SECURITY IN ANDROID OS

Prof. Sumedh P. Ingale **Prof. Ankit R.Mune** **Prof. Aditya O. Sable**

Abstract- **Today's Smartphones are small personal computers with added services, because of it we say that next generations of operatingsystem will be on these handheld Smartphones and the OS of these Smartphones are similar to windows, IOS and android areshowing us the way to the future. Android OS has already gained significant popularity over its counterparts and in terms gainedmuch of market share. One of the reason behind this result and most important feature of Android is that it is open-source andDeveloper friendly so anyone could easily develop their own applications and publish them freely. This openness of androidbrings the developers and users a wide range of convenience but leads to some gaps in security. One of the major threat ofAndroid users is Malware infection via Android Applications which is targeting some loopholes in the architecture mainly on theend-users part. In this paper we presents the current state of Android OS its security mechanisms and their limitations.**

Keywords-**Android, Android Security, Architecture, Threats, Malware, Smartphone.**

I. INTRODUCTION

Android OS is a modern mobile platform OS that wasdesigned to be truly open. In the OS Android applicationsmake use of advanced hardware and software, as well as localand served data exposed through the platform to bringinnovation and value to consumers [1]. Android wasdeveloped by the Open Handset Alliance (which was visiblyled by Google), is OS based on Linux platform [2], It is anow widely used open source operating system for mobiledevices which provides a base OS, an application middlewarelayer, a Java software development kit (SDK) and a collectionof system applications .The widespread usage of Smartphone'sand with its increasing functionalities to meet newrequirements has made both industry and common consumersto rely on these handheld devices for their daily life routine .The most striking feature of Android OS is its openness ,Because of which anybody can publish their applicationsfreely on the android market . This openness brings largenumbers of developers which use these platform, but with thisopenness comes some risk in likes that user is may downloadand use a malicious software made hackers causing harm totheir privacy .Thus we need to study of the SecurityMechanisms for Android and on the way make it simple anduser understandable making the user And make the end-useraware of areas where he has to be careful.

II. ANDROID OVERVIEW

The Android operating system began its release with Android beta in November 2007. It was designed with keeping in mind, with both the developers and the end user, because of which the for the developer can easily develop its idea into and application and user is given some visibility over applications work. The first complete version Android 1.0 was released in September 2008. Android is under ongoing development by Google and the Open Handset Alliance (OHA) is a consortium of 84 firms, and has seen a number of updates to its base operating system since its initial release. Since April 2009, Android versions have been developed under a confectionery-themed code name with significant feat improvement over time to time is listed below

Cupcake (1.5)
Soft-keyboard with text-prediction, Record/watch videosBluetooth A2DP, AVRCP support

Donut (1.6)
Turn-by-turn navigation , Gesture framework

Éclair (2.0–2.1)
HTML, Microsoft Exchange support, Bluetooth 2.1,Digitalzoom, Live Wallpapers,Updated UI

Froyo (2.2–2.2.3)
Speed improvements, JIT implementation, Applicationsinstallation to the expandable memory, Upload file support inthe browser, USB Tethering, Animated GIF

Gingerbread (2.3–2.3.7)
Updated UI, Improved copy/paste, Improved keyboard easeof use, Improved power management, Near FieldCommunication support, Native VoIP/SIP support, Video callsupport, Social networking features.

Honeycomb (3.0–3.2.6)

Multi core support, Media/Picture transport protocol,Updated 3D UI, Private browsing, Better tablet support, HTTP Live streaming, System-wide Clipboard.

Ice Cream Sandwich (4.0–4.0.4)

Better voice recognition (dictating/Voice typing), Facialrecognition (Face Unlock),UI use Hardware acceleration, Webbrowser, allows up to 16 tabs, Updated launcher(customizable),Android Beam app to exchange data throughNFC

Jelly Bean (4.1–4.3)

Voice Search, Speed enhancements, Google Now , Cameraapp improvements, Accessibility: gesture mode, app stacknavigation to define a parent activity in manifest for deepnavigation, MediaActionSound class to make sounds likewhen a camera takes a photonic supports large payloads overBluetooth, WIFI/WIFI-Direct service discovery, Large,detailed multi-action notifications, enable Braille externalkeyboards.

