ANDROID APPLICATION TO MONITOR AND PROTECT FROM REMOTE ACCESS TROJAN

Muhammad Alwi Shihab – 201481096
wishihab@gmail.com
Faculty of Computer Science – Information Technology
Esa Unggul University
Advisor I : Li Qing
Advisor II : Malabay, S.Kom, M.Kom

Abstract – The Mobile phone, where people always use on daily life and total of phones is bigger than the people population itself and internet has become an essential part of mobile phone for half of people population. The mobile and internet are together had many services and growing every second then more people are using these services. Internet banking and Social media are the examples of internet services using mobile phone, these services make bad people try to take advantage of legitimate users whenever money is involved. Android operating system has the highest market share in the world, this fact makes android users target group criminals. Malware is one of cybercrime software attacked on mobile phone, Remote Access Trojan included in variant of malware. Trojan is one of the most malware founded in android application, this Trojan allows attacker to go through the device system without notice to the owners. The user awareness is one of point that vulnerability, doesn't realize if users installed some malware and it steal anything. This paper presents a network activity, permission usage and encryption-based to detect and protect Android system.

Keywords: *Android, Trojan Malware, Remote Access Trojan, User Awareness.*

INTRODUCTION

These days a modern era, technology are growing so fast on every aspect, mobile phone is one of technology. Current mobile devices offer a large amount of service and applications than those offered by personal computers. At the same time, security threats increase and mobile devices became a target. The malicious users and hackers are taking advantage of both the limited capabilities of mobile devices and the lack of standard security mechanisms to design malicious application that can access sensitive data, stealing phone credit, spying users and crashing the device. Everyday malware attack increased in specially on Android platform.

Android's security model significantly relies on permission-based mechanisms. Permission are requested by applications and they are required to access application's interface/data that are defined in its manifest file. They are used to inform users about application's capabilities and the user does not realize/aware that Android security frame work does review applications permissions, but Android security framework does not review system though their network activity which to inform users which connection has connected to the device.

Kabakus et al. (2015) present a tool named APK Auditor that offers static analyses with permission-based Android malware assessment system.

Burguera et al. (2011) present Crowdroid, a framework to detect anomalous behavior of Android application. In order to categorized Android application as malicious, Crowdroid that collects Linux Karnel system calls containing device information.

Acosta-Guzman et al. (2015) present a Network Activity Monitoring Against Malware in Android Operating System as methodology to malware detection.

This application presents a tool to against Malware as specific Malware is Remote Access Trojan based on Network Activity, Permission Usage and Encryption. There are variety of Trojan which has been created in this world for example Backdoor, Exploit, Ransom, Spy, etc. Remote Access Tool is software that used to control operating system on devices and in this case Android system. However, there are a lot of illegal remote access tool one of which is known as Trojan horse malware on the internet to attack device and gain full control without any permission and user cognition. Malicious software or malware is used by Cybercriminals, Hacktivists, and Government to harm devices, steal personal information, spying users, etc. Malware contains an executable script or code used to bypass access control and disguise itself as a common software to deceive system and also antivirus.

Problem Identification

Based on the background, the author can identify the problems for development of this application are as following:

1. How to identify Remote Access Trojan via *Network Activity-Based?*
2. How to Identify Remote Access Trojan via *Permission-Based?*
3. How to identify Remote Access Tool Trojan via *Third-Party* application installed?
4. How to identify Remote Access Trojan via *Permission-Usage?*
5. How to monitor and protect from Remote Access Trojan?

Limitation of the study

On this application project has some limitation in this study are as follows:

1. This application is applied or require specification to the Android operating system version 5.0 to version 9.0
2. This application cannot identify network status with application name detail.
3. This application has no uninstall feature.

Purpose of the study

The purpose of this system development are as follows:

1. This application is built to helpm facilitate and increase user awareness from remote access Trojan.
2. This application is designed to monitor and protect mobile phone from remote access Trojan.

Benefits of the study

The benefits of this research are as follows:

1. This application can help users to monitoring background connection between the device and other network.
2. This application can help users to finding a third-party application installed in device.
3. This application can help users to find which third-party application use specific permission.
4. This application can help users to finding location of network address which specify to Trojan server.

Research Methodology

In this thesis, the methodology to be used are as follows:

1. Data Collection Method
   In this method, the authors utilize the resources of the library references such as journals, articles and the internet to get the supporting theories and references for the development of application.
2. Software Design Method
   The method used in the design of software is using extreme programming (Pressman, 2012). The process used in as follows:
   a. Analysis of needs
      Analysis of needs applied by identifying that needs to be obtained. The analytical method used by using the modelling language UML (Unified Modelling Language). Analysis of needs is done by identifying all of the software requirements based on the data obtained from the study of literature and then modelled into the use case diagram.
   b. Coding
      Coding is the stage for designing the application using Android Studio, and implement the UML model also Java Android.
   c. Design
      This stage emphasizes the design of simple applications. Each requirement that have been poured into the UML model is determined in the construction period.
   d. Testing

This stage focuses on testing the feature and functionality of the application, whether it meets the requirements and solve the problem or not.

e. Documentation
This stage is the preparation for the application documentation.

THEORETICAL BASIS

Remote Access Trojan

A remote access Trojan (RAT) is a malware program that used for malicious purposes, this Trojan inspired by remote access tools. Remote access tool is a software used to remotely or control a computer with legitimately used by system administrator. RAT also known as a backdoor for administrative control over the target device. These days, a million of remote access Trojan has been created which they free or paid with multiple operating system as target. RATs are usually downloaded invisibly with a user-requested program such as common software, game or sent as an email attachment. Once the target system is infected, it will perform unauthorized operations and hide in the system. The attacker can remotely control the system to gaining the camera, microphone, key logs, screen capture, harm the device, etc.

