tyler, "User-centric monitoring of sensitive information access in android applications," in *2015 2nd ACM International Conference on Mobile Software Engineering and Systems*, 2015, pp. 144–145. DOI: 10.1109/MobileSoft.2015.36.
- [8] S. S. Shinde and S. S. Sambare, "Enhancement on privacy permission management for android apps," in *2015 Global Conference on Communication Technologies (GCCT)*, 2015, pp. 838–842. DOI: 10.1109/GCCT.2015.7342779.
- [9] 2017. [Online]. Available: <https://developers.facebook.com/docs/graph-api/overview/>.
- [10] M. Li, W. Wang, P. Wang, *et al.*, "Libd: Scalable and precise third-party library detection in android markets," in *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*, 2017, pp. 335–346. DOI: 10.1109/ICSE.2017.38.
- [11] S. E. Polykalas, G. N. Prezerakos, F. D. Chrysidou, and E. D. Pylarinou, "Mobile apps and data privacy: When the service is free, the product is your data," in *2017 8th International Conference on Information, Intelligence, Systems Applications (IISA)*, Aug. 2017, pp. 1–5. DOI: 10.1109/IISA.2017.8316392.
- [12] R. Talreja and D. Motwani, "Sectrans: Enhancing user privacy on android platform," in *2017 International Conference on Nascent Technologies in Engineering (ICNTE)*, 2017, pp. 1–4. DOI: 10.1109/ICNTE.2017.7947884.
- [13] V. F. Taylor, A. R. Beresford, and I. Martinovic, "There are many apps for that: Quantifying the availability of privacy-preserving apps," in *Proceedings*

- of the 10th ACM Conference on Security and Privacy in Wireless and Mobile Networks, ser. WiSec '17, New York, NY, USA: Association for Computing Machinery, 2017, pp. 247–252, ISBN: 9781450350846. DOI: 10.1145/3098243.3098266. [Online]. Available: <https://doi.org/10.1145/3098243.3098266>.
- [14] M. Van Kleek, I. Liccardi, R. Binns, J. Zhao, D. J. Weitzner, and N. Shadbolt, “Better the devil you know: Exposing the data sharing practices of smartphone apps,” in *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, ser. CHI '17, New York, NY, USA: Association for Computing Machinery, 2017, pp. 5208–5220, ISBN: 9781450346559. DOI: 10.1145/3025453.3025556. [Online]. Available: <https://doi.org/10.1145/3025453.3025556>.
- [15] R. Binns, U. Lyngs, M. Van Kleek, J. Zhao, T. Libert, and N. Shadbolt, “Third party tracking in the mobile ecosystem,” in *Proceedings of the 10th ACM Conference on Web Science*, ser. WebSci '18, New York, NY, USA: Association for Computing Machinery, 2018, pp. 23–31, ISBN: 9781450355636. DOI: 10.1145/3201064.3201089. [Online]. Available: <https://doi.org/10.1145/3201064.3201089>.
- [16] Y. He, B. Hu, and Z. Han, “Dynamic privacy leakage analysis of android third-party libraries,” in *2018 1st International Conference on Data Intelligence and Security (ICDIS)*, 2018, pp. 275–280. DOI: 10.1109/ICDIS.2018.00051.
- [17] Y. He, X. Zhao, and C. Wang, “Privacy mining of large-scale mobile usage data,” in *2019 IEEE International Conference on Power, Intelligent Computing and Systems (ICPICS)*, 2019, pp. 81–86. DOI: 10.1109/ICPICS47731.2019.8942559.
- [18] May 2020. [Online]. Available: <https://www.firstpost.com/tech/news-analysis/samsung-teams-up-with-facebook-to-train-its-offline-retailers-go-digital-8397651.html>.
- [19] 2021. [Online]. Available: <https://www.statista.com/statistics/1058715/china-mobile-applications-available-app-stores/>.
- [20] H. Chen, Y. Gu, P. Wang, J. Dong, and Y. Ren, “Research on privacy data protection in mobile applications,” in *2021 33rd Chinese Control and Decision Conference (CCDC)*, 2021, pp. 4912–4915. DOI: 10.1109/CCDC52312.2021.9602169.
- [21] O. Haggag, S. Haggag, J. Grundy, and M. Abdelrazek, “Covid-19 vs social media apps: Does privacy really matter?” In *2021 IEEE/ACM 43rd International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*, 2021, pp. 48–57. DOI: 10.1109/ICSE-SEIS52602.2021.00014.
- [22] K. Kollnig, R. Binns, M. Van Kleek, *et al.*, *Before and after gdpr: Tracking in mobile apps*, Dec. 2021.
- [23] C. F. Libaque-Sáenz, S. F. Wong, Y. Chang, and E. R. Bravo, “The effect of fair information practices and data collection methods on privacy-related behaviors: A study of mobile apps,” *Information Management*, vol. 58, no. 1, p. 103284, 2021, ISSN: 0378-7206. DOI: <https://doi.org/10.1016/j.im.2020.103284>. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S0378720618301599>.
- [24] K. O’Flaherty, *All the data whatsapp and instagram send to facebook*, Jul. 2021. [Online]. Available: <https://www.wired.co.uk/article/whatsapp-instagram-facebook-data>.

- [25] C. Page, “Whatsapp clarifies facebook data-sharing as users flock to rival signal,” *Forbes*, Jan. 2021. [Online]. Available: <https://www.forbes.com/sites/carlypage/2021/01/13/whatsapp-clarifies-facebook-data-sharing-as-users-flock-to-rival-signal/?sh=bab0c2716396>.
- [26] G. Tangari, M. Ikram, K. Ijaz, M. A. Kaafar, and S. Berkovsky, “Mobile health and privacy: Cross sectional study,” *BMJ*, vol. 373, 2021. DOI: 10.1136/bmj.n1248. eprint: <https://www.bmj.com/content/373/bmj.n1248.full.pdf>. [Online]. Available: <https://www.bmj.com/content/373/bmj.n1248>.
- [27] 2022. [Online]. Available: <https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/>.
- [28] 2022. [Online]. Available: <https://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/>.
- [29] 2022. [Online]. Available: <https://www.statista.com/outlook/dmo/app/worldwide>.
- [30] 2022. [Online]. Available: <https://www.statista.com/statistics/289657/youtube-global-quarterly-advertising-revenues/>.
- [31] S. Bernazzani, *A basic definition of first party, second party, third party data*, Oct. 2022. [Online]. Available: <https://blog.hubspot.com/service/first-party-data>.
- [32] J. Hildenbrand, *Androidcentral*, Aug. 2022. [Online]. Available: <https://www.androidcentral.com/android-security-bulletin>.
- [33] *The New York Times*, 2023. [Online]. Available: <https://www.nytimes.com/2018/04/11/technology/facebook-privacy-hearings.html>.