ANDRORISK: A TOOL FOR PRIVACY RISK ASSESSMENT

Touret, Christophe Ripley, Gareth

Software Engineering Department, Embry-Riddle Aeronautical University
Daytona Beach, FL 32114
SE590: Graduate Seminar Research Paper
Email: {touretc,ripleyg}@my.erau.edu

Abstract – The Android operating system does not provide users with a way to determine the user's risk of being exposed to privacy threats. When an application is installed, a list of permissions the application requires to access sensitive data is provided with no supporting details describing the context in which the data is used. Furthermore, as applications may interact, sensitive data be used by a restricted application through other applications on the phone. Using the open source custom firmware TaintDroid developed by a collaborative group of researchers from Intel Labs, Penn State, and Duke University, we can monitor in Real-Time the communication of sensitive data by installed third-party applications. Considering the frequency of these communications and an impact evaluation that we suggest, we are able to make a per-user assessment of the risk involved with using the device in its current state with the third-party applications that are installed. In this paper, we present AndroRisk, an application to run on TaintDroid OS that analyzes the handling of sensitive data by applications to better identify the privacy risk in which a user is exposed.

Keywords: Android, Personal Data, Privacy, Risk, Permission.

1. Introduction

Smartphones have become a necessity in the modern world. Their unmatched utility as an everyday device has made them a central hub of user sensitive data. A key feature of modern smartphone platforms is a centralized service for downloading third-party applications [1]. For Android, the Google Play Store as well as other app stores like Amazon Appstore brings over a million third-party applications to the hands of every user. During the 2013 Google I/O conference, it was announced that there have been 50 billion application downloads from their store alone and they expected to see 100 billion by the end of the year based on the insane growth rate of Android device activations they are currently seeing (about 1.5 million devices every day) [1].

Many of these application use user stored data as well as information from device sensors (GPS, camera, microphone, etc) to bring functionality to the stock Android phone. To access this sensitive data, applications ask for the user's permission upon installation. Although this lets users know exactly what information an application will have access to, it does not let the user know the context in which it is used. There are applications that use sensitive data for more than what a user might initially intend. For example, an application that accesses an individual's GPS location for tracing their steps when going on a trip might also be sending that information to an advertisement agency.

There are detection tools that have been developed that have tried to identify when a third-party application is using sensitive information for reasons unbeknownst to the user. There have also been risk assessment methodologies established that can be used to assess the risk level of individual applications installed on a user's device. However, the existing tools are per application focused, while privacy issues can also be resulting from apps interactions. Indeed, the namely transivity-of-thrust problem stated that because applications may interact, sensitive data be used by a restricted application through other applications on the phone. [1] Hence, in order to take into consideration this phenomena, we attempted to provide a per user risk assessment tool.

Using the previously completed work, we have tried to create a proof of concept for unifying the threat detection and assessment methodologies and calculate the overall threat level of an Android device in its current state. In this report, we discuss TaintDroid and use it as the platform from which we develop out Android user risk assessment application.

2. Method

The following subsections discusses the use of existing tools and methodologies and the work completed to adjust the risk assessment to suit our needs and work with the TaintDroid.

2.1 TaintDroid

The pre-existing open source tool TaintDroid was developed to monitor the flow of user sensitive data used by third-party applications installed on a mobile Android device in real-time and alert the user when their sensitive data has been sent out over the network to an external target. Detection is accomplished using what is known as dynamic taint analysis (or taint tracking). This involves identifying the sensitive information that is to be monitored and "tainting" the information at its source so it can be tracked through the system whenever it is accessed, communicated, or manipulated. As seen in Figure 1, TaintDroid tracks the user-sensitive information as it moves through the system at four defined levels.