KitKat (4.4)

Enhanced notification access, Screen recording, NewTranslucent system UI, System-wide settings for closedcaptioning, Performance improvements[3]

Android "Lollipop" (5.0-5.1.1)

Support for 64-bit CPUs, Android Runtime (ART) with ahead-of-time (AOT) compilation and improved garbagecollection (GC), Replacing Dalvik that combines bytecode interpretation with trace-based just-in-time (JIT)compilation,

Android "Marshmallow"(6.0-6.0.1)

Post-install/run-time permission requests, that is APPpermissions now granted individually at run-time, not all-ornothingat install time, USB-C support, Larger Applicationfolders with multiple pages.

Android "Nougat"(7.0-7.1.2)

Improved Doze functionality, which aims to prolong batterylife, Improvements to file browser, More Quick Settingsoptions, Battery usage alerts,

Android "Oreo" (8.0-8.1)

Project Treble which is the biggest change to the foundationsof Android till date: a modular architecture that makes it easier& faster for hardware makers to deliver Android updates,Picture-in-picture support, Auto fill framework updates

Android "Pie" (9.0)

Rounded corners across the UI, A new optional gesturebasedsystem interface, allowing users to navigate the OSusing swipes more often than the traditional UI [4]

III. STRUCTURE OF ANDROID

3.1 ARCHITECTURE:

Android is an open source, Linux-based software stackcreated for a wide array of devices and form factors. Thefollowing diagram shows various layers in the Linux basedstack.

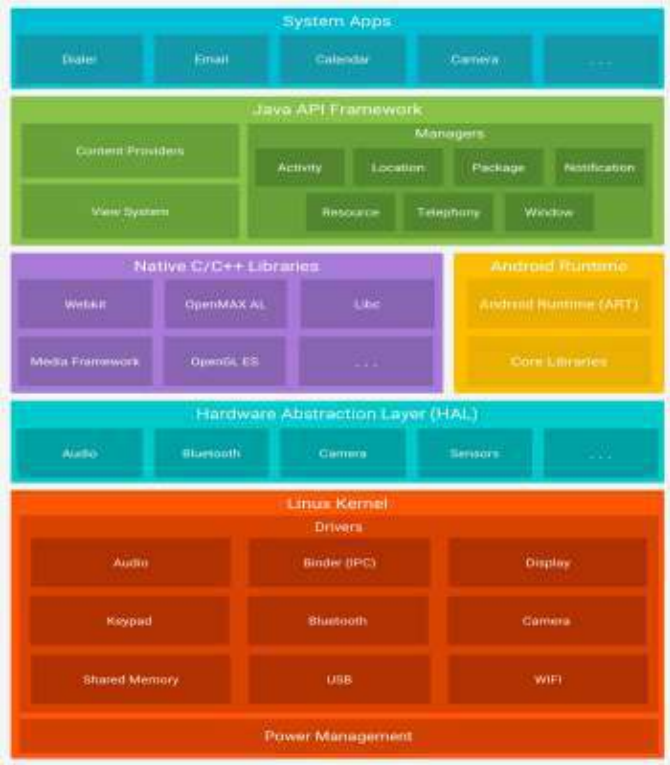

Fig .android architecture (from Top to Bottom are Application, Application Framework, Libraries & Android Runtime and Linux Kernel) [6]

1) Linux Kernel: AT the foundation of the Android platform is the Linux kernel. An example of this is the Android Runtime (ART) relies on the Linux kernel for underlying functionalities such as threading and low-level memory management. Use of the Linux kernel allows Android OS to take advantage of key security features of widely used Linux and allows device manufacturers ease to develop hardware drivers for already well-known kernel.

2) **Hardware abstraction layer (HAL):** The HAL provides standard interfaces that links device hardware capabilities to

the higher-level Java API framework. It consists of multiple library

modules, each of which implements an interface for a specific type of hardware component, such as the camera or Bluetooth module. When a framework API makes a call to access device hardware, the Android system loads the library module for that hardware component.

3) **Android Runtime: Each** app runs in its own process and with its own instance of the Android Runtime (ART) is a managed Runtime environment used by the app. ART is written to run multiple virtual machines on low-memorydevices by executing DEX files, a byte code format designedespecially for Android that's optimized for minimal memoryfootprint. Build tool chains, such as Jack, compile Javasources into DEX byte code, it provide Ahead-of-time (AOT)and just-in-time (JIT) compilation and Optimized garbagecollection (GC) plus various features for app development anddebugging. Prior to Android version 5.0 (API level 21), Inearly versions Dalvik was the Android runtime. If your appruns better on ART, then it should work on Dalvik also,but the opposite may not be always true.

