Characterization of Early Smartwatch Apps

Jagmohan Chauhan*†, Suranga Seneviratne †, Mohamed Ali Kaafar †, Anirban Mahanti †, Aruna Seneviratne *†
*School of EET University of New South Wales , †NICTA, Australia
* Email: †first name.last name@nicta.com.au

***Abstract*—Wearable smart devices are already amongst us. Currently, smartwatches are one of the key drivers of the wearable technology and are being used by a large population of consumers. This paper takes a first look at this increasingly popular technology with a systematic characterization of the smartwatch app markets. We conduct a large scale analysis of three popular smartwatch app markets: *Android Wear*, *Samsung*, and *Apple*, and characterize more than 14,000 smartwatch apps in multiple aspects such as prices, number of developers and categories. Our analysis shows that approximately 41% and 30% of the apps in Android Wear and Samsung app markets are *Personalization* apps that provide watch faces. Further, we provide a generic taxonomy for apps on all three platforms based on their packaging and modes of communication, that allow us to investigate apps with respect to privacy and security. Finally, we study the privacy risks associated with the app usage by identifying *third party trackers* integrated into these apps and *personal information* leakage through network traffic analysis. We show that a higher percentage of Apple apps (62%) are connected to third party trackers compared to Samsung (36%) and Android Wear (46%).**

I. Introduction

Smartwatches are one of the most popular wearable device type in today's market as their sales reached 5 million units [1] in 2015 and is expected to grow further to realize 101 million devices by 2020 [2]. Apple watches, have acquired an impressive market share of 75.5% followed by 7.5% for Samsung devices [3]. Multiple vendors such as Motorola, and LG have adopted Google's *Android Wear* operating system, making it the second most widely used smartwatch operating system next to Apple's watchOS [4]. Similar to smartphones, the predominant component of the smartwatch ecosystem is the availability of third party apps. As of September 2015, Apple's app store was composed of more than 10,000 smartwatch apps [5]. Google Play has around 4,000 apps for Android Wear devices [6], while Samsung Gear Store was reported to contain over 1,000 smartwatch apps [7].

Despite the overwhelming interest in smartwatches and the increasing importance of apps, we have a very little understanding of the types of available apps and associated characteristics in different app markets. There is also a lack of knowledge of privacy and security issues, which exists in current smartwatch apps. Early evolving ecosystems are very well susceptible to privacy and security threats as known from previous research on smartphone apps [8], [9]. This paper takes a first step in addressing aforementioned knowledge gaps by presenting a data-driven study of the population of apps from three app markets.

Overall, this paper makes the following contributions:

- We investigate a dataset of over 14,000 smartwatch apps across three app markets: Android Wear, Samsung, and Apple and provide characteristics and insights from the descriptive statistics of the studied apps related to app prices, categories, and number of developers.
- We provide a generic taxonomy for the apps on all three platforms based on their packaging and modes of communication. This allows one to obtain insights into how the current smartwatch apps are designed and their potential ability to leak sensitive data.
- We perform static analysis on app code and show that Apple apps connect to more trackers compared to Android Wear and Samsung.
- By analyzing the collected network traces after executing 28.2% (1,813) of all free apps across the three platforms, we show that although unique device information of the smartwatch or health related information is never leaked, 6% of Android Wear apps and 11% of Apple apps leak user activities from the smartwatch to third party trackers.

The rest of the paper is organized as follows. Section II lists the related work and Section III explains our data collection procedure. Section IV presents the characterization of the app markets in terms of descriptive statistics and introduces a generic taxonomy for smartwatch apps. Privacy and security threats associated with the current apps are presented in Section V. Section VI concludes the paper.

II. Related Work

We describe related work in the areas of **i)** measurement studies of smartphone app markets, and **ii)** measurement studies related to security and privacy of wearables.

i) Smartphone app markets: Several studies based on large scale crawls of smartphone app markets have been conducted [10], [11]. Heureuse et al. [10] analyzed four popular app markets including Google Play Store and Apple App Store and presented statistics on the market growth, app pricing, and other app attributes such as app sizes, categories, ratings, and downloads. More recently, Viennot et al. [11] analyzed source codes of over 880,000 free apps and characterized the ads library usage and duplicative content.