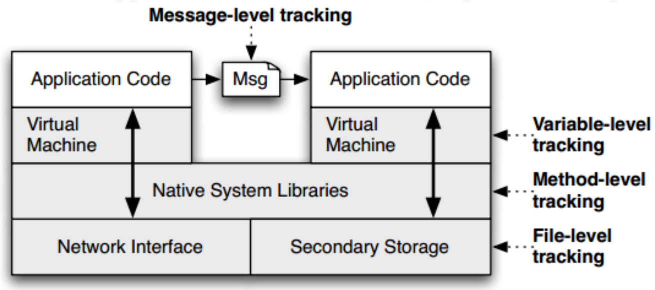


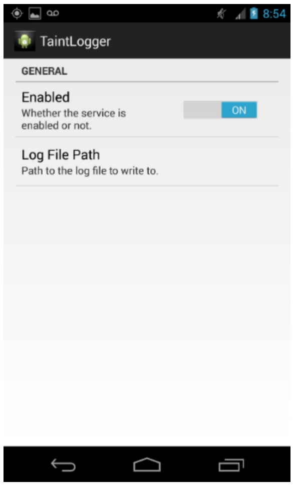
FIGURE 1: TaintDroid's Four Levels of Tracking [3]

The first level is tracking sensitive information at the variable-level within the third-party application code. The second is tracking at the message-level which monitors the transfer of sensitive information between two or more installed applications. The third level is tracking at the Android OS method-level using its native libraries. Lastly, the fourth level of tracking is at the file-level were TaintDroid makes sure that the sensitive information being tracked remains tainted so that it can continue to be tracked when accessed again [3].

2.1.1 TaintDroid TaintLogger Application

Being able to identify when third-party applications are communicating user sensitive data is a big step to classifying which applications may be posing privacy risks or even using the user-given permissions to access their information for reasons other than what the user initially intended. The missing piece is analyzing what TaintDroid has detected with a standardized risk assessment methodology. In order to use the TaintDroid tool for this kind of assessment, extra functionality was required to keep a log of all detected events. Currently, the logs are built and can be viewed using logcat Android viewer or their built-in notification application but a list of all detected occurrences is not saved to do any sort of risk assessment over time with all detected occurrences. The construction of this log file was developed to allow the communication of all detected possible threats to the application AndroRisk that would quantitatively estimate the overall risk of the Android device in its current state.

To use the developed TaintLogger application, the SuperUser application must first be installed on the rooted TaintDroid Device so that the application has the proper root permissions to write to the created text file. After installation of TaintLogger, the logging is enabled by toggling the Enabled switch to On.


8:54

TaintLogger

GENERAL

Enabled
Whether the service is enabled or not.
ON

Log File Path
Path to the log file to write to.

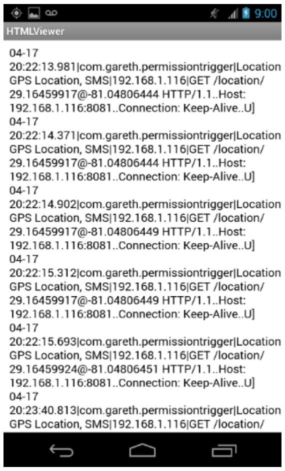
FIGURE 2: TaintLogger Application

Each time the Enabled switch is toggled to On, the log file is cleared and the logger will begin logging each TaintDroid logcat message. The logger can continue to run for almost 25 consecutive days before it stops monitoring the logcat.

2.1.2 Log File Construction

Whenever a threat is detected by TaintDroid, an entry will be added to the log file. Each entry will follow the following format:

