# Online Communication Traces in Android Memory

Jørgen Ellingsen, Emil Volckmar Ry, Anders Sefjord Torbjørnsen
Espen Kjellstadli Lund,
{jorgeell, emivr, andetorb, espenklu} @stud.ntnu.no

## ABSTRACT

Write a short abstract here.

## Keywords

Android; Memory; Forensics;

## 1. INTRODUCTION

A lot of communication today are done with mobile applications like Facebook, GMail and Snapchat. In this work we analyze what information is stored in memory, and how long it is recoverable for the different applications.
Some applications claim to have features like anonymity and auto-delete funcionallity, and applications like this is interesting to examine for memory leaks.

Today the battery life of smartphones has reached the point that even with all the usage we do on our phones today, it will in most cases last the entire day. As a result of this, smartphone can stay on for weeks, or even months, at a time and old information can still be present in the memory.

## 2. BACKGROUND

Many of todays smart phones are running Android as their operating system, and data claims it dominates the market with an 87.6% share in the second quarter of 2016[1]. Making it essenstial for future mobile forensic work, for gathering information in an investigation.

When an investigation occurs, there are several approaches to data acquisistion: (1) Manual acquisition, (2) Logical acquisition, (3) Physical acquisition, (4) Brute force acquisition. The field of interest for this paper is physical acquisition of the primary storage; it focuses primarly on creating a bit-by-bit copy of the random access memory (RAM). Due to RAM being volatile, it is often the target of stealthy illegal activites to avoid leaving data. If data was stored in the flash drive, it would still be resident until OS tries to overwrite the same physical area and garbage collector is initiated. The point being that data residing on a secondary storage device, will be living longer and probably be logged more extensively.

The Android OS is in short words a Linux-based OS, where most OS tasks are performed by open source C libraries, and Java is used for the development of Android applications. These applications are compiled to bytecode for the Java virtual machine (JVM), which is then translated for a second virtual machine which executes them. Depending on which version of Android a smart phone is running, different virtual machines are used for execution. For Android versions 4.4 and prior, Dalvik was used. It was replaced by ART in 4.5, and is still the de-facto standard.

Why does this matter? The virtual machines that are used are suppose to run Java applications, and have therefore gained several features of the Java programming language. One of these features are the memory management module, which has a built-in garbage collector(GC). It lets the user create objects without worrying about memory allocation and deallocation. Reducing the need for boilerplate code, and problems with memory leaks and such, which often are languages like C and C++ are subject to.

The GC have the job of cleaning RAM, therefore removing potential information to be gathered. It is therefore of severe importance, that data residing in memory gets gathered before the device eventually power offs, or the garbage collector initializes.

Both ART and Dalvik use by default a method called ”concurrent mark and sweep” for their GC[2, 3]. It works by traversing the heap for objects that are ”reachable” or used by applications, those who are not will be regarded as free space again. Making space for potential new data to be stored within it. A figure of the process, can be seen at figure 1. Generally the heap is a region of the memory that is regarded as free memory for any process to use, however in Java it is used to store all objects created. Therefore the GC may potentially remove one of the better sources for information, the objects. The good news are that once data is freed, no overwriting is enforced; in other words, data is not overwritten until another object takes its place[3]; in addition to each application running its own garbage collector and private heap, making the RAM data resident for possibly long periods of time.

Because all objects created in the heap, there may be a potential trace of information of an application. Through the use of several popular online communication applications, the amount of data that can be collected through a memory dump will be explored; despite knowing that the GC might start, or extra security measures might prevent us to do so.

## 3. PRACTICAL APPLICATION

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec

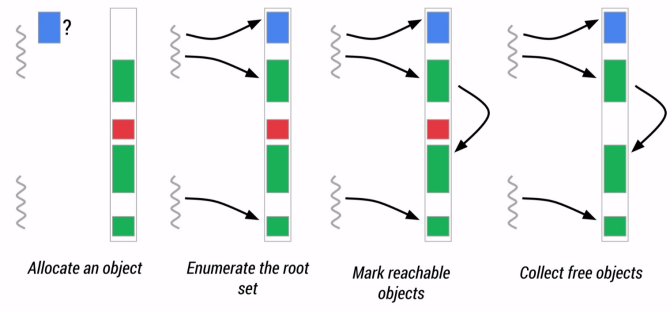


**Figure 1: Mark and Sweep[2]**

vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

## 3.1 Labratory Environment

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetuer.

## 3.2 What did we do

Suspendisse vel felis. Ut lorem lorem, interdum eu, tincidunt sit amet, laoreet vitae, arcu. Aenean faucibus pede eu ante. Praesent enim elit, rutrum at, molestie non, nonummy vel, nisl. Ut lectus eros, malesuada sit amet, fermentum eu, sodales cursus, magna. Donec eu purus. Quisque vehicula, urna sed ultricies auctor, pede lorem egestas dui, et convallis elit erat sed nulla. Donec luctus. Curabitur et nunc. Aliquam dolor odio, commodo pretium, ultricies non, pharetra in, velit. Integer arcu est, nonummy in, fermentum faucibus, egestas vel, odio.

### *3.2.1 Extracting memory*

To extract the memory AMExtract was chosen. It did not have a profile for the phone , so a profile for extraction method, size of and other properties was created and tested.

The tool was then compiled using the ndk-build tool. As part of the linux kernel headers has been modified, one of the types had to be redefined using an older header file.

### *3.2.2 Searching memory*

### *3.2.3 Carving memory*

## 3.3 What were we looking for

Sed commodo posuere pede. Mauris ut est. Ut quis purus. Sed ac odio. Sed vehicula hendrerit sem. Duis non odio. Morbi ut dui. Sed accumsan risus eget odio. In hac habitasse platea dictumst. Pellentesque non elit. Fusce sed justo eu urna porta tincidunt. Mauris felis odio, sollicitudin sed, volutpat a, ornare ac, erat. Morbi quis dolor. Donec pellentesque, erat ac sagittis semper, nunc dui lobortis purus, quis congue purus metus ultricies tellus. Proin et quam. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos hymenaeos. Praesent sapien turpis, fermentum vel, eleifend faucibus, vehicula eu, lacus.

# 4. RESULTS

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

# 5. DISCUSSION

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

# 6. CONCULSION

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

[4]

# 7. ADDITIONAL AUTHORS

# 8. REFERENCES


[1] IDC. Smartphone OS Market Share, 2016 Q2; 2016. http://www.idc.com/prodserv/smartphone-os-market-share.jsp.

[2] Carlstrom B, Ghuloum A, Rogers I. The ART runtime. Google I/O; 2014. .

[3] Anikeev M, Freiling FC, Götzfried J, Müller T. Secure garbage collection: Preventing malicious data harvesting from deallocated Java objects inside the Dalvik VM. Journal of Information Security and Applications. 2015;22:81–86.

[4] Androulidakis II. In: Mobile Phone Forensics. Cham: Springer International Publishing; 2016. p. 87–109. Available from: http://dx.doi.org/10.1007/978-3-319-29742-2_6.