Multiple work investigated PII leakages and tracker connectivity in smartphone apps [12], [13]. Grace et al. [12] detected a number of leaks by advertisement libraries related to user's call logs, account information, and phone number by analyzing 100,000 free apps collected between March and May 2011. Leontiadis et al. [13] studied the requested permissions of

978-1-5090-1941-0/16/$31.00 ©2016 IEEE

around 250,000 Android apps and showed that free apps asked for more "risky" permissions.

ii) Wearables security and privacy: Recently, HP [14] studied the security and privacy vulnerabilities of 10 popular smartwatches from an OS and in-built application's perspective. The study found out that intercepting communication between the smartwatch and the smartphone is trivial. Wang et al. [15] highlighted how the smartwatch motion sensors can leak what the user is typing on the keyboard of a laptop.

III. Datasets in Use

Similar to the smartphone apps, smartwatch apps are hosted and maintained in app markets (Google Play Store, Samsung Gear Store and Apple App Store). We now describe the app crawling methodology for each app market.

Android Wear: We discovered Android Wear apps available on Google Play Store by collecting Android Wear app identifiers from two alternative app markets, *Android Wear Center* [16] and *Goko* [17] that lists Android Wear apps. App identifiers were then used to access the corresponding Google Play Store page and download app metadata and executable using a Python based scraper. We collected metadata of 3,623 apps out of which 2,332 app were free. This represents 90% of the reported number of Android Wear apps (4,000 apps[1]) as of May 2015 [6].

Samsung: Data collection from Samsung Gear Store was more challenging. First, Samsung Gear apps are only accessible from the Samsung Gear app installed on a Samsung smartphone and there were no alternative markets listing Samsung smartwatch apps. By inspecting the traffic generated when users browse a particular category of smartwatch apps via the Samsung Gear app on the smartphone, we observed that Samsung Gear app sends HTTP POST requests with a category identifier and a number of items to fetch from the app store. In response, the app stores provide an XML file containing the metadata of apps belonging to the requested category. Our app scraping technique was then to forge HTTP POST requests for each app category from the desktop and in turn receiving the metadata for an exhaustive list of apps under that category. We collected metadata of 1,789 distinct apps which approximately tallies with the figure reported [7].

Downloading the app executable was more challenging as the URL called when a user installs an app via the Samsung Gear App includes a dynamic identifier. The URL is valid only for a short period of time once the user clicks the install button. Thus, we had to resort to a semi-manual setup. The methodology consists of manually installing an app on a Samsung smartwatch and automatically capturing the requested URL by inspecting the network traffic. While the new app is being installed on the smartwatch, we download the app executable on a desktop by visiting the "temporary" URL separately. Out of 1,789 discovered app identifiers, we downloaded 700 apps that were free.

Apple: Similarly to Android Wear, we crawled WatchAware [18] app market to discover app identifiers. We then accessed the iTunes pages of these apps and downloaded the corresponding metadata. To download app executable on a desktop, we automatically replayed the URL for each app in a web browser and generated a click event on the *install* button on the webpage URL. We collected metadata of 9,355 apps and downloaded app executable of 5,615 apps that were free. This covers approximately 90% of the total number of Apple Watch apps (10,000) reported as of September 2015 [5].

The data was collected during September, 2015 and Table I provides a summary of the three datasets.

TABLE I: Summary of the datasets

	Android Wear	Samsung	Apple
Total apps	3,623	1,687	9,355
Free apps	2,332 (64.37%)	700 (41.49%)	5,615 (60.02%)
Paid apps	1,291 (35.63%)	987 (58.51%)	3,740 (39.98%)
Categories	37	8	22
Developers	1,789	394	5,436

IV. Analysis of smartwatch Apps

A. Characterization of the Apps

In this section we provide a characterization of the smartwatch apps found in the three app markets. We first provide a basic description of the app market in the likes of price, developers, and categories of apps. Then, we do cross market analysis to find out the categories of apps which are early adopters driving the uptake of the wearable technology.

App Prices: Free apps represent, respectively 64%, 41% and 60% of the apps in Android Wear, Samsung and Apple (cf. Table I). Notably, Samsung provides a higher ratio of paid apps compared to free apps. We did not expect such a landscape of the smartwatch app market as this is significantly different from the smartphone "regular" app market (for Android Wear and Samsung). For instance, according to a recent report [19], 88% of apps in Google Play Store are free. However, this phenomenon of a higher ratio of paid apps was also observed in the early days of multiple smartphone app markets. Generally, the ratio of free apps increases as the number of apps increases in the app market [20]. On the other hand, Apple while having the highest number of smartwatch apps on the market, exhibits approximately the same ratio of Free/Paid smartwatch apps as iPhone/iPad apps. Similar to our estimate of the Apple app market for smartwatches, approximately 60% of the regular Apple smartphone apps are free [21].