Network Activity Based

Network Activity Monitoring Based, according to Acosta-Guzman [11]. It is important to emphasize that the approach of this methodology consists in the development of the mechanism to keep track of the network activity not the mechanisms to evaluate and detect any suspicious behavior and nor to identify malware. However, methodology shows the implementation of a Network Activity Based for the application named Network Status related on Acosta-Guzman project, this Network Status is capable to running and update every user reopen this feature in order to identify new established connections and attempts of connection. This Network Status capable show internet protocol and listening open ports on it, lead the user to identifying network on device.

Considering the aforementioned Acosta-Guzman methodology, Network Activity Monitoring App works without the need of rooted device. The network activity itself cannot be used to identify malware but it could be used to identify all connection running on device which a Remote Access Trojan has server address used on target client. As a consequence, the user running this network status on its device may have an opportunity to kill the process and uninstall application that is generating new network activity. Alerting the user is the only reactive measure the network status can take without requiring root permissions.

Permission Usage

Permission Usage Based, according to Abdullah Talha on APK Auditor(2014). Android's security mechanism is based on an instrument that informs users about which permissions the application needs to be granted before installing them. This permission system provides an overview of the application and may help gain awareness about the risks. However, user do not have enough information to conclude that standard users read or digital investigators understand these permissions and their implications. Digital investigators need to be on the alert for the presence of malware when examining Android devices, and can benefit from supporting tools that help them understand the capabilities of such malicious code. Android applications are packaged as Android application (APK) files which contain manifest file (AndroidManifest.xml), compiled Java classes and application resources. The system developed uses static analysis techniques based on permissions in order to characterize and extract profiles for Android applications [8].

AES Encryption

The Advanced Encryption Standard, or AES, is a symmetric block cipher chosen by the U.S. government to protect classified information and is implemented in software and hardware throughout the world to encrypt sensitive data.

AES, or Advanced Encryption Standards, is a cryptographic cipher that is responsible for a large amount of the information security that you enjoy on a daily basis. Applied by everyone from the NSA to Microsoft to Apple, AES is one of the most important cryptographic algorithms being used in 2018. AES or Advanced Encryption Standards (also known as Rijndael) is one of the most widely used methods for encrypting and decrypting sensitive information in 2017 [28].

ANALYSIS AND DESIGN

Analysis of the Current System

Based on the author experiences, experiment, observations to find any system has been use and created there are two system that used by people to against remote access Trojan, as follows:

a. Manual System
   The manual system for finding remote access Trojan by internet or article tutorial.
b. Application System
   The application system is people are using Anti-Virus.

Weakness of the Current System

In the analysis of the current system above, the author found some problems encountered. The following is an analysis of the problems of the current System:

a. Manual System
   The manual system can take a long time and many step for finding, identifying and monitoring Android device from remote access Trojan, because it is possible that internet source tutorial different and
   too much sources make a user confused. This manual system also possible to destroy the Android system, because there no experience on user.
b. Application System
   The application system can take faster than manual, but antivirus system is decrease perform for some Android system. A lot of antivirus has no monitoring feature because they use automatically system and 2017 Google launch Google Play Protected as security system default of Android Play Store.

Literature Review

At this stage conducted literature review, which is studying reference similar research result that have been done, journal and article. It aims to obtain the theoretical foundation needed in the study.

Study of Literature

At this stage the author taken a data by comparing similar literature, either from literature or from the field in the form of similar applications that have been made before. The data generated at this stage are advantages and disadvantages of comparable objects.

Software Development Method

Software development method is a set of rules and guidelines used in the research process, planning, designing, developing, testing, setup and maintaining a software product. There are many software development models and many organizations create and use their own model. Choosing the model has a high impact on testing. The most commonly used models are:

1. Waterfall;
2. V Model;
3. Rapid Application Development;
4. Iterative Model;
5. Spiral Model;
6. Crystal Model.

Each model has advantages and drawbacks. Agile methods are based on adaptive software development methods, while traditional SDLC models (waterfall model, for example) are based on a predictive approach. In traditional (Stoica et al., 2013)

Diagram Application with UML

In Design Applications to be proposed using UML diagrams as follows:

Use Case Diagram

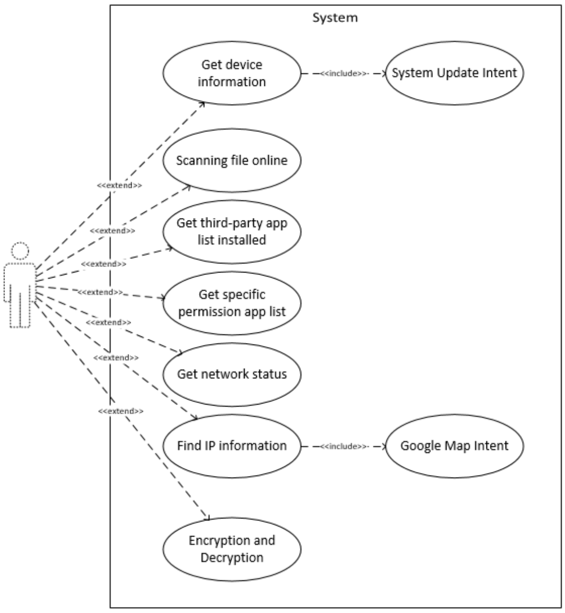


Figure 1 *Usecase Diagram* WeDefend

The explanation of eaarch use case is as follows:

1. *Use Case* Get Device Information
   This system will display all information of device.
2. *Use Case* Scanning File Online
   System will open webView of Kaspersky online scanner and users can use this feature to scan files.
3. *Use Case* Get Third-Party App
   Users can get all third-party list application.
4. *Use Case* Get Specific Permission
   The system displays the specific application list with specific permission.
5. *Use Case* Get Network Status
   Users can get all information of network status, between device to other device or server.
6. *Use Case* Finding IP Address
   Users can use this feature to identify IP information detail.
7. *Use Case* Encryption & Decryption
   This feature can help users to protect files.