4) Native C/C++ Libraries: It is a set of tools that allows one to use C and C++ code with Android, and provides platform libraries that can be used to manage native activities and access the physical device components, such as sensors and touch input

5) Java API Framework: The complete feature-set of the Android OS is available to you through APIs written in the Java lang. These APIs forms the building blocks one needs to create android apps by simplifying the reuse of core, modular system components and services, which include the following:

- Rich and extensible View System can be use to build Application UI, including lists, grids, text boxes, buttons, and even an embeddable web browser.
- Resource Manager, which is providing access to noncode resources such as localized strings, graphics, and layout files.
- Notification Manager that enables all Apps to display alerts in the status bar.
- Activity Manager that manages the lifecycle of Apps and provides a common navigation back stack.
- Content Providers that enable Apps to access data from other Apps, such as the Contacts app, or to share their own data
- Developers have a full access to the same framework APIs that Android system apps use.

6) **System Apps:** The OS comes with a group of core apps such as SMS messaging, calendars, email, internet browser, contacts, and more. Apps included with the platform have no special status among the apps the user chooses to install. So a third-

party app can become the user's default web browser, SMS messenger, or even the default keyboard.

3.2 *Security Mechanism In Android*

3.2.1 Sandboxing:

The Android platform takes advantage ofthe Linux user-based protection as a means of identifying andisolating application resources. The kernel implements aprivilege separation model i.e. sandbox when it comes toexecuting applications. That is like on an UNIX system, theAndroid OS requires every application to run with its own useridentifier (uid) and group identifier (gid). Parts of the systemarchitecture themselves are separated in this way. This is doneensures that an applications or processes have no permissionsto access other processes or applications. same the Linuxandroid kernel implements privilege separation, it is one of thecore design features of Android. The idea behind this designis to ensure that no application can read or write to code ordata from other applications, device user, or the operatingsystem itself. This prevents every Application from accessingthe private information of the Other application. Unauthorizedaccess to hardware components like GPS, Camera or networkcommunication can be prevented using the sandboxingmechanism. As in sandboxing two processes have run in theirown sandbox, the only way they can communicate with eachother is to explicitly generating request permission to accessdata thus reassuring privilege separation [9]

3.2.2. Application

Permission Mechanism: Application onAndroid run in its application Sandbox. because of which anAndroid application can only access a certain range of systemresources. While installing a new Application, The OSprompts user with the all declared permissions which areRequired by the application. The user needs to grant therequested permissions, For smooth running of installed app. Inearly versions the only way for not granting the permissions isnot to install the Application .Once the permissions are grantedand the application is installed, it cannot request for furtherpermissions and in early version the user cannot change thesepermissions, and only way remove permission is uninstallingthe application from the device. Granting permission is veryimportant part of installing the app because the result ofgranting permission to an application, the

application canhave unrestricted access the corresponding resources. Thesystem manages Android application access to resources thatif used incorrectly or maliciously, could adversely impact theuser experience, privacy, the network or data on the device.Some API are considered to be very important with regards touser privacy.These protected APIs include: Location GPS, Bluetoothfunctions, Telephony functions, Camera functions, SMS/MMSfunctions, Network/data connections.These resources are only accessible through the operatingsystem. To make use of the protected APIs on the device, anapplication must define the capabilities it needs in its manifest.When preparing to install an application, the system displays adialog to the user that indicates the permissions requested andasks whether to continue the installation. If the user continuewith the installation, the system accepts that the user hasgranted all of the requested permissions. In old versions usercan not grant or deny individual permissions to these API, Butin newer version of Android the ask for individual permissionbut once the user granted or denied the requested permissionsits some tedious task to again revoke or grant thosepermissions. Once granted, the permissions access are grantedto the application as long as it is installed or permission isrevoked. To avoid user confusion, the system does not notifythe user again, of the permissions granted to the application,and applications that are included in the core operating systemor bundled by an OEM do not request permissions from theuser. In newer version user can grant or revoke the individualpermission to the Application from the setting menu.