Figure 1 shows the *cumulative distribution function (CDF)* of the paid app prices in the three app markets (in AUD). Apple apps are relatively more expensive. Approximately 20% of the Apple apps cost more than 5 AUD, unlike Android Wear and Samsung. Apple also has the highest minimum price of 1.29 AUD for the apps. For Android Wear and Samsung the minimum prices are 0.99 AUD and 0.95 AUD respectively.

App Developers: There are 5,436, 1,789 and 394 unique developers that have developed smartwatch apps for Apple, Android Wear and Samsung respectively. Figure 2 shows the

[1] These are estimated numbers collected from various reports disclosed by officials of Google, Samsung and Apple.

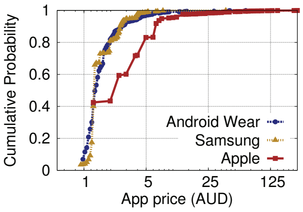


Fig. 1: App price

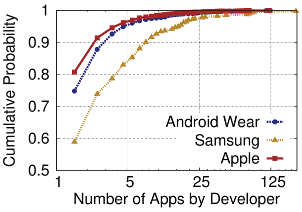


Fig. 2: Apps per developer

CDF of the number of apps per developer in each of the app market. While Android Wear and Apple show similar distributions with a significant number of developers having only one app (75% to 80%), and only 5% of the developers having more than 5 apps, Samsung shows a different distribution shape. Developers in the Samsung app market seem to develop more apps with approximately 40% of the developers having more than one published app, and 15% having more than 5 Apps. It is worth mentioning that a single developer on the Samsung app market owns 219 apps, representing about 13% of the total apps.

App Categories: The category-wise distribution of free and paid apps in Android Wear, Samsung and Apple app markets is shown in Figure 3a, Figure 3b, and Figure 3c respectively. The three app markets have different categories, which does not allow comparing the categories across the systems. Android Wear has 24 categories and the category *Games* is further divided into 18 categories (not shown for brevity reasons). Apple provides 22 categories, and Samsung has only 8.

In Samsung, the category *Clock* contributes to approximately 30% of the apps, while in Android Wear the *Personalisation* category contributes to 41% of the apps. These two categories are commonly referred as *watch face* apps, typically analogous to wallpapers applications for PCs. Apple does not allow third parties to develop watch face apps. Other notable category in Samsung is *Fonts* that contribute to approximately 4% of total apps, and contains only paid apps. Fonts apps provide a way to customise the font settings and font size on the device.

Cross market app Analysis: For a given pair of apps from two separate app markets, we calculated the *character level cosine similarity* between the two app names as well as the two developer names and decided that the two apps are same if both similarity levels are greater than 90%. We manually checked all the detected pair for false positives and found only 10. The threshold of 90% is manually set by observing the false positive rate at different values.

We found 85 common apps between Android Wear and Apple, 11 between Android and Samsung, and 5 between Samsung and Apple. We found only 2 apps that are present across all three app markets. Apps that are present in all three app markets are *iDTGV Watch*: Official application of the state-owned France train and *Winbank MyCard* which offers real-time information about discounts, loyalty programs, contests and special promotions to the customers of Greek bank Piraeus. Some example apps that are common between Android Wear and Apple are *Babel Voice Translator, Capitaine Train: train tickets,* and *Priceline Hotels, Flight & Car*.

B. Taxonomy of smartwatch apps

To investigate further on smartwatch app internals, specifically their design and communication patterns, we propose a generic taxonomy that is applicable to all three ecosystems, based on the following criteria *i) Number of binaries or packages present in each app*, *ii) Presence of communication between smartwatch and smartphone when the app executes* and *iii) Apps' ability to connect to the internet.*

Criterion (i) helps to understand how the current smartwatch apps are designed, while criterion (ii) and (iii) helps as an initial screening to identify apps that might have privacy and security issues. For instance, a smartwatch app that either communicates with the smartphone, which in turn communicate with the internet or a smartwatch app that can directly connect to the internet has the potential of leaking data collected from the smartwatch. The overall methodology adopted to create taxonomy is outlined in Figure 4 and the detailed process followed for each platform is described below.