Activity Diagram

Activity diagram is similar to flowchart, because it can model a workflow from one activity to another. Here is some activity diagram for this thesis:

1. Activiy Diagram Main Page

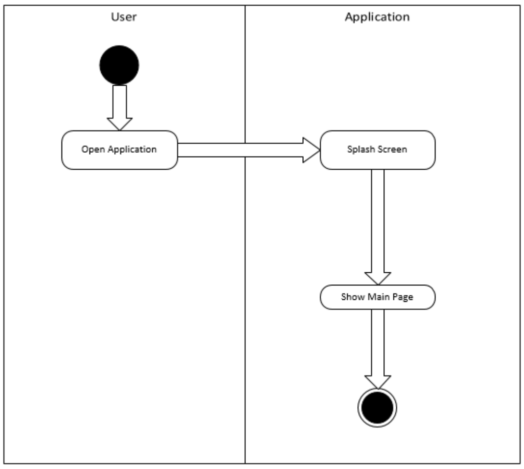


Figure 2 *Activity Diagram* Main Page

From the diagram above, the main activity starts with a splash screen and the application will show main page.

2. Activity Diagram My Device

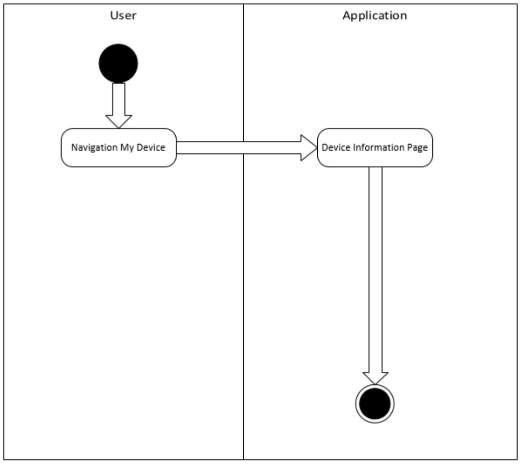


Figure 3 *Activity Diagram* My Device

3. Activity Diagram Scan File

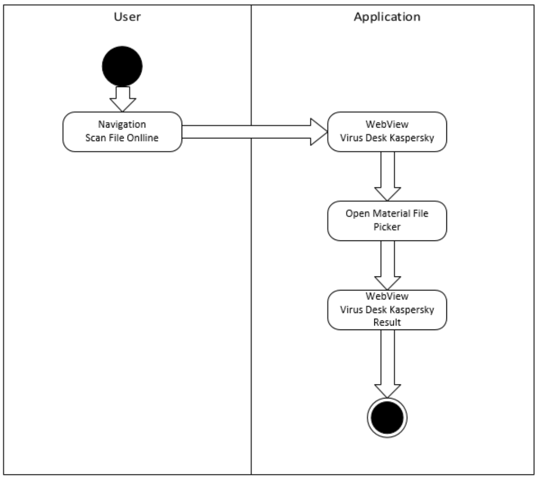


Figure 4 *Activity Diagram* Scan File

4. Activity Diagram Check App

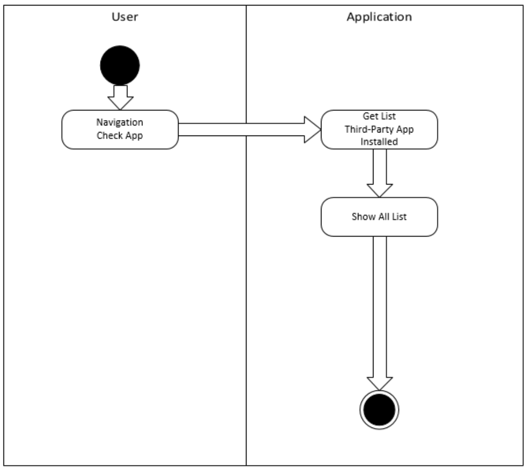


Figure 5 *Activity Diagram* Check App

5. Activity Diagram Permission List

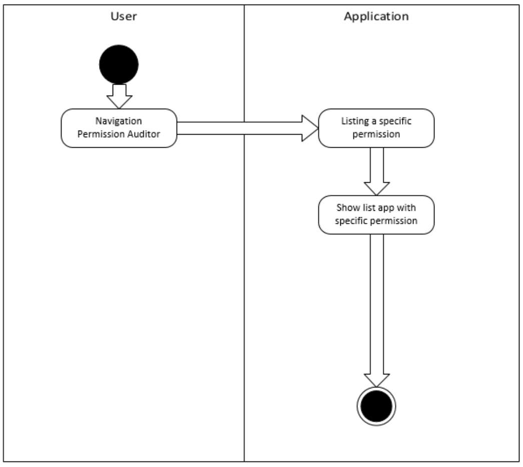


Figure 6 *Activity Diagram* Permission List

6. Activity Diagram Network Status

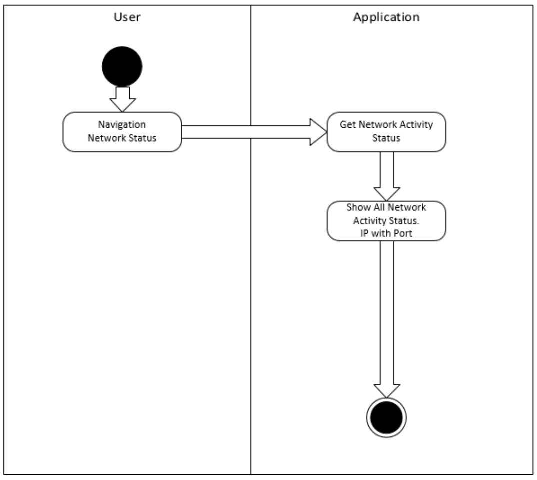


Figure 7 *Activity Diagram* Network Status

7. Activity Diagram Finding IP

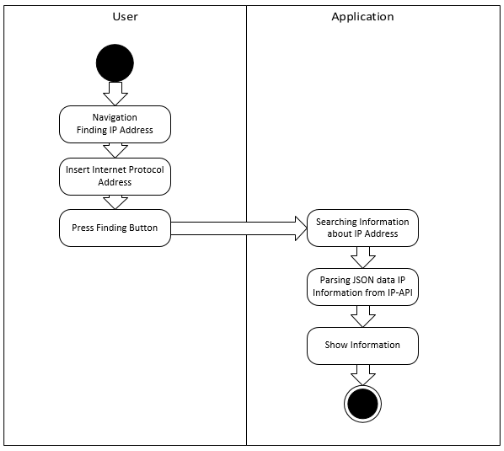


Figure 8 *Activity Diagram* Finding IP

8. Activity Diagram Encryption

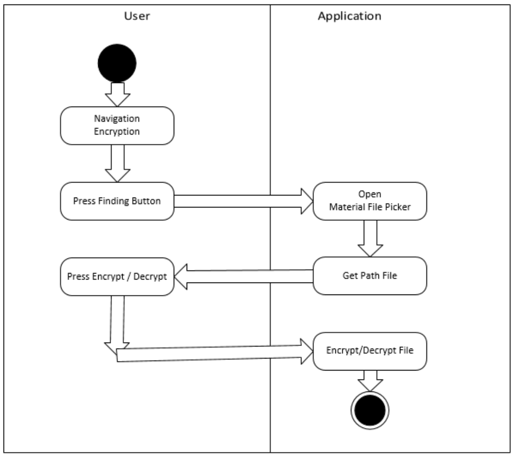


Figure 9 *Activity Diagram* Encryption

State Chart Diagram

1. State Machine Scan File Online

