# Analysis and Design

## Introduction

This chapter explains the details of analysis and design process of this project that will development of malware analysis tool. The figures follow will show the graphical representation of the software phases.

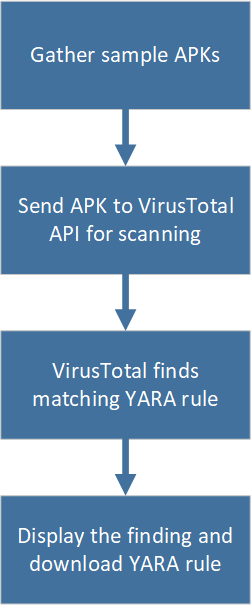


Figure ‎4.1.1 Yara Generation Phases

## Sample Malware

Sample of malwares are handpicked from the year 2019 to 2021 to be tested by this project. The source of the malwares can be found in Koodous and github repositories. The table below show the list of malware samples that are used.

|  |  |  |
| --- | --- | --- |
| Package name | SHA256 hash | Year in the wild |
| com.saver.batterymobi | 64ebe9b975de022b888f17db429af3a93d3db95db5af274e3eefd3ca7f24e350 | 2019 |
| operatore.italia | 0f5f1409b1ebbee4aa837d20479732e11399d37f05b47b5359dc53a4001314e5 | 2019 |
| com.gelini | 48618153df1b2b5be3f83e6e1fa6aa5f517b173b10f3f6e925d1598a22b459e1 | 2019 |
| com.tencent.mm (flubot) | 30937927e8891f8c0fd2c7b6be5fbc5a05011c34a7375e91aad384b82b9e6a67 | 2021 |
| kijxlnbftwdhbbet.eaafsym.fziuffcjyjetmqxsmcd (aug\_banking.apk) | fe2e8b115b3ffc2f3ab668c08c67b21afa6761426cef1c6a99f6cb9074d8076f | 2020 |

## Yara Rule Structure

Each Yara rule has to start with the word rule, followed by the name or identifier. Rules are composed of several sections, the meta, strings, and condition. The condition section is the only one that is required. It contains a Boolean expression that determines the result whether it is true for the object (file). The strings section is where the strings that will be looked for in a file will be defined. Things that can be looked for in strings are hexadecimals, text strings, and regular expressions. The meta can be added to help identify the files that were picked up by a certain rule. The only purpose for it is to store additional information about the rule.

Graphical user interface, text

Description automatically generated

Figure ‎4.3.1 PromonShield Rule

## Yara Generation Tools Analysis

### YaraGenerator

YaraGenerator is a tool to attempt to allow for quick, simple, and effective yara rule creation to isolate malware families and other malicious objects of interest.

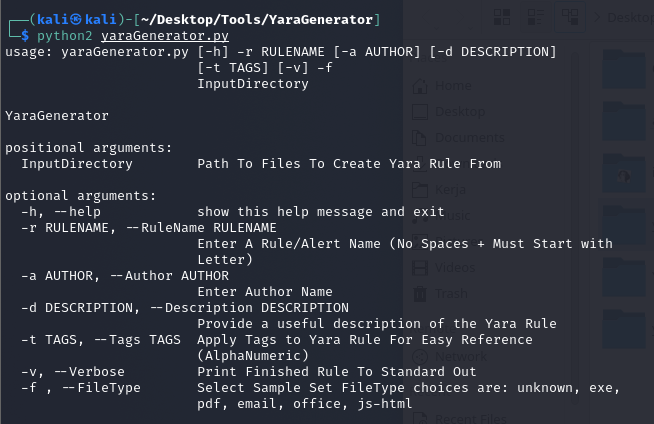


Figure ‎4.4.1 yaraGenerator.py

The file types that are accepted are exe, pdf and emails. However, it does not support APK file and the project is outdated since its last update is on 2013.

### YarGen

YarGen is also a Yara creation tool using strings found in malware files while removing all strings that also appear in goodware files.

Text

Description automatically generated

Figure ‎4.4.2 yarGen.py

It’s repository also details on how to write and post-process Yara rules generated by YarGen. It does not recognize APK file or extracted APK file however.