Android Wear: Android Wear apps can be developed in two ways: (i) Apps only sending notifications from smartphone to smartwatch and has only ***handheld*** APK, (ii) Apps having both ***handheld*** and wearable APKs. To check if a given app has only a ***handheld*** APK or consists of ***handheld*** and ***wearable*** APK, we look for the presence of wearable APK inside the *res* directory of the decoded handheld APK.

Next, we identified the apps that make communication between handheld component on smartphone and a wearable component on smartwatch to perform their functions. The identification of such apps is done by inspecting the decompiled source code for certain classes, permissions, and methods as discussed:

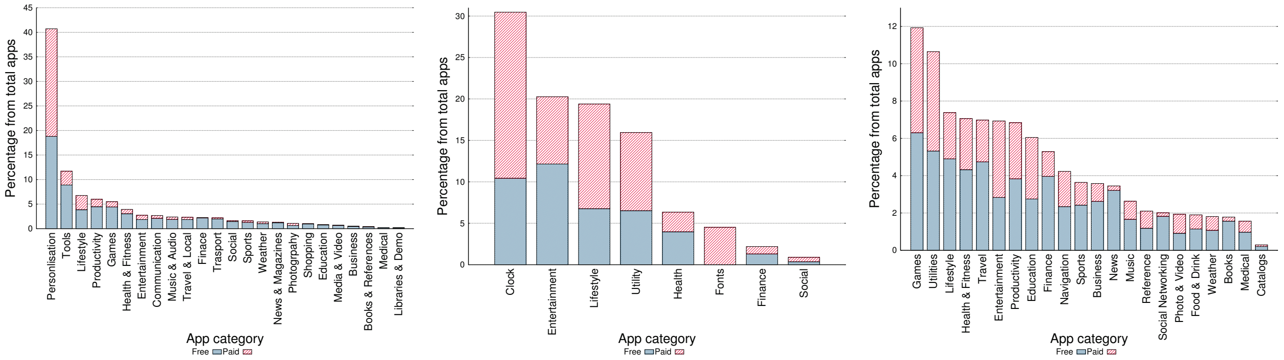
(i) Class names containing Wearable Data Layer APIs: Wearable Data Layer API [22] interfaces between the wearable and the handheld component. The class names we looked for are: *DataApi, MessageApi, ChannelApi* and *NodeListener*.

(ii) Wearable.BIND_LISTENER permission: This permission indicates that the app listens for the events delivered as a result of changes through the Wearable Data Layer APIs.

(iii) Invocation of notification related methods: Methods such as *NotificationCompat, WearableExtender,* and *RemoteView* in the app indicate notifications are being exchanged [23].

Finally, we checked for the app's ability to connect to internet by searching for the INTERNET permission in the AndroidManifest.xml file of the app.

Samsung: Samsung provides two methods to develop apps: ***standalone*** and ***companion***. Standalone apps are Tizen based web apps called *widgets* written using HTML5, JavaScript, and CSS, and execute independently on the smartwatch. Companion apps have two parts: a handheld component (Android APK) running on the smartphone and a wearable component