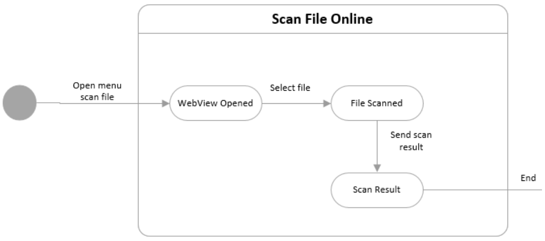


Figure 10 *State Machine*

2. State Machine Check App

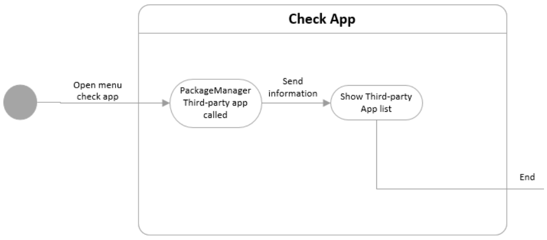

Check App
Open menu check app
PackageManager Third-party app called
Send information
Show Third-party App list
End

Figure 11 *State Machine*

3. State Machine Permission Auditor

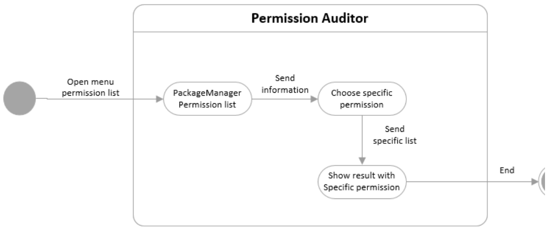

Permission Auditor
Open menu permission list
PackageManager Permission list
Send information
Choose specific permission
Send specific list
Show result with Specific permission
End

Figure 12 *State Machine*

4. State Machine Network Status

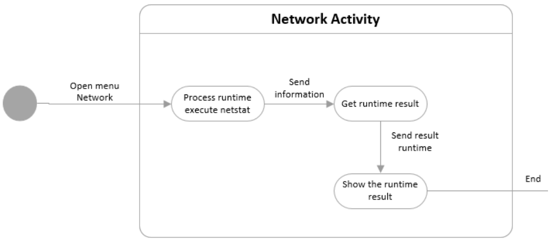


Figure 13 *State Machine*

5. State Machine Finding IP

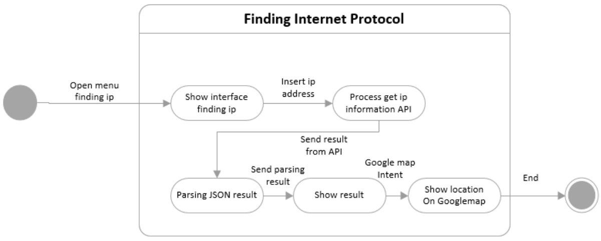


Figure 14 *State Machine*

6. State Machine Encryption

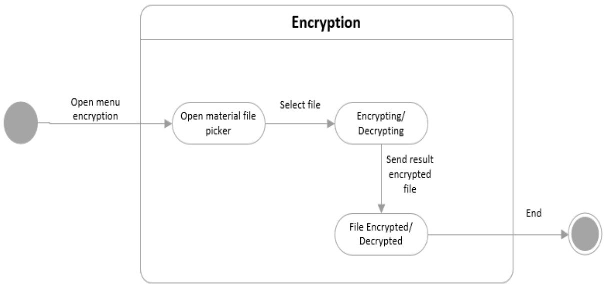

Encryption
Open menu encryption
Open material file picker
Select file
Encrypting/ Decrypting
Send result encrypted file
File Encrypted/ Decrypted
End

Figure 15 *State Machine*

Deployment Diagram

Deployment diagrams model the run-time architecture of a system. The diagram shows the configuration of a hardware element used by the system in the form of a node and shows how the software and system artifacts are mapped in the node.