3.2.3 Application Code Signing:

For any Application to run on the Android operating system it must be signed. AndroidOS requires that all APKs to be digitally signed with acertificate beforehand to their installation. Android uses these certificate of individual developers in order to identify them and thus establishing a trust relationships among the various applications running in the OS. The operating system will never allow an unsigned application to install on system. Any third party certification authority to sign the certificate is not required, and Android will happily run any application which has been signed with a self-signed certificate. A publickey certificate, also known as a digital certificate or an identity certificate, contains the public key of the public/private key pair, as well as some other metadata identifying the owner of the key attribute (for example, name and location). The owner of the certificate holds the corresponding private key. Every app must use the same certificate throughout its lifespan. In order for users to be able to install new versions as updates to the App.

IV. VULNERABILITIES IN ANDROID SECURITY

4.1. Resource Draining:

Android applications are using devices disk space or memory. Any malicious application can use more memory, disk and CPU hogging is also possible. Undetected malware can drain the resource by remaining hidden. By using host based IDS we can detect the malwares that are using more disk space or memory.

4.2. Reading Contents:

Applications in android can read the data or contents of Smartphone by implicitly or explicitly gaining the permission and on wireless communication eavesdropping can be done by attackers remotely.

4.3. Attack Through Installed Application:

Android application ask for permission to data, phone ,message , contacts, network etc. to use it at the time ofinstallation. These types of attack have high impact [8]

V. SOLUTIONS FOR USERS

5.1 Avoid Rooting - Among Android users it is popular to root their phone, by which it allows users to bypass android security mechanisms and unlock their phones, gaining full access to settings and features that are often blocked. It should be avoided as the Android security on users phone and ends users Android support warranty, preventing you from getting Android support from Google or the manufacturer.

5.2 Avoid App loading from 3rd party -Some Android users choose to side load apps and programs that the Google Play Store does not offer and for which Google does not provide Android support. These apps usually come from 3rd party unofficial Internet sources. Installing these apps is a major risk as many of them may contain malware. These Side loaded apps account for most attacks on Android security.

5.3 Educate User about Permissions -Every App you run on Android needs permissions to execute its features. Knowing what permissions of users apps need and when an app is asking for permissions it should not. This will prevent

Android – security risks such as apps accessing personal information and sending it elsewhere

5.4 Use of Default Android Browser or Google Chrome - As these are specially designed for Android the browser and Google Chrome provides the best Android security while accessing the web. Studies show that third-party browsers are more exploitable and can be arising risks to Android security.

5.5 Operating System Update -Cell phone manufacturer's are not always releasing updates or providing Android support regularly. However, always advisable install new updates when they are available. Each new version of Android plus its apps includes new, more powerful Android security features.

5.6 Backing up Data - It is advisable to regularly backing up user data on trusted backup services provided by google such as google drive to keep user data safe at remote location . [9]

IV. CONCLUSION

The growing popularity of smart phone usage withAndroid OS and its wide adoption in the market has given riseto various important security concerns. This paper analyzesthe current state of the OS and its security mechanism.Discussed security mechanisms and there limitation.Malwares or malicious Apps are the major threat to theAndroid user ,The best way to avoid them is to make useraware of the security mechanism and how to use them forusers benefit and stop malwares at their installation phase.

REFERENCES

1. Android Open Source Project. Android Security Overviewhttp://source.android.com/devices/tech/security/index.html
2. "Android", at: https://www. android.com/.
3. "Android versions comparison "at :http://socialcompare.com/en/comparison/android-versions-comparison.
4. "Android version history" at:http://wikipedia.org/wiki/Android_version_history.
5. Android versions" at :http://www.kinveyhttp://source.android.com/devices/tech/security/.com/blog/2586/android-version-history-a-visual-timeline.
6. Platform Architecture" at:https://developer.android.com/guide/platform/
7. Android Reference: Application Fundamentals-Components," available at:http://developer.android.com/guide/topics/fundamentals.html.
8. Chetan C.Kotkar, Pravin Game, "Exploring SecurityMechanisms to Android Device" , International Journal of Advanced Computer Research, Volume-3 Number-4 Issue 13 December-2013.
9. AhamedShibly ," Android Operating System:Architecture, Security Challenges and Solutions" 7th International Symposium 2017 (IntSym2017)-SEUSL - 07th & 08th December 2017