Imperial Journal of Interdisciplinary Research (IJIR)
Vol.2, Issue-1 , 2016
ISSN : 2454-1362 , www.onlinejournal.in

Sharing Files via Wi-Fi with advanced Security features

Aamir N. Khan
Dept. of Computer Engineering.
Sinhgad college of Engineering
Vadgaon(BK), Pune

Kunal S. Shah
Ramdeobaba College of Engineering,
Nagpur, India.

Abstract

A Wi-Fi-enabled device can connect to the Internet when within range of a wireless network which is configured to license this. The coverage of one or more (interconnected) access points called hotspots—can extend from an area as small as a few rooms to as large as many square-kilometer. Coverage in the larger area may require a group of access points with overlapping coverage. Wi-Fi provides service in private homes, businesses, as well as in public spaces at Wi-Fi hotspots set up either free-of-charge or commercially, often using a captive portal webpage for access. Organizations and businesses, such as airports, hotels, and restaurants, often provide free-use hotspots to attract customers. Enthusiasts or authorities who wish to provide services or even to promote business in selected areas sometimes provide free Wi-Fi access. A secure digital (SD) card is a tiny memory card use to make storage portable among various devices such as car navigation, cellular phones, digital cameras, personal computers. An SD card features high data transfer rate and low battery consumption, both principal considerations for portable devices. It uses flash memory to provide non-volatile storage which means power storage is not required to retrieve stored data. SD card provides encryption capabilities for protected content to ensure distribution of copyrighted material.

1. Introduction

A secure digital (SD) card is a tiny memory card use to make storage portable among various devices such as car navigation, cellular phones, digital cameras, personal computers. An SD card features high data transfer rate and low battery consumption, both primary considerations for portable devices. It uses flash memory to provide non-volatile storage which means power storage is not required to retrieve stored data. SD card provides encryption capabilities for protected content to ensure distribution of copyrighted material. Laptops that have a cellular modem card can also act as mobile Internet Wi-Fi access points. Wi-Fi also connects places that normally don't have network access, such as kitchens and garden sheds.

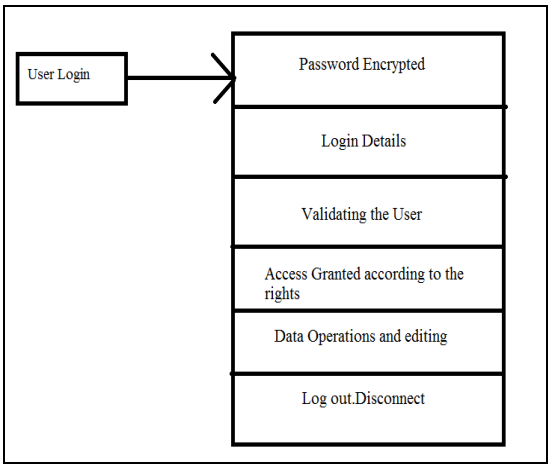

Figure 1 : Module Structure

The development of this approach has many hurdles viz. (1)Total there are many obstructions in implementing the RSA algorithm (2)Based on the Android Kernel a lot of operations get difficult to perform.

2. Literature Survey

The world has become increasingly mobile and wireless networks allows users to work and move freely, therefore wireless technologies, nowadays, are more popular that wired or fixed networks .A hotspot often operate Wi-Fi technology via router, offering

internet access. Free hotspot generally offer free access to menu or purchase list also providing payment systems like PayPal, or via credit card, or work public network in which authentication and verification features are turned off. The term poisoned hotspot or rogue hotspot refers to a malicious individual who sniff the data sent by user on a free hot spot including decipher passwords, and login names. We did a literature study focusing on Wi Fi technology, and its comparison with other technologies. Android OS is designed for smartphones. It provides a sandboxed application execution ambiance. A Linux system that is customized interacts with the phone hardware ensuring the required task to be performed. The application API and the Binder middleware runs on the top of Linux. To state as a fact or rather reduce to bare bones, an applications only interface to the phone is through these APIs. A typical android phone is preinstalled with utility applications such as phone book and dialer. Applications interrelate with each other through inter-process communication accounting to numerous of them available. Through SQL like interfaces persistent content provider data stores are queried. RPC and callback interfaces that applications use to trigger actions or access data that background services provide. Lastly user interface activities receive named action signals from the system and other applications. Binder acts as a mediation point for all IPC. Access to system resources (e.g., GPS receivers, text messaging, phone services, and the Internet), data (e.g., address books, email) and IPC is governed by authorizations assigned at install time. The permissions requested by the application and the permissions required to access the application's interfaces/data are defined in its manifest file. To simplify, an application is allowed to access a resource or interface if the required authorization allows it. Permission consignment—and indirectly the security policy for the phone—is largely delegated to the phone's owner: the user is offered a screen listing the permissions an application requests at install time, which he can accept or reject. All related information is gathered from websites, e-books, online articles and journals, other internet sources and library books. There was some difficulty about finding particular information (like frequency, upstream and downstream speed). We congregated all the required information, and checked with at least two sources, assuring that it is true information. The literature study did not analyze technologies like Bluetooth, Near Field Communication (NFC), and Radio frequency identification (RFID). There are variety of general purpose and commonly used mobile applications developed for sharing of files among different smartphones. This range encompasses of a limited functions that are dominant on the sending end. The functionality of our software is better and different than that of other available applications in market.