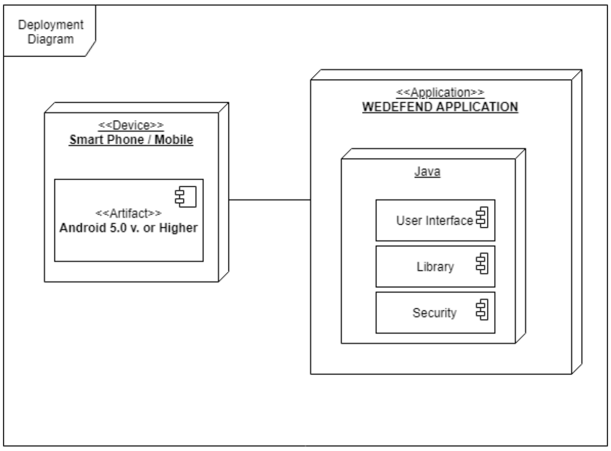


Figure 16 *Deployment Diagram*

IMPLEMENTATION

Implementation

For developing this application, the author uses some devices software and hardware. Software that used in the development of this application is as follows:

1. Android Studio 3.0.1
2. Android Virtual Device 8.0
3. MeMu Emulator Android 5.1

Android Studio is used for writing program code and to test program in Android Virtual Device before the program is tested on real devices.

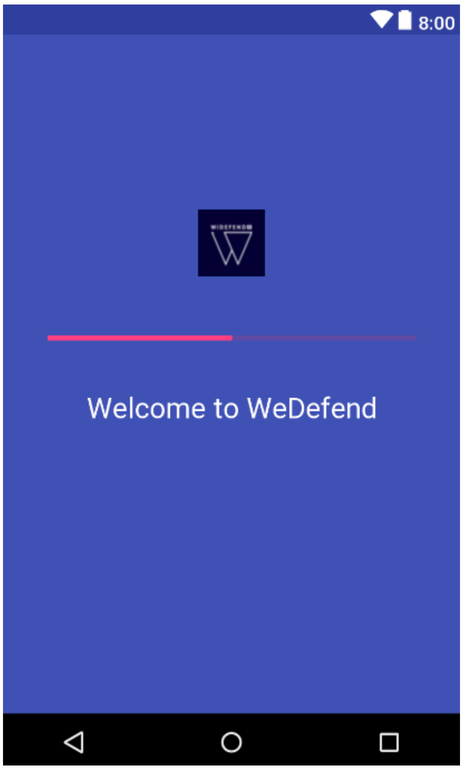


Figure 17 Splash Screen

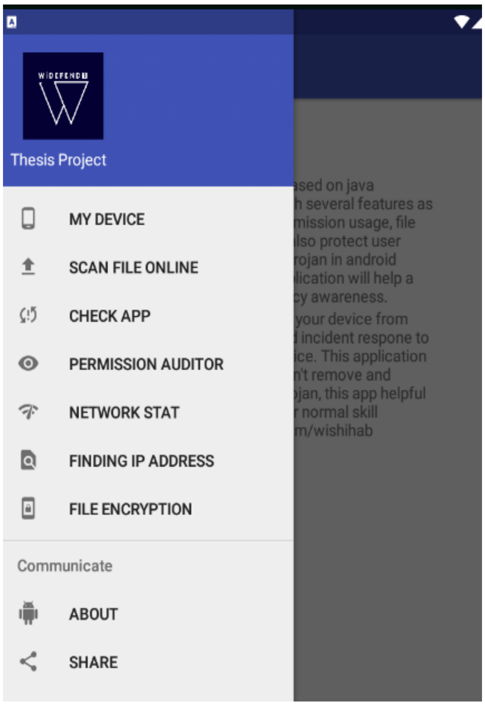
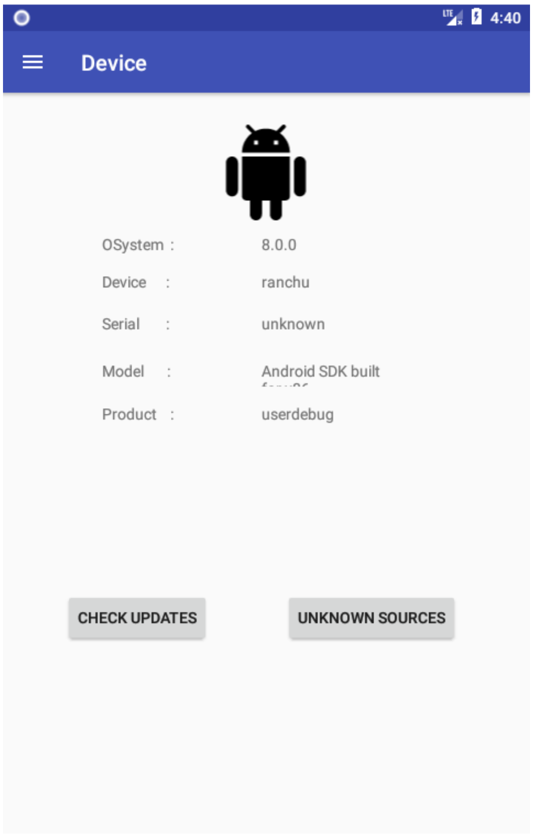