(a) Android Wear

(b) Samsung

(c) Apple

Fig. 3: Category-wise app distribution

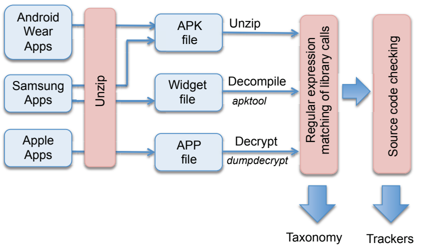


Fig. 4: Process of app analysis

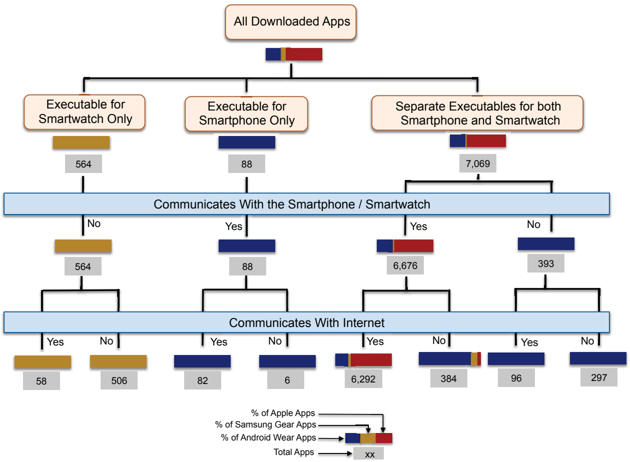

All Downloaded Apps
Executable for Smartwatch Only
Executable for Smartphone Only
Separate Executables for both Smartphone and Smartwatch
564
88
7,069
Communicates With the Smartphone / Smartwatch
No
Yes
Yes
No
564
88
6,676
393
Communicates With Internet
Yes
No
Yes
No
Yes
No
Yes
No
58
506
82
6
6,292
384
96
297
% of Apple Apps
% of Samsung Gear Apps
% of Android Wear Apps
Total Apps
xx

Fig. 5: Taxonomy of smartwatch apps

(Tizen widget) running on the smartwatch. For each app, we checked for the type of the file extension of the app. If the app extension ends with .wgt, then it is classified as a standalone app, otherwise it is classified as a companion app. We used *apktool* to decompress and dissemble the APK part. Widget is decompressed using *unzip*. To establish connections or share data between the wearable and the handheld component in a companion app, the developer has to use Samsung Accessory APIs [24]. The APIs which can be used by the developer are: *sendData, sendSecureData, sendFile, receiveFile, onreceive, setDataReceiveListener, and setFileReceiveListener*. We searched for these API calls in each app's code.

Finally, to check whether apps connect to the internet, we searched for INTERNET permission in the manifest file of the handheld component of the companion apps. Tizen has a similar permission named, "http://tizen.org/privilege/internet" in the config.xml file of standalone apps. config.xml in Tizen is similar to the manifest file in Android where all the permissions needed by the app are declared.

Apple: Apple Watch apps are designed using WatchKit [25] and has two parts: *WatchKit app* (that runs on Apple Watch) and *WatchKit extension* (that runs on the user's iPhone). The WatchKit app contains only the storyboards and resource files associated with the app's user interface. The WatchKit extension contains the code for managing the WatchKit app's user interface and for responding to user interactions. The WatchKit app and the WatchKit extension are packaged inside an iOS app. We first used unzip to unpack the iOS app and then used *dumpdecrypted* [26] to decrypt the iOS binary.

To communicate between the iOS app and the WatchKit extension, developers can use two methods [27]: (i) Sending a request using *openParentApplication* from the WatchKit extension to iOS app, which in turn handles the request using *handleWatchKitExtensionRequest* function, (ii) *initWithSuite* function that lets WatchKit extension and iOS app to save, read and share data using the *NSUserDefaults* API. We searched for *handleWatchKitExtensionRequest* and *initWithSuite* with *NSUserDefaults* in the app binary to decide if there exists an interaction between the WatchKit extension and iOS app.

Apple apps do not have a specific permission for accessing the internet. Hence, we rely on searching for the function names and classes that are used to access internet in iOS apps in the iOS app binary. Examples of functions and classes we searched for are: *NSURLConnection, NSURL, NSURLRequest, CFHTTPStream, NSURLDownload, openURL, NKAssetDownload, and loadRequest* [28].

The results of taxonomy based classification is shown in Figure 5. As, Samsung Gear S supports WiFi and 3G, we can see that more developers are developing standalone apps. 90% of all free apps in Samsung are widgets. However, to our surprise only 10% of the standalone apps use internet, despite Samsung smartwatch having independent WiFi and 3G unit. Contrary to Samsung, in Android Wear 96% of the apps have two separate components, handheld on smartphone and wearable on smartwatch. In Apple, all the apps have handheld and wearable component and the two components communicate with each other in all the apps.

V. Privacy and Security

In this section, we first explore the ***third party tracking*** in smartwatch apps through static analysis. Then, we analyze the network traces generated by executing a sample of apps and identify associated ***personal information leakages***.

A. Trackers

Third party advertising and analytics companies known as *trackers* collect personal information such as device identifiers, location and user credentials from smartphone apps. When it comes to smartwatches such information collection might become more critical as smartwatches can contain more sensitive user information such as health information. Thus, we investigated the tracker connectivity in smartwatch apps. The overall methodology is illustrated in Figure 4 in Section IV-B and platform specific details are described below.