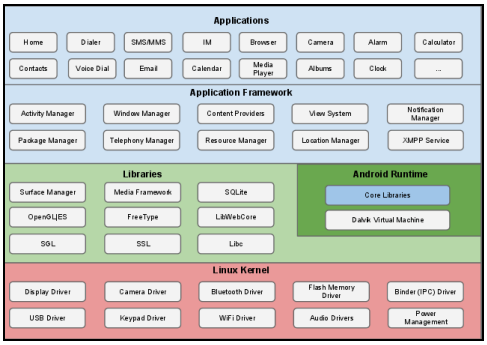

Figure 2 : The android system architecture

3. Proposed Methodology

The given Fig.3 shows the workflow of the application. In the first step, the owner puts the files he wants to share in a folder generated by the application. It depends on the owner to let access to the entire device or that particular folder which he can select dynamically on the server side of the application . The user starts the application he gets a screen where he needs to input the credentials that he possesses, provided by the owner. According to the privilege he is granted he will be able to go through the RSA encryption allowing him to access the files there available in the folder/device. If the user wants to download a particular file from the possessor he will be able to do so. The downloaded file will get converted into a .zip format in the user's device which will be again password protected. Showing the cryptography used and enhancing the security and the privacy of the owner.

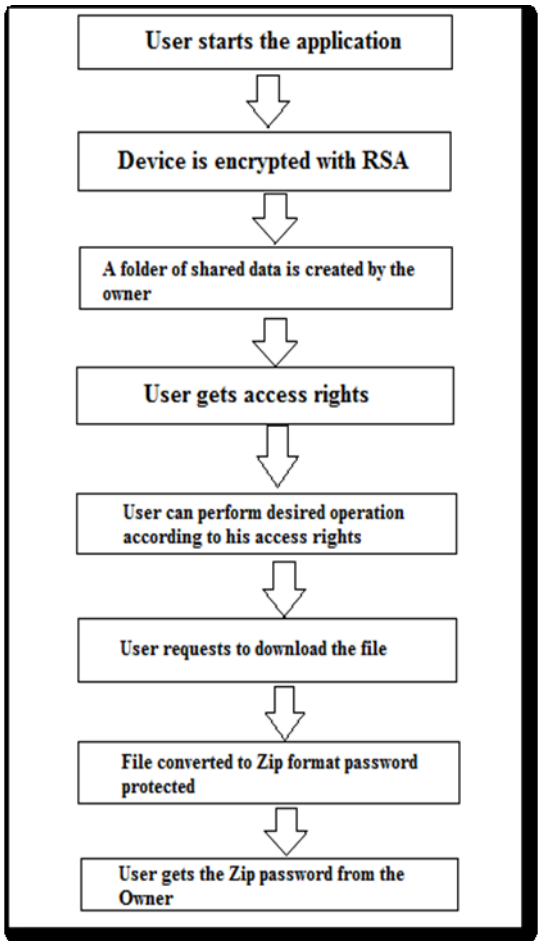

Figure 3 : Workflow of the Proposed Methodology

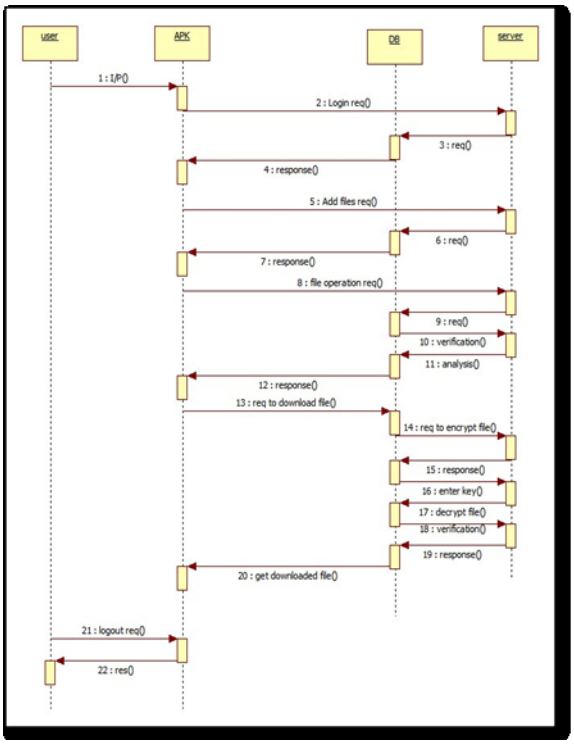

Figure 4 : Sequence diagram

4. Experimental Setup

What is Cryptography? It is something that is very commonly heard about in our time. Cryptography comprises of various technique that ensures security to the information under consideration. It can be simply stated as a science of providing security. In cryptography, encryption is the course of encoding messages or data in such a way that only authorized parties can read it. Encryption does not of itself prevent interception, but denies the message content to the interceptor. In an encryption scheme, the message or information, referred to as plaintext, is encrypted using an encryption algorithm, generating a particular cryptogram text that can only be read if decrypted in a proper manner. For technical reasons, an encryption scheme usually uses a pseudo-random encryption key generated by an algorithm which can be any algorithm