Figure 18 Navigation Drawer Interface



OSystem : 8.0.0

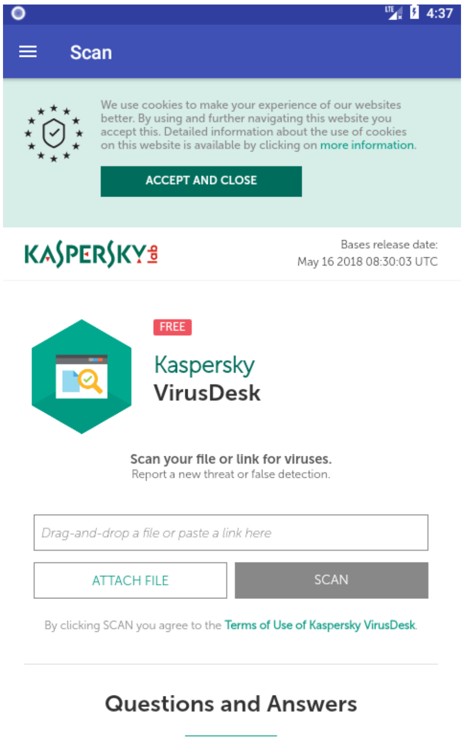
Device : ranchu

Serial : unknown

Model : Android SDK built

Product : userdebug

Figure 19 My Device Interface



4:37

Scan

We use cookies to make your experience of our websites better. By using and further navigating this website you accept this. Detailed information about the use of cookies on this website is available by clicking on more information.

ACCEPT AND CLOSE

KASPERSKY lab

Bases release date:
May 16 2018 08:30:03 UTC

FREE

Kaspersky VirusDesk

Scan your file or link for viruses.
Report a new threat or false detection.

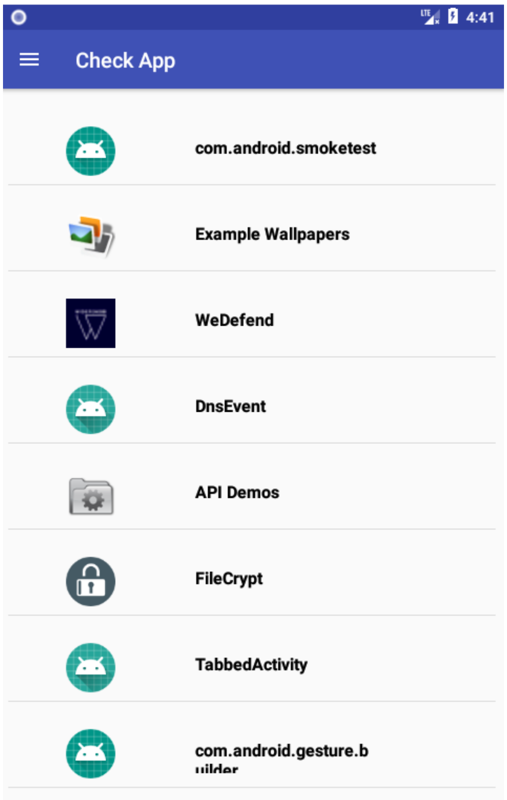
Drag-and-drop a file or paste a link here

ATTACH FILE | SCAN

By clicking SCAN you agree to the Terms of Use of Kaspersky VirusDesk

Questions and Answers

Figure 20 Online Scanner Kaspersky

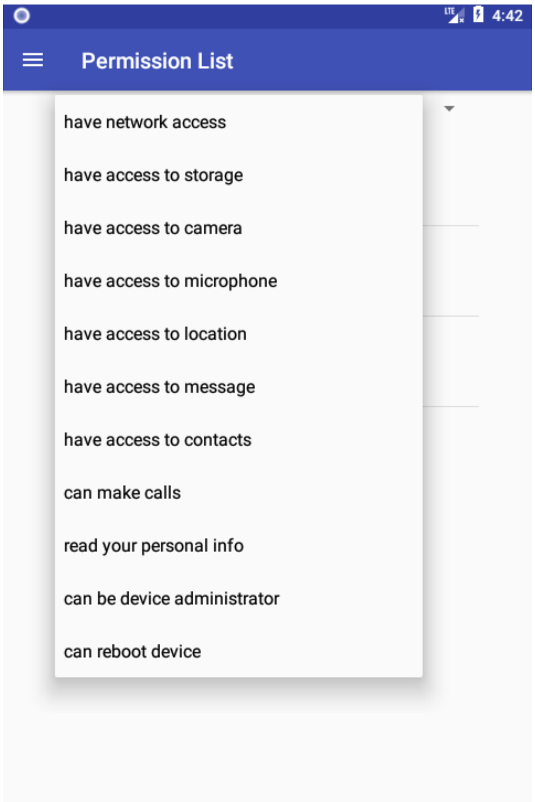


4:41

Check App

- com.android.smoketest
- Example Wallpapers
- WeDefend
- DnsEvent
- API Demos
- FileCrypt
- TabbedActivity
- com.android.gesture.builder

Figure 21 Check Third-party List Interface

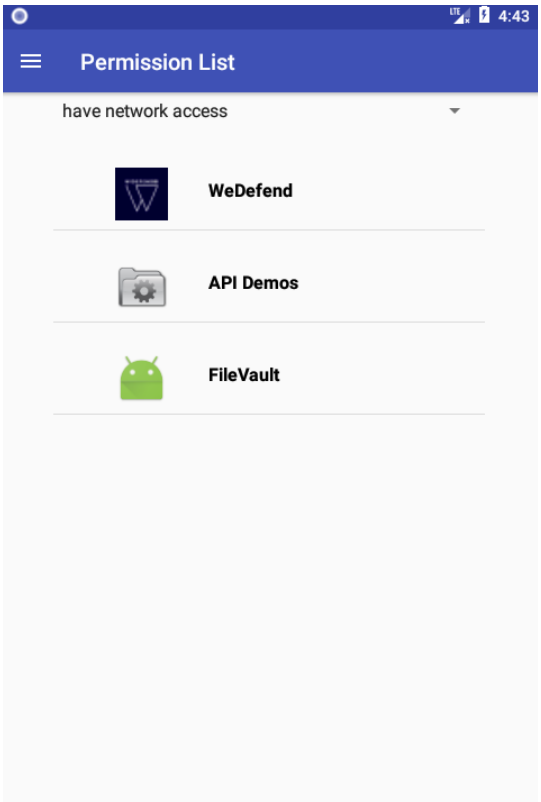


4:42

Permission List

- have network access
- have access to storage
- have access to camera
- have access to microphone
- have access to location
- have access to message
- have access to contacts
- can make calls
- read your personal info
- can be device administrator
- can reboot device