Timestamp|ApplicationName|InfoAccessed|
DestinationIP|Data



```
04-17
20:22:13.981|com.gareth.permissiontrigger|Location
GPS Location, SMS|192.168.1.116|GET /location/
29.16459917@-81.04806444 HTTP/1.1..Host:
192.168.1.116:8081..Connection: Keep-Alive..U]
04-17
20:22:14.371|com.gareth.permissiontrigger|Location
GPS Location, SMS|192.168.1.116|GET /location/
29.16459917@-81.04806444 HTTP/1.1..Host:
192.168.1.116:8081..Connection: Keep-Alive..U]
04-17
20:22:14.902|com.gareth.permissiontrigger|Location
GPS Location, SMS|192.168.1.116|GET /location/
29.16459917@-81.04806449 HTTP/1.1..Host:
192.168.1.116:8081..Connection: Keep-Alive..U]
04-17
20:22:15.312|com.gareth.permissiontrigger|Location
GPS Location, SMS|192.168.1.116|GET /location/
29.16459917@-81.04806449 HTTP/1.1..Host:
192.168.1.116:8081..Connection: Keep-Alive..U]
04-17
20:22:15.693|com.gareth.permissiontrigger|Location
GPS Location, SMS|192.168.1.116|GET /location/
29.16459924@-81.04806451 HTTP/1.1..Host:
192.168.1.116:8081..Connection: Keep-Alive..U]
04-17
20:23:40.813|com.gareth.permissiontrigger|Location
GPS Location, SMS|192.168.1.116|GET /location/
```

FIGURE 3: TaintLogger Log.txt File

Timestamp: Month-Day Hr:Min:Sec.Millisec
Hour will be in 24 hour military time (24 hour clock).

Application name: Listed names correspond to the name displayed on the phone and not necessarily the name listed in the Android Market. This is determined by the process ID that is tied to the threat detected.

InfoAccessed: Tainted data identified via the tag attached to the data. The following is defined in Taint.h header file:

```
#define TAINT_CLEAR                    ((u4)0x00000000)
#define TAINT_LOCATION                 ((u4)0x00000001)
#define TAINT_CONTACTS                 ((u4)0x00000002)
#define TAINT_MIC                      ((u4)0x00000004)
#define TAINT_PHONE_NUMBER             ((u4)0x00000008)
#define TAINT_LOCATION_GPS             ((u4)0x00000010)
#define TAINT_LOCATION_NET             ((u4)0x00000020)
#define TAINT_LOCATION_LAST            ((u4)0x00000040)
#define TAINT_CAMERA                   ((u4)0x00000080)
#define TAINT_ACCELEROMETER            ((u4)0x00000100)
#define TAINT_SMS                      ((u4)0x00000200)
#define TAINT_IMEI                     ((u4)0x00000400)
#define TAINT_IMSI                     ((u4)0x00000800)
#define TAINT_ICCID                    ((u4)0x00001000)
#define TAINT_DEVICE_SN                ((u4)0x00002000)
#define TAINT_ACCOUNT                  ((u4)0x00004000)
#define TAINT_HISTORY                  ((u4)0x00008000)
```

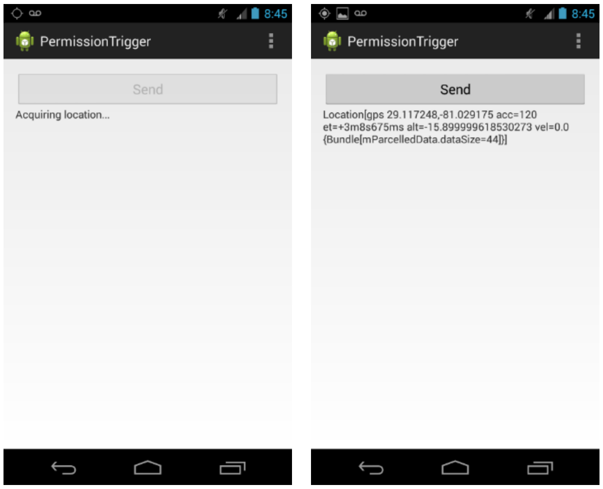
This allows TaintDroid to identify if multiple types of sensitive data are detected together. For example, if GPS location and IMEI (Phone ID) are both detected, the data tag would be read as 0x410.

DestinationIP: The IP address where the user's sensitive data was sent to.