Android Wear: We inspect the decompiled code for the presence of 124 trackers identified in our previous work [29].

Samsung: We followed the same approach as Android Wear for Samsung companion apps. To identify, the trackers in the wearable components (widgets), we searched for the names of the trackers as strings among the files in each widget directory.

Apple: Integrating trackers to iOS apps requires importing specific header files in the app code. For example, integrating Flurry analytics needs *Flurry.h* to be imported inside the code. We searched for the presence of such header files for each tracker from the tracker list in the decrypted iOS app binary.

TABLE II: Number of apps connected to trackers

	Number of connected trackers			
	0	**1**	**2**	**>=3**
	Handheld component			
Android W.	1,042 (53%)	898 (47%)	397 (21%)	167 (9%)
Samsung	58 (64%)	34 (36%)	13 (14%)	6 (7%)
Apple	1,936 (38%)	3,189 (62%)	2,224 (43%)	1,320 (25%)
	Wearable component			
Android W.	1,807 (93%)	133 (7%)	30 (1.5%)	25 (1.3%)
Samsung	654 (99.7%)	2 (0.3%)	0 (0%)	0 (0%)
Apple	4,385 (86%)	740 (14%)	426 (8%)	72 (1.4%)

Table II shows the results. As can be seen from wearable components, Apple had the highest percentage (14%) of apps that are connected to at least one tracker. The lesser number of trackers in Samsung wearable components can be attributed to the lack of support of popular tracker SDKs for Tizen platform. The trackers that are present in Samsung ecosystem are the *javascript* based trackers that do not require additional modifications for Tizen.

Figure 6 shows frequently observed trackers in the handheld component. The top three trackers across all ecosystems are Google Analytics, Crashlytics, and Flurry. Among Apple smartwatch app developers MoPub is the popular ad platform choice rather than Google Ads.

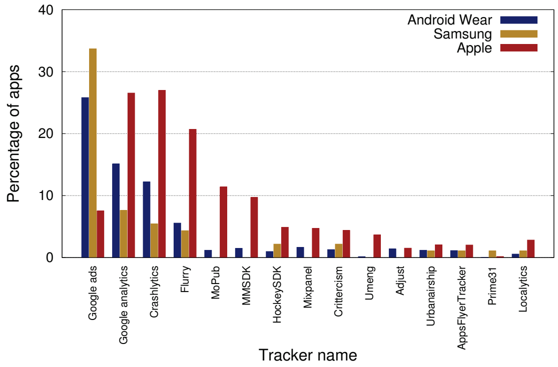


Fig. 6: Top Trackers - Handheld component

B. Personal Information Leakage

To investigate what types of personal information are being collected by smartwatch apps, we executed a sample of apps either ***automatically*** or ***manually*** and collected the network traffic generated by those apps using MITM proxy [30]. MITM proxy allows decrypting of HTTPS traffic in addition to sniffing the traffic. We only pick the apps which can connect to the internet and whose handheld and wearable component interacts with each other as was shown in Figure 5.

In ***automated*** testing, we executed 1,183 Android Wear apps. The procedure followed was to install the app using *adb shell*, start the MITM proxy, start the app, perform 5000 actions on the app on smartwatch using Monkey [31], uninstall the app, stop the MITM proxy and save the network flows.

Due to the lack of automated execution library, we manually tested apps for Samsung and Apple. For each app, we install it, enter user credentials on the smartwatch or smartphone if required, interact with it for 5 minutes on the smartwatch, uninstall it and save the network flows. In Samsung, as there are only 129 apps that communicate with the internet, we tested all apps. However, since there are 5,125 apps in Apple, we tested a 10% random sample of apps, which contains 10% of the apps from each category. For comparison, we also manually checked a similar 10% sample from Android Wear.

We inspected the traffic generated by each app to check whether it contains personal information by parsing it through a Python script. The personal information we focused in this experiment are: *i) Unique identifiers* (E.g. IMEI, Android ID, and WiFi MAC Address), *ii) Location, iii) Credentials* (E.g. user names and passwords of stored accounts), *iv) Health data* such as heart rate, calorie intake, and water intake and *v) User activities* such as opening an app, changing watch faces etc. The devices used in the experiments are: LG G Watch R paired with Nexus 5, Samsung Gear S paired with a Galaxy S IV and Apple Watch paired with iPhone 5C.