Figure 22 Permission List



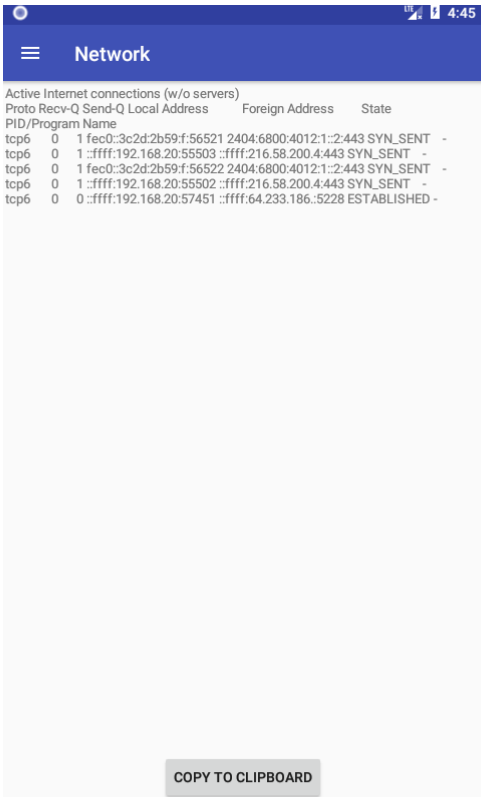
4:43

Permission List

have network access

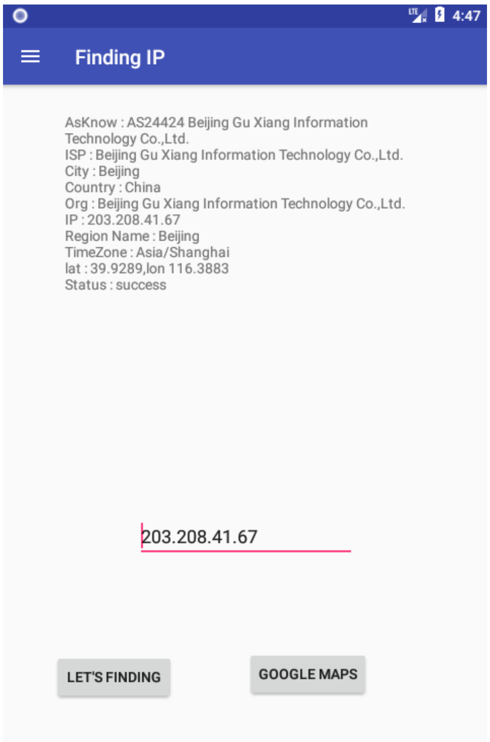
- WeDefend
- API Demos
- FileVault

Figure 23 Permission List with specific permission



```
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address          Foreign Address        State
PID/Program Name
tcp6       0      1 fec0::3c2d:2b59:f:56521 2404:6800:4012:1::2:443 SYN_SENT    -
tcp6       0      1 ::ffff:192.168.20:55503 ::ffff:216.58.200.4:443 SYN_SENT    -
tcp6       0      1 fec0::3c2d:2b59:f:56522 2404:6800:4012:1::2:443 SYN_SENT    -
tcp6       0      1 ::ffff:192.168.20:55502 ::ffff:216.58.200.4:443 SYN_SENT    -
tcp6       0      0 ::ffff:192.168.20:57451 ::ffff:64.233.186.:5228 ESTABLISHED -
```

Figure 24 Network Activity



AsKnow : AS24424 Beijing Gu Xiang Information Technology Co.,Ltd.
ISP : Beijing Gu Xiang Information Technology Co.,Ltd.
City : Beijing
Country : China
Org : Beijing Gu Xiang Information Technology Co.,Ltd.
IP : 203.208.41.67
Region Name : Beijing
TimeZone : Asia/Shanghai
lat : 39.9289,lon 116.3883
Status : success

203.208.41.67

Figure 25 Finding IP Address

CONCLUSION AND SUGGESTION

Conclusion

This thesis aims to help people monitoring and protecting their Android device from remote access Trojan, the applications very useful for users whom want to identify and find manually if the system infected or not. This application can be concluded as follows:

1. Identify current system information and easy lead users to update manually.
2. Scanning files before use it, is the first aid method to identify malware.
3. Users can be easy to check list third-party app installed in device.
4. Permission Auditor is the easy way to categorize third-party application on specific permission for users.
5. Network Status is the best way to identify device infected by remote access Trojan.
6. This application help the users to trace or find sources Internet Protocol Address detail information, very useful for identification.
7. Encryption is very helpful for users who want protect their sensitive files.
8. This application is not difficult to use for Non-experienced users.

Suggestion

This application is still had a weakness and shortcomings therefore this application still requires further development to improve the features in this application. Here are some suggestions for further development of the application:

1. This application only scan by online scanner. It is better if this application can scan offline for specific remote access Trojan virus.
2. This application only shows all the list third-party installed on device. It is better if the application show detail information of third-party app and can terminate process and uninstall the app.
3. This application network status feature only shows connection between device and other connection, it is better if this feature can tell which application use specific Internet Protocol address, port and categorize with whitelist address.

4. Encryption feature only support for a few extension and size files. It is better if this application support for all extension and all size and much better if this application has feature files vault to encrypt, save and open inside app.