Data: The complete Android logcat entry constructed by TaintDroid tool that has been read by the TaintLogger application.

2.2 PermissionTrigger Application

The developed permission trigger application is not a required application for the threat detection or the risk assessment. This application is strictly for the testing purposes of the TaintLogger and AndroRisk application. This simple application was developed to access the user's sensitive data and send the acquired data to an external server to trigger a TaintDroid logcat message to be fabricated. Being able to trigger our own TaintDroid detections made it much easier to test the developed applications during development and could be used to test the implemented risk assessment methodology.



Acquiring location...

Location[gps 29.117248,-81.029175 acc=120 et=+3m8s675ms alt=-15.899999618530273 vel=0.0 {Bundle[mParcelledData.dataSize=44]}]

FIGURE 4: PermissionTrigger Application

The application currently only handles the acquiring and sending of user's GPS location but the application can be extended to handle all other sensitive data. The application works in unison with the node.js code test_server.js which should be ran on a computer with internet access to receive the sensitive data sent by the PermissionTrigger application of the device. In the PermissionTrigger source code, the server IP needs to be edited to match the IP of the computer running test_server.js.

2.3 Quantitative Privacy Risk Assessment

Quantitative methods for risk assessment have been widely used in the financial field, and they are generally defined as the product between the likelihood and the impact of the risk. With the logcat file produced from the TaintLogger application, we are able to assess different privacy risks for a user using the same logic.

2.3.1 Threats

In our case, we are interested in assessing the risk level for each privacy threat defined in Table 1. [4]

Threat	Description
Tracking/Surveillance (T1)	Monitoring of user's location. User's location can be retrieved directly using user GPS (fine location) or indirectly using the smartphone's sensors, network or the user's calendar.
Interception/Eavesdropping (T2)	Unlawful interception of communications and is applicable to all communication.
Profiling (T3)	Monitoring of user's activities for advertising purposes.
Phishing (T4)	Disclosure of user's credential by tricking.
Personal Information Disclosure (T5)	Disclosure of all other types of personal information, which do not fall in the other four threat types.

TABLE 1: Privacy Threats

For possible threat to occur, specific sensitive data needs to be retrieved from the user's Android device. The log file generated from TaintLogger provides us with the sensitive data accessed which can be correlated with the permission required by the system to allow the third-party application access.

Hence, a study defined a mapping of sensitive data, permissions and privacy threats which is presented in table 2. [4]

Data category	Sensitive Data Type	Permission	Privacy Threats				
			T1	T2	T3	T4	T5
Comm	SMS	RECEIVE_SMS		✓			
		READ_SMS		✓			
	MMS	RECEIVE_MMS		✓			
	Voice	PROCESS_OUTGOING_CALLS		✓			
	Wap	RECEIVE_WAP_PUSH		✓			
Sensor / Loc	Push Video	CAMERA	✓	✓			
	Audio	RECORD_AUDIO	✓	✓			
	Loc	ACCESS_COARSE_LOCATION	✓		✓		
		ACCESS_FINE_LOCATION	✓		✓		
		BLUETOOTH_ADMIN	✓		✓		
		ACCESS_NETWORK_STATE	✓		✓		
		ACCESS_WIFI_STATE	✓		✓		
		READ_SOCIAL_STREAM	✓		✓		
External Storage		WRITE_EXTERNAL_STORAGE		✓	✓		✓
		READ_EXTERNAL_STORAGE		✓	✓		✓
Contacts		READ_CONTACTS		✓			✓
		READ_SOCIAL_STREAM		✓			✓
History/ Usage		READ_CALL_LOG		✓	✓		
		READ_HISTORY_BOOKMARKS			✓		
		GET_TASKS			✓		
		READ_LOGS			✓		
		READ_USER_DICTIONARY			✓		✓
		READ_SOCIAL_STREAM			✓		
		SUBSCRIBED_FEEDS_READ			✓		
Calendar		READ_CALENDAR	✓				✓
Credentials		READ_SMS - RECEIVE_SMS				✓	
		AUTHENTICATE_ACCOUNTS				✓	
		USE_CREDENTIALS				✓	

TABLE 2: Mapping of Sensitive data, Permissions, and Privacy Threats

With the mapping on table 2, we were able to affect the collected logs from TaintLogger to corresponding threats. For each threat, by analyzing the logs, i.e. the permissions' accesses, we were able to assess a risk.