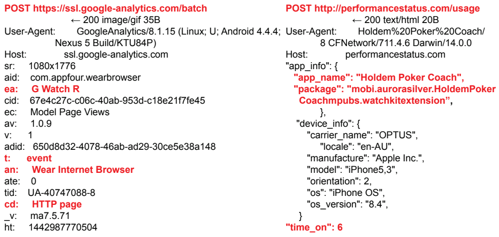
The results are shown in Table III. 5.2% apps in Android Wear and 1.5% apps in Samsung are collecting unique

TABLE III: Summary of the Apps leaking Private Information

	Android Wear (Manual)	Android Wear (Auto)	Samsung (Manual)	Apple (Manual)
No. of apps	118	1,183	129	512
Unique IDs	6	51	2	0
Location	2	4	0	6
Credentials	3	14	0	1
Activity	6	65	6	57

identifiers. However, these identifiers are associated with the smartphone than the smartwatch. We found out that across all ecosystems, the device ids or MAC addresses unique to the smartwatches are never leaked. 12 apps collected user location and 18 apps sent user's personal email ids to the third parties.

Interestingly, we observed that 6% of tested apps in Android Wear, 4% in Samsung and 11% in Apple apps send smartwatch specific user activities to third party trackers. Figure 7 shows two examples. We found Google Analytics to be the most used third party tracker (85%) on both Samsung and Android Wear for sending watch activities. For Apple Watch, MoPub is present in 65% of the cases.



```
POST https://ssl.google-analytics.com/batch
                    ← 200 image/gif 35B
User-Agent:       GoogleAnalytics/8.1.15 (Linux; U; Android 4.4.4;
              Nexus 5 Build/KTU84P)
Host:            ssl.google-analytics.com
sr:    1080x1776
aid:   com.appfour.wearbrowser
ea:    G Watch R
cid:   67e4c27c-c06c-40ab-953d-c18e21f7fe45
ec:    Model Page Views
av:    1.0.9
v:     1
adid:  650d8d32-4078-46ab-ad29-30ce5e38a148
t:     event
an:    Wear Internet Browser
ate:   0
tid:   UA-40747088-8
cd:    HTTP page
_v:    ma7.5.71
ht:    1442987770504
```

(a) Android Wear

```
POST http://performancestatus.com/usage
                    ← 200 text/html 20B
User-Agent:       Holdem%20Poker%20Coach/
            8 CFNetwork/711.4.6 Darwin/14.0.0
Host:            performancestatus.com
"app_info": {
  "app_name": "Holdem Poker Coach",
  "package": "mobi.aurorasilver.HoldemPoker
   Coachmpubs.watchkitextension",
        },
   "device_info": {
     "carrier_name": "OPTUS",
         "locale": "en-AU",
     "manufacture": "Apple Inc.",
     "model": "iPhone5,3",
     "orientation": 2,
     "os": "iPhone OS",
     "os_version": "8.4",
   }
"time_on": 6
```

(b) Apple

Fig. 7: Examples of activity tracking

VI. Discussion & Future Work

In a first study of its kind, we studied more than 14,000 smartwatch apps across three popular app markets: Android Wear, Samsung and Apple, specifically comparing the number of apps, prices, number of developers and categories. Our results showed that Apple who was last to enter the market has obtained a significant growth and has more apps compared to Android Wear and Samsung. On the other hand, we also showed that Apple smartwatch apps are slightly expensive than Android Wear and Samsung. By doing cross-market analysis of apps, we found that banking and transit related apps have quickly embraced smartwatches. Our results also showed that even at this early stage there is some tracking happening in smartwatch apps. We found that 14%, 7%, and 0.3% apps from Apple, Android Wear, and Samsung ecosystems respectively, had third party trackers in the wearable component of the apps. While these numbers are low compared to smartphones, when smartwatches become further ubiquitous, more tracker companies are likely to be attracted due to the volumes of rich user data smartwatches can provide.

As of now smartwatch ecosystem is a highly dynamic space with new software updates and devices being released continuously. In such a changing landscape, it will be interesting to study the spatial and temporal evolution of smartwatch ecosystems along the dimensions of app market growth, the influence of external events such as new software and hardware updates on the market growth and user adoption of apps. To this end, we plan to observe smartwatch app ecosystems periodically and collect snapshots. We are also planning to gain further insights into app functionalities by mining the metadata of the smartwatch apps.