It is in principle possible to decrypt the message without keeping the key, but, for a well-designed encryption scheme, large computational resources and skill are required. An authorized recipient can easily decrypt the message with the key provided by the originator to recipients that need not to be the same, but not to unsanctioned interceptors. RSA Algorithm: User Defined Library in Java. RSA: (Rivest, Shamir, Aldeman). In 1997 RSA came into being, it is an encryption/ decryption and authentication system, an algorithm by Ron Rivest, Adi Shamir, and Leonard Adleman. RSA uses public and private key cryptosystem, which is also known as public-key cryptosystems (Public Key Encryption). RSA is normally used for protected data transmission. A user of RSA creates product of two enormous prime numbers, along with an auxiliary value, as public key. The prime numbers given to algorithm kept as secret. The public key is used to encrypt a message whereas private key is used to decrypt a message. The question that arises iteratively is “Does RSA works efficiently?” or “Is it strong enough to be trusted? RSA might be a very old cryptographic technique but it is still widely used because of its utility. Experts, learners and scholars are of the opinion that it is the best cryptographic algorithm as it is very difficult to break. Even a combination of million values will be negligible for breaking the encryption. The data or information thus

remains safe when encrypted. The RSA algorithm is stated below. It ensures the computation of the public key through which the private key is further computed. For an encryption done using a particular public key only the private can be used to remove the encryption and access the information that is available.

==========

Step 1: Start
Step 2: Pick up any two random prime numbers
y = 3 and z = 11
Step 3: Compute the value for 'n'
n = RSA.n_value (RSA_P, RSA_Q);
n = y * z = 3 * 11 = 33
Step 4: Compute the value for? (n)
? (n) = (y – 1) * (z -1) = 2 * 10 = 20
Int phi = RSA.cal_phi (RSA_Y, RSA_Z);
Step 5: Choose e such that 1 < e <? (n) And e and n are co-prime. Let e = 7
Step 6: Compute a value for d such that (d * e) % ? (n) = 1. D = 3
Public key is (e, n) => (7, 33)
Private Key is (d, n) => (3, 33)
Step 7: Stop.
Let M, is plain text (message), M= 2.
Encryption of M is: C = Me % n.
c = "" + RSA.BigMod (ar[i], RSA_E, n);
Cipher text is, C = 27 % 33.
C = 29.
Decryption happens to be of C as: M = Cd % n.
dc = dc + (char) RSA.BigMod(Integer.parseIntI , d, n);
Plain text (message), M= 293 % 33.
M= 2

