Dynamic analysis of Android applications to extract spam caller IDs

Christiaan van Luik (c.c.vanluik@student.tudelft.nl) Supervisors: Apostolis Zarras, Yury Zhauniarovich CSE3000 Research Project

Introduction

- People receive 16 **spam calls** per month on average.[1]
- Multiple smartphone applications exist to detect spam calls and block them
- **No insight or research** on how these applications work technically.

Research into extracting blocked Caller IDs has not been done yet.

Finding out how these applications block spam calls, is useful in order to be able to compare **effectiveness** of multiple methods and to determine what would be the **ideal technique or combination of techniques to combat spam calls.**

Research Question

Can we do a brute force dynamic analysis on Android spam call blocking apps, **to extract caller ID** information from apps that cannot be or is not extracted through static analysis?

Subquestions

- How long does this analysis take and how can it be speed up?
- Can we include other information, apart from caller ID, that is used to determine if a call is blocked or not in the analysis?

Method

Tools

Android Debug Bridge – interact with device, send simulated phone calls
Android UI Automator – interact with app layout, send taps and swipes
Appium – open source test automation framework
Appium Inspector – inspect Android layout XML
OpenCV – Computer vision, used for image comparison

References:
1: Hiya, "State of the Phone Call: Half Yearly Report 2019 https://assets.hiya.com/public/pdf/HiyaStateOfTheCall2019H1.pdf?v=6b7b682837c56c47656c012c1da0e6a0

Method (continued)

App selection

- Android App, preferably in the Google Play Store
- App has a significant number of downloads
- App is free
- App uses a database or algorithm to decide whether a call is a spam call. App should not only use a user-defined blocklist.

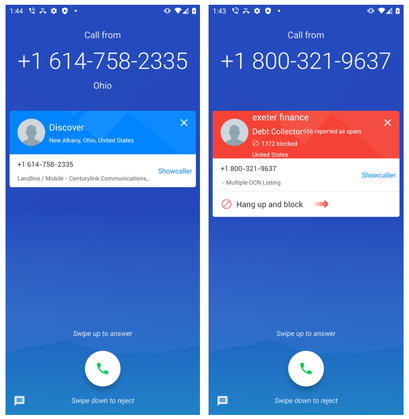

Fig 1: Screenshots of incoming call screens

Fig 2: Allow reference image for Showcaller

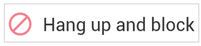

Fig 3: Block reference image for Showcaller

Automated analysis

The tool loops over the 10 apps, **for each app:**

- The app is installed and configured on an emulator
- **For each number** in the input list of numbers:
  a. The device screen is locked (turned off)
  b. A call is simulated
  c. The tool continuously takes screenshots until an occurrence of either an allow, caution (if applicable) or block reference image is present, in which case we can decide on the status of the call.
  d. The simulated call is cancelled.
  e. The number combined with the found status (allow/ caution/ block) is saved.
- The app is uninstalled

Automating app installation

- Installing via ADB and granting permissions not enough
- Every app needs custom setup steps like:
  - Setting default caller ID app
  - Accepting terms & conditions
  - Logging in (with Google account)

Reference image comparison

- Screenshots are taken continuously
- Compared with OpenCV against reference images, until match is found or until timeout.

Fig 4: Reference images of ShouldIAnswer, for respectively the allow, caution and block statuses.

Results

Software tool that can dynamically extract the caller ID information from Android spam blocking apps.

- Determine whether a caller ID is labelled as spam or not by such an app
- Measure the time it takes before this information pops up on the screen.

The results shown in tables 1-3 are obtained by running the tool on a 100-number dataset of both US and Dutch caller IDs.

Incompatible apps

Truecaller – number verification fails on emulator
Call Blocker – no visual indication, only lookup
Call Control – paid functionality

Optimizing performance

- Running headless
- Using multiple threads

App Comparison

Table 1: Time comparison – difference between apps in speed

- Stop Calling Me – Blocks numbers directly, instead of visual indicator. Fastest in detecting and taking action.

Table 2: Call Blocking Statistics – performance on the 100-number dataset.

- Chosen set of numbers can heavily influence how the apps work and influence allow and block percentages.
- Two applications, com.allinone.callerid, com.callerid.block, both show the same behaviour, not only are the percentages the same in the table, but they block exactly the same numbers. Might use same dataset under the hood.
- Same com.webascender.callerid percentages are a coincidence.

Table 3: Dutch and US Numbers comparison

- For US numbers, all tested apps block at least some
- For Dutch numbers, some apps block significantly fewer.

Some apps might be optimized for the US market, and others have more data on Dutch or maybe European numbers.

Limitations

Vulnerable for application changes, mainly UI

- Layout changes can lead to tool failing

Tool cannot deal with caller ID specific layouts
Can only reliably run on 2 threads

Conclusion

- Developed tool can fully automatically test caller IDs on 7 apps
- 1.5 seconds on average for one number (on one app)
- Speedup by:
  - Headless
  - Threads
  - Efficient image processing (OpenCV)
- No results on other information apart from caller ID gathered, interesting for future work

Package Name	Avg. of delta	Min. of delta	Max. of delta	Min. of acc.	Avg. of acc.
com.allinone.callerid	1,26	1,17	1,46	100%	100%
com.callapp.contacts	3,14	1,79	7,69	96%	100%
com.callerid.block	1,33	1,22	2,01	100%	100%
com.mglab.scm	0,03	0,00	0,20	100%	100%
com.telguarder	1,44	1,14	3,22	100%	100%
com.webascender.callerid	1,13	1,05	1,47	98%	100%
org.mistergroup.shouldianswer	1,86	1,59	2,17	100%	100%
Total	**1,45**	**0,00**	**7,69**	**96%**	**100%**

Table 1: Statistics on running time (delta) per number in seconds and accuracy of the image comparison per number. Timeouts are excluded.

Package Name	allowed	blocked	caution	timeout
com.allinone.callerid	68%	32%	0%	0%
com.callapp.contacts	85%	11%	0%	4%
com.callerid.block	68%	32%	0%	0%
com.mglab.scm	74%	26%	0%	0%
com.telguarder	39%	24%	37%	0%
com.webascender.callerid	68%	32%	0%	0%
org.mistergroup.shouldianswer	12%	82%	6%	0%
Total	**59%**	**34%**	**6%**	**1%**

Table 2: Overview of percentage of allowed and blocked numbers for the 100-number test dataset.

	Dutch Numbers			US Numbers		
Package Name	allowed	blocked	caution	allowed	blocked	caution
com.allinone.callerid	54%	1%	0%	14%	31%	0%
com.callapp.contacts	50%	4%	0%	35%	7%	0%
com.callerid.block	54%	1%	0%	14%	31%	0%
com.mglab.scm	55%	0%	0%	19%	26%	0%
com.telguarder	29%	16%	10%	10%	8%	27%
com.webascender.callerid	31%	24%	0%	37%	8%	0%
org.mistergroup.shouldianswer	2%	53%	0%	10%	29%	6%
Total	**39%**	**14%**	**1%**	**20%**	**20%**	**5%**

Table 3: Comparison of app performance between Dutch and US caller IDs