2.3.2 Risk Assessment Method

Our method consists of calculating a quantitative risk value for each threat by doing the product of impact and likelihood values retrieved from an analysis of the threat's permissions' accesses list. By using the smartphone for a certain period of time with TaintLogger, a user will accumulate a list of logs that are corresponding to the user permissions' accesses. During that period, the user should use the phone as they would normally, and for at least one complete day. The longer the period of use, the more accurate the risk assessment is. AndroRisk tool will then extract from the log file the five threats' permissions' accesses list. From each list the application calculate an impact and a likelihood value, and the threat risk is calculated from their product.

Then, the user can enter into AndroRisk to see each threat's risk as well as view the detailed list of permissions' accesses so the user can understand where the threat comes from. An overall view of the AndroRisk functioning is shown in Figure 5.

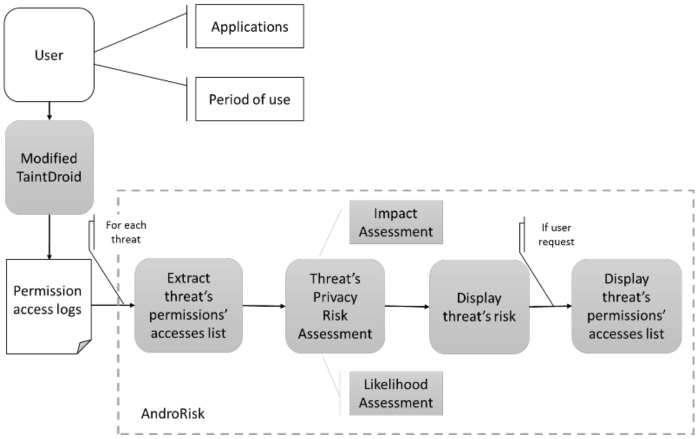


FIGURE 5: AndroRisk Functioning Scheme

Impact

For assessing a threat's overall impact, we are calculating the average impact value from the threat's permissions that have been accessed. An individual impact value of a permission's access should take into consideration three different source of inputs: the identifying level of the sensitive data, the user defined severity level, [4] and an indicator representing the chance that the data is accessed for a malicious purpose.

The identifying level is addressed using the table 3, which assigns for each permissions a value from 1 to 4.

A questionnaire is used for assigning the user severity level for the permissions. From that questionnaire, a value from 0 to 4 is assigned to the accessed information provided by the TaintLogger application. As an indicator representing the chance that the data is accessed for a malicious purpose, we consider that a sensitive data access being done by background application is more likely to be malicious than an active application.

ID	Identification Impact	Permission
1	**Negligible:** Identifying a user using this permission appears to be virtually impossible.	ACCESS_COARSE_LOCATION ACCESS_FINE_LOCATION ACCESS_NETWORK_STATE ACCESS_WIFI_STATE BLUETOOTH_ADMIN GET_TASKS READ_CALENDAR READ_HISTORY_BOOKMARKS READ_LOGS READ_USER_DICTIONARY RECEIVE_WAP_PUSH SUBSCRIBED_FEEDS_READ
2	**Limited:** Identifying a user using this permission appears to be difficult but is possible in certain cases.	CAMERA PROCESS_OUTGOING_CALLS READ_CALL_LOG READ_CONTACTS READ_EXTERNAL_STORAGE READ_SMS READ_SOCIAL_STREAM RECEIVE_MMS RECEIVE_SMS RECORD_AUDIO WRITE_EXTERNAL_STORAGE
3	**Significant:** Identifying a user using this permission appears to be relatively easy.	AUTHENTICATE_ACCOUNTS GET_ACCOUNTS USE_CREDENTIALS
4	**Maximum:** Identifying a user using this permission appears to be extremely easy.	READ_PHONE_STATE READ_PROFILE