References

[1] N. Mawston, "Apple watch captures 75 percent global smartwatch marketshare in q2 2015," https://www.strategyanalytics.com, 2015.

[2] P. Lamkin, "101 million smartwatch shipments by 2020 with apple and google leading the way," http://www.wareable.com, 2015.

[3] Statista Inc., "Apple is already dominating the smartwatch market," http://www.statista.com/chart/3674/smart-watch-market-in-q2-2015/.

[4] PR Newswire, "Strategy analytics: Android wear falls to 11 percent global smartwatch os marketshare in q2 2015," http://www.prnewswire.com/news-releases.

[5] R. Wong, "Apple watch now has more than 10,000 apps," http://mashable.com/2015/09/09/apple-watch-os-2/, 2015.

[6] T. Haselton, "Android wear now has more than 4,000 apps," http://www.technobuffalo.com/2015/05/28/android-wear-now-has-more-than-4000-apps/, 2015.

[7] A. Kharpal, "Samsung: Our smartwatch has over 1,000 apps," http://www.cnbc.com.

[8] 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 *Proc. of OSDI*, 2010.

[9] M. Egele, C. Kruegel, E. Kirda, and G. Vigna, "PiOS: Detecting privacy leaks in iOS applications," in *Proceedings of the NDSS*, 2011.

[10] N. d'Heureuse, F. Huici, M. Arumaithurai, M. Ahmed, K. Papagiannaki, and S. Niccolini, "What's app?: A wide-scale measurement study of smart phone markets," *MCCR*, vol. 16, no. 2, pp. 16–27, 2012.

[11] N. Viennot, E. Garcia, and J. Nieh, "A measurement study of Google Play," in *Proc. of the SIGMETRICS*. ACM, 2014.

[12] M. C. Grace, W. Zhou, X. Jiang, and A.-R. Sadeghi, "Unsafe exposure analysis of mobile in-app advertisements," in *Proc. of WiSec*, 2012.

[13] I. Leontiadis, C. Efstratiou, M. Picone, and C. Mascolo, "Don't kill my ads! balancing privacy in an ad-supported mobile application market," in *Proc. of HotMobile*, 2012.

[14] HP Fortify, "Internet of things security study: Smartwatches," http://go.saas.hpe.com/fod/internet-of-things, 2015.

[15] H. Wang, T. T.-T. Lai, and R. Roy Choudhury, "Mole: Motion leaks through smartwatch sensors," in *Proc. of MobiCom*, 2015, pp. 155–166.

[16] "Android Wear Center,," http://www.androidwearcenter.com.

[17] "Goko," http://goko.me.

[18] "Watchaware," http://watchaware.com.

[19] AppBrain Inc., "Distribution of free vs. paid Android apps," http://www.appbrain.com/stats/free-and-paid-android-applications, 2015.

[20] R. Triggs, "The history of app pricing, and why most apps are free," http://www.androidauthority.com/history-of-app-pricing-245832/, 2013.

[21] C. Jones, "Apple's app store about to hit 1 million apps," http://www.forbes.com, 2013.

[22] Google Developers, "Google apis for android," https://developers.google.com/android/reference/com/google/android/gms /wearable/package-summary.

[23] Android Developers, "Creating a notification for wearables," https://developer.android.com/training/wearables/notifications/creating.html.

[24] Samsung, "SAP," http://developer.samsung.com/technical-doc/view.do?v=T000000188.

[25] Apple, "Developing for apple watch," https://developer.apple.com, 2015.

[26] S. Esser, https://github.com/stefanesser/dumpdecrypted.

[27] C. Herbert, "Getting data to your watchkit app," http://blog.curtisherbert.com/data-synchronization-with-watchkit.

[28] Apple, "Networking overview," https://developer.apple.com/library/ios/ documentation/NetworkingInternetWeb/Conceptual/NetworkingOverview.

[29] S. Seneviratne, H. Kolamunna, and A. Seneviratne, "A measurement study of tracking in paid mobile applications," in *ACM WiSec*, 2015.

[30] "Mitmproxy," https://mitmproxy.org/.

[31] "UI monkey," http://developer.android.com/tools/help/monkey.html.