### Koodous

Koodous is a collaborative web platform for research on Android malware using malware analysis tools and Yara rule matching. Koodous will use it’s own tools to extract information from APK files into a JSON report to be matched with rules.

Graphical user interface, text, application

Description automatically generated

Figure ‎4.4.3.1 DroidGuardVM ruleset on Koodous

Despite having a large collection of community written Yara rules, recent changes to API usages have made malware analysis somewhat limited. Malware analysis requests and downloading malware sample are only made exclusive for paid API keys.

Graphical user interface, application

Description automatically generated

Figure ‎4.4.4.2 Koodous API usage limit

### Androguard module for Yara

Androguard is part of Koodous project that integrates static APK analysis with Yara. This can be used to find APKs by package name, permissions, or API level. This module is ready to use with Koodous reports using python script to get it automatically. Unfortunately, the project is no longer maintained as Koodous rejects old API key.

### VirusTotal API

VirusTotal's API lets you upload and scan files, submit and scan URLs, access finished scan reports and make automatic comments on URLs and samples without the need of using the HTML website interface which allows for simple scripts to access the information generated by VirusTotal.

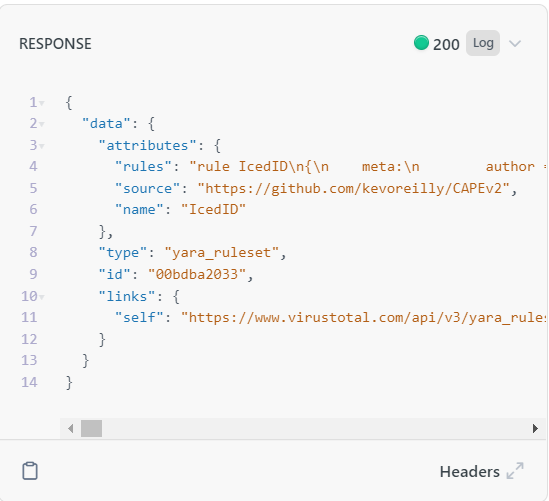


Figure ‎4.4.5 VirusTotal Yara ruleset API response

Using this API we can implement malware analysis in app.

## App Design



Figure 4.5.1 App main page

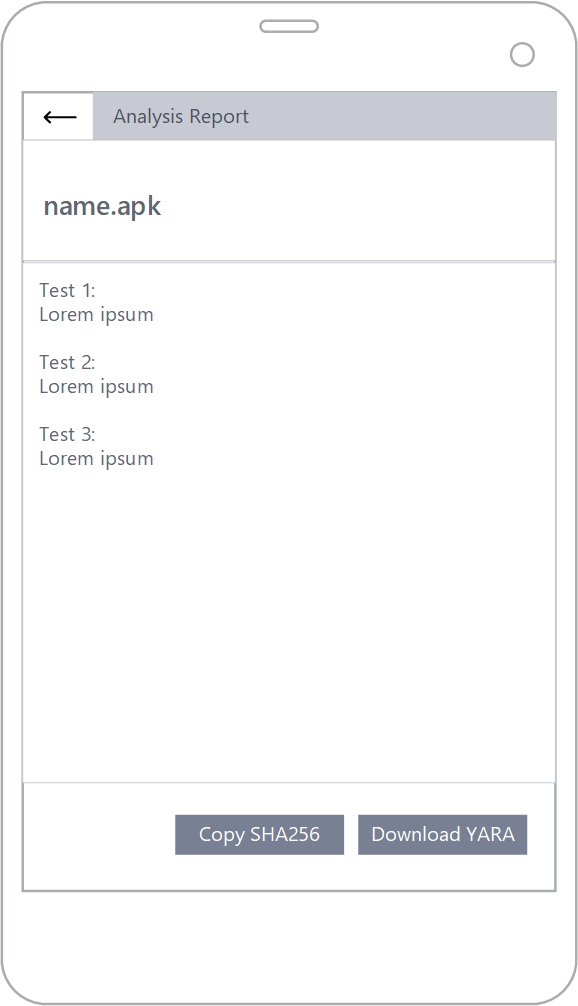


Figure 4.5.2 App analysis report page

## Conclusion

This chapter explores the Yara generation phases along with the tools that will be used. This is followed by the design of the app that’s going to be developed. The next chapter will go over the project’s implementation.