REFERENCES

[1] I. Burguera, U. Zurutuza and S. Nadjm-Tehrani, "Crowdroid: Behavior-based Malware Detection System for Android," *ACM,* 2011.

[2] W. Te-En, M. Ching-Hao, A. B. Jeng, W. Horng-Tzer and W. Dong-Jie, "Android Malware Detection via a Latent Network Behavior Analysis," *IEEE,* 2012.

[3] B. Sanz, I. Santos, C. Laorden, X. Ugarte-Pedrero, P. G. Bringas and G. Alvarez, "PUMA: Permission Usage to Detect Malware in Android," *Springer-Verlag,* 2013.

[4] A. Arora, S. Grag and S. K. Peddoju, "Malware Detection Using Network Traffic Analysis in Android Based Mobile Devices," *IEEE,* 2014.

[5] D. Arp, H. Gascon and K. Rieck, "DREBIN: Effective and Explainable Detection of Android Malware in Your Pocket," *Internet Society,* 2014.

[6] Z. Aung and W. Zaw, "Permission-Based Android Malware Detection," *International Journal of Scientific & Technology Research,* vol. 2, no. 3, 2013.

[7] T. Isohara, K. Takemori and A. Kubota, "Kernel-based Behavior Analysis for Android Malware Detection," *IEEE,* 2011.

[8] K. A. Talha, D. I. Alper and C. Aydin, "APK Auditor: Permission-based Android Malware Detection System," *Digital Investigation,* 2015.

[9] S. Liang and X. Du, "Permission-Combination-based Scheme for Android," *IEEE,* 2014.

[10] S. Y. Yerima, S. Sezer and I. Muttik, "High Accuracy Android Malware Detection using Ensemble Learning," *IET Information Security,* 2015.

[11] L. M. Acosta-Guzman, G. Aguilar-Torres and G. Gallegos-Garcia, "Network Activity Monitoring Against Malware in Android," *International Journal of Electrical and Computer Engineering,* 2016.

[12] H. A. Lashkari, A. F. Kadir, H. Gonzalez, K. F. Mbah and A. A. Ghorbani, "Towards a Network-Based Framework for Android Malware Detection and," *CIC,* 2014.

[13] Z. Liu, Y. Li, H. Yang and J. Qiu, "A Case Study on Key Technologies of Android Trojan," *IEEE,* 2014.

[14] A. Shabtai, U. Kanonov, Y. Elovici, C. Glezer and Y. Weiss, "Andromaly: A Behavioral Malware Detection Framework for Android Devices," *J Intell Information System,* 2012.

[15] W. Dong-Jie, M. Ching-Hao, W. Te-En, L. Hahn-Ming and W. Kuo-Ping, "DroidMat: Android Malware Detection through Manifest and API Calls Tracing," *IEEE,* 2012.

[16] Z. Yuan, Y. Lu, Z. Wang and Y. Xue, "Droid-Sec: Deep Learning in Android Malware Detection," *SIGCOMM,* 2014.

[17] A. Shabtai, L. Tenenboim-Chekina, D. Mimran, L. Rokach, B. Shapira and Y.

Elovici, "Mobile Malware Detection through Analysis of Deviations in Application Network Behavior," *Elsevier,* 2014.

[18] J. Li, L. Sun, Q. Yan, Z. Li, W. Srisa-an and H. Ye, "Significant Permission Identification for Machine Learning Based Android Malware Detection," *IEEE,* 2017.

[19] M. Grace, Y. Zhuo, Q. Zhang, S. Zuo and X. Jiang, "RiskRanker: Scalable and Accurate Zero-day Android Malware Detection," *ACM,* 2012.

[20] H. Gascon, F. Yamaguchi and D. Arp, "Structural Detection of Android Malware using Embedded Call Graphs," *ACM,* 2013.

[21] K. O. Elish, X. Shu, D. Yao, B. G. Ryder and X. Jiang, "Profiling User-Trigger Dependence for Android Malware Detection," *IEEE,* 2013.

[22] G. Dini, F. Martinelli, A. Saracino and D. Sgandurra, "MADAM: A Multi-Level Anomaly Detector for Android Malware," *IIT,* 2012.

[23] J. Sahs and L. Khan, "A Machine Learning Approach to Android Malware Detection," *IEEE,* 2012.

[24] S. Bhardwaj, P. Chauhan and R. Sharma, "Android Operating System," *International Journal of Engineering Technology and Management System,* 2013.

[25] B. N. Rupa, G. K. Mohan, J. S. Babu and T. H. Kim, "Test Report Generation Using JSON," *International Journal of Software Engineering and Its Applications,* 2015.

[26] Z. U. Haq, G. F. Khan and T. Hussain, "A Comprehesive Analysis of XML and JSON Web Technologies," *New Dovelopments in Circuits, System, Signal Processing, Communications and Computers,* 2012.

[27] R. A. Grimes, "TechNet," Microsoft, 2002. [Online]. Available: https://technet.microsoft.com/en-us/library/dd632947.aspx.

[28] J. Mason, "Advanced Encryption Standard (AES)," thebestvpn, 2017. [Online]. Available: https://thebestvpn.com/advanced-encryption-standard-aes/.

[29] F. Alhumaidan, "A Critical Analysis and Treatment of Important UML Diagram Enchancing Modeling Power," *Intelligent Information Management,* 2012.

[30] S. Lee, "Unified Modeling Language (UML) for Database System and Computer Applications," *International Journal of Database Theory and Application,* 2012.

[31] P. Kolte, T. Bhujbale and A. Chaware, "Web Portal Development using Programming Practices," 2012.

[32] W. Pierce, "Disadvantages and Advantages of Extreme Programming," 2016. [Online]. Available: https://atlaz.io/blog/disadvantages-and-advantages-of-extreme-programming/.