TABLE 3: Permission's Identification Value

A permission impact value is calculated considering the following formula:

$$I(A) = Id(A) + U(A) + B(A)$$

where I(A) is the impact value of the permission access A, Id(A) its identification value, U(A) the user's impact value, and B(A) the background value.

The background value is calculated considering the following function:

$$B: \begin{cases} A \rightarrow 2 \; if \; done \; in \; background, \\ A \rightarrow 0 \; if \; done \; in \; front. \end{cases}$$

At the end, the impact value can vary from 1 to 10.

Likelihood

Many studies showed that the more a sensitive data is accessed, the more the risk is likely to occur [1]. Thus, the likelihood of a privacy threat should increase in function of the number of sensitive data accesses for each threat's list. However, the number of sensitive data access can be large and a range of value from 1 to 10 was necessary. Hence, we decided to use the following function for assessing the likelihood:

$$L: \quad \begin{cases} T \rightarrow 1 \; if \; N(T) \leq 2, \\ T \rightarrow \lceil \log_{1.7}(N(T)) \rceil \; if \; 2 < N(T) \; \leq 200, \\ T \rightarrow 10 \; if \; N(T) > 200, \end{cases}$$

where T is a threat and N(T) is the number of T's permissions' accesses per day.

We are using as a metric the number of access per day in order to keep the likelihood independent from the period of use. Indeed, because the usage of a smartphone should not be significantly different one day from another, we valued that a per day basis would make a good metric.

Risk

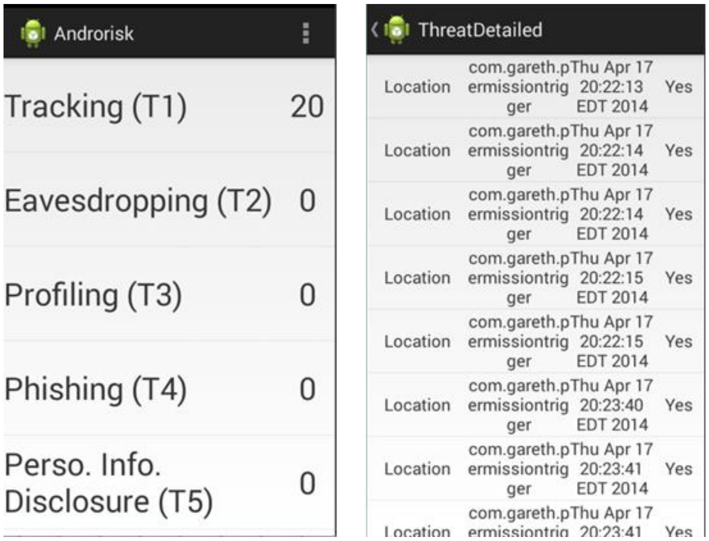
A threat's risk is calculated as the product of its impact and likelihood values:

$$R(T) = L(T) * I(T)$$

where R(T) is the risk of the threat T for the user. Because L(T) and I(T) has both a range of values from 1 to 10, the range of risk value is from 1 to 100.

3. Results and Conclusions

We implemented a proof of concept for AndroRisk, a quantitative privacy risk assessment tool. By using the application over a certain period of time as usual, a user can identify the risk level of privacy threats he is exposed to. There are five privacy threats to be considered: Tracking (T1), Eavesdropping (T2), Profiling (T3), Phishing (T4), and Personal Information Disclosure (T5) [4]. For each of them, a risk value from 1 to 100 is calculated. The user can view the detailed list of permissions' accesses specific to a threat from the application.