==========

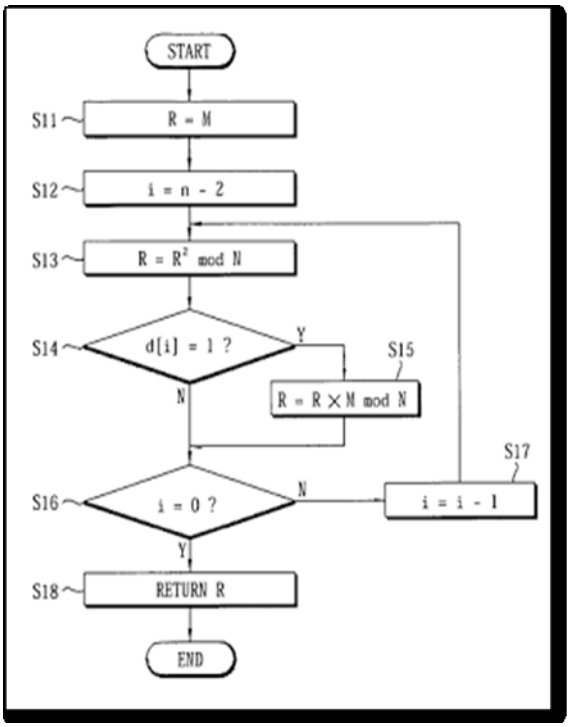

Figure 5 : Flowchart for RSA

5. Results and Discussion

Dripping Information to Logs:Android provides integrated logging via the Log API, which can displayed with the "logcat" command. While logcat is a debugging tool, applications with the Read_Logs permission can read these log messages. The Android citations for this permission indicates that "[the logs] can contain marginally cloistered information about what is happening on the device, but should never comprehend the user's sequestered information." We looked for data flows from phone identifier and location APIs to the Android logging interface and found the following. Sequestered information is written to Android's general logging interface. Frequently, URLs containing this private information are logged just before a network connection is made. Thus, the Read_Logs permission permits access to sequestered information. SD-card Use of application that has access to read or write data on the SD-card can read or write any other application's data on the SD-card. Sampling these applications, we found a few unexpected uses. For example, the com/ tap-joy ad library (used by com.jnj.mocospace.android) determines the free space available on the SD-card. Another application (com.rent) obtains a URL from a file named connRentInfo.dat at the root of the SD-card.

Sr. No.	Risk	Category	Probability	Impact	Risk Management
1	Huge quantity of memory required by the system.	PS	11%	4	Lessen the size of the specification engine so that overhead falls.
2	User may not straightforwardly familiarize to the system interface.	CC	38%	2	Redesign User Interface to make it more user-friendly.
3	Lack of sufficient hardware resources	CC	44%	2	Upgrade resources in order to meet minimum specified hardware requirements.
4	Too much time required in the detection of attacks	PD	30%	3	Use improved specification engine
5	Expected functionalities not supported by the selected technology.	DE	15%	1	This risk has to be mitigated by doing enough research prior to technology selection

6. Future Scope

Smartphones are rapidly becoming a dominant computing platform.. In this paper, Security thus is provided using Zip file conversion on the data at outer level. On the inner level of our software security enhancement will be done using the different profiles we have created and the access rights we have granted to the users of that profiles .The software will be developed using android application development for the data transfer between users. The software we developed allows users to share data and receive the files the want as per their requirements. This will prove of great utility for the users of various domains as it is a common application. This is very beneficial for new era of technology, as this software eases up the data sharing process .This provides very interactive interface for the users. The modules of the project can be further used for development of many future projects. Improving the interface and making multiple users to access at once can make it a top-notch application.

References

1. Lextrait, Vincent (July 2010). "The Programming Languages Beacon, v10.3". Retrieved 5 September 2010.
2. Milinkovich, Mike. "IBM and Eclipse: A Decade of Software Innovation". *Building a Smarter Planet*. Retrieved 3 November 2011
3. R.H.Campbell, J.AL Muhtadi,P.Naldurg,G.sampemane,and M.d.Mickunas."Towards security and privacy for pervasive computing" In ISSS,pages 1-16 ,2002.
4. Khan, S ; Nauman , M; Othman ,A.T. ;Musa ,S . "How secure is your SD-card; Analyssis of smartphone security mechanisms" IEEE, cyber security, Cyber Warfare and digital forensic(cyber sec)2012.

5. Poonam N .Railkar , Parikshit N Mahalle, ”Activity modelling and threat proactive system in smartphone” International journal of computer application [0975-887] volume 70 NO 25,May 2013.
6. BURNS, J. Developing Secure Mobile Applications for Android. iSEC Partners, October2008. http://www.isecpartners.com/files/iSEC_Securing_Android_Apps.pdf.
7. OCTEAU, D., ENCK, W., AND MCDANIEL, P. The ded Decompiler. Tech. Rep. NAS-TR-0140-2010, Network and Security Research Center, Department of Computer Science and Engineering,Pennsylvania State University, University Park, PA, USA, Sept 2010.
8. JOHNS, T. Securing Android LVL Applications. http://androiddevelopers.blogspot.com/2010/09/securing android-lvl-applications.html, 2010
9. ENCK, W., ONGTANG, M., AND MCDANIEL, P. Understanding Android Security. *IEEE Security & Privacy Magazine* (January/February 2009), [50–57].