Androrisk

Tracking (T1)	20
Eavesdropping (T2)	0
Profiling (T3)	0
Phishing (T4)	0
Perso. Info. Disclosure (T5)	0

ThreatDetailed

Location	com.gareth.permissiontrigger	Thu Apr 17 20:22:13 EDT 2014	Yes
Location	com.gareth.permissiontrigger	Thu Apr 17 20:22:14 EDT 2014	Yes
Location	com.gareth.permissiontrigger	Thu Apr 17 20:22:14 EDT 2014	Yes
Location	com.gareth.permissiontrigger	Thu Apr 17 20:22:15 EDT 2014	Yes
Location	com.gareth.permissiontrigger	Thu Apr 17 20:22:15 EDT 2014	Yes
Location	com.gareth.permissiontrigger	Thu Apr 17 20:23:40 EDT 2014	Yes
Location	com.gareth.permissiontrigger	Thu Apr 17 20:23:41 EDT 2014	Yes
Location	com.gareth.permissiontrig	Thu Apr 17 20:23:41	Yes

FIGURE 6: AndroRisk Application

Our risk assessment method consist of multiplying an impact value and a likelihood value for each threat. The impact value take into account three components: the identification level of the permissions, the user's severity level and if the task was ran in background or not. The likelihood is taking consideration the number of permission's accesses per day. As shown in figure 6, AndroRisk displays to the user the privacy risk for each privacy threat, and, if requested, the detailed logs corresponding to a threat. Our goal was uniting a risk assessment methodology with a pre-existing sensitive data access/communication detection tool for assessing privacy risks on an Android Phone from a per user perspective. Our work can act as a platform for others to continue toward this goal.

4. Future Work

The following subsections discuss what work can be further explored or developed that this study was not able to complete.

4.1 TaintDroid Tools

The TaintLogger application is functionally complete but it can be extended to provide more information to the AndroRisk application that could be used by the risk assessment methodology. One of these that was explored but not completed was Active/Background application detection.

Another functional addition would be have to be developed within TaintDroid itself. This addition involves adding a taint tag that would save the name of each application that has touched the sensitive data before it is sent out to an external target. This addition could show if multiple applications are working together to acquire and communicate user data.

Lastly, the PermissionTrigger application can also be completed for testing purposes but it is not required for full functionality of the tool.

4.2 AndroRisk and Risk Assessment Methodology

The methodology as well as the tool need important improvements and completion in order for the tool to be fully relevant.

The following constants should be refined properly with an empirical analysis:

- The impact value of the B function if the task is done in background. (If the value is changed, in order to keep the impact range from 1 to 10, the new value of the total impact should be normalized).
- The base of the logarithm method for the likelihood formula. (Depending on the range of the logs' number, the logarithm base should be adapted)

The questionnaire for the user's impact assessment shall be implemented as suggested in the user-centric approach study [2].

In order to be able to interpret the real meaning of the risk values assessed by the tool, pattern shall be defined with empirical study. For example, a business pattern can be defined saying that the company's smartphones shall not have more than 65 for T1, 55 for T2, 48 for T3, 79 for T4 and 40 for T5.

The tool was completed only for the threat T1 "tracking" and should be completed for all the other threats (the PERMISSION_IMPACT and THREAT_PERMISSIONS maps should be completed).

5. References

[1] M. H., «Google announces 1.5M Android activations per day, and 50B app downloads.,» 2013.

[2] S. Bartsch, «The Transitivity-of-Trust Problem in Android Application Interaction,» 2012.

[3] P. g. B.-G. C. William Enck, «TaintDroid : An Information-Flow Tracking System for Realtime Privacy Monitoring on Smartphones,» *USENIX Symposium ,* 2010.

[4] M. T. a. D. G. Alexios Mylonas, «Assessing privacy risks in Android: A user-centric approach,» 2013.