AN ADHOC NETWORK BASED MOBILE ANDROID REMOTE CONTROLLER (REMODROID)

M. D. K. N Senevirathne[1], P. H. V. Vishwani[2], K. T. S. Kasthuriarachchi[3]

Department of Information Technology, Sri Lanka Institute of Information Technology, Sri Lanka.
Email: [1]niroshana1988@gmail.com, [2] vindyavishwani@gmail.com, [3]sanvitha.k@sliit.lk

ABSTRACT

This research is based on an Android Mobile Based AD-hoc (Mesh Network) Network Remote Controller (RemoDroid) specially designed for students and lectures to perform the teaching and learning process efficient using Ad-Hoc wireless mesh network (MANET) and Android smart phones. The innovativeness of the RemoDroid application is that it can be used in different environments and different situation such as Tsunami, Flooding, Fire and the situation where the network or communication infrastructure is unavailable such as war, boom explosion, and military communications. The research process is mainly around on BATMAN mobile Ad-hoc network. MANET routing, MANET Communication/Messaging and MANET File Sharing is discussing throughout the research study. In order to perform multi-hop IP-based routing, the routing table of a node must contain a link-local gateway for each host or network route. BATMAN chooses the most reliable route upon the next hop routing decision of individual nodes. The Android application starts the communication based on the options; whether the user chooses for peer to peer file sharing or peer to peer messaging. This approach has shown in practice that it is reliable because it ensures highly adaptive and loop-free routing while causing only low processing and traffic cost.

***Keywords**:* Mobile Ad-Hoc Network, MANET, Android Platform, Ad-Hoc File Sharing, Multi Peer Communication, Wireless Mobile Connectivity, Communication Without Network Infrastructure.

1. INTRODUCTION

Today world, Mobile learning is considered as an extension of E Learning. Without development of education system than the traditional system of any society in the Sri Lanka is not possible. Education is an essential human activity. At this time of mentioning education, classrooms are recalled as this typical classroom is familiar to us the traditional way to transfer knowledge to student with the various technological advancements in different areas like Information Technology [1], Mobile Technology and lot if fields are affected. As a result, learning is no longer confined to the classroom. To convey or deliver knowledge to students, lectures are not the only method or the only way. Mobile based technologies, web based technologies are more powerful and those are served several new opportunities for the development of education technologies. Consequently, Mobile learning with an android (M Learning) came to exist as new learning methodology and research field.

RemoDroid is an Android based application run/interconnected on Open source base platforms such as Linux/Android and consists with highly efficient file sharing, chatting application without any internet or GSM network. Message and file sharing for exchange the information's, ideas between teachers/lecturer s and students, Thus, all functions are sending wirelessly from mobile to mobile (P2P) in a mesh network [2, 3].

2. METHODOLOGY

A MANET is generally implemented in the Linux platform. Since the Android Operating System kernel is based on the Linux kernel, Cross compilation of Linux components is possible to a great extent. [4,5] Most of the MANET implementations available are developed as a kernel module for the Linux kernel and therefore in order to create a fully cross-compatible network between Android devices and Linux Devices, a compilation of a similar kernel module was required. The other option would be to create a fully compatible Application Layer Implementation

2.1. Procedure for Analysis

The RemoDroid study mainly focused on BATMAN protocol deployment on Android, Wireless LAN on Android, Messaging Standard, Android service controller and Use of Android as the Operating System for RemoDroid [6].

2.1.1. System Model

MANET usage is most profitable and fastest in situations where users are in need to share resources. File sharing and peer messaging for number of devices can be done through gateways as shown in Figure 1. Those gateways are allowed to other nodes to communicate according to the user inputs. The basic target behind RemoDroid is to make all the file sharing and messaging information available to all the nodes within the MANET as much as closer to the requesting point.

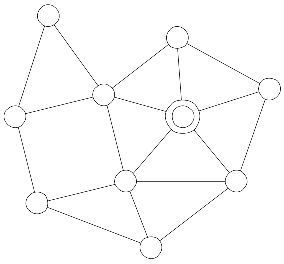


Figure 1: System Model (Communication Gateway)

RemoDroid protocol is implemented to run on top of Better Approach to Mobile Ad Hoc Networking (BATMAN) [7] protocol which handles the network routing. BATMAN is a proactive routing protocol which is a more developed solution [8] of the Optimized Link State Routing Protocol (OLSR). The strategy of BATMAN is to determine for each destination in the mesh one single-hop neighbor, which can be utilized as best gateway to communicate with the destination node. In order to perform multi-hop IP-based routing, the routing table of a node must contain a link-local gateway for each host or network route. To learn about the best next hop for each destination is the main objective of the BATMAN algorithm. Therefore, the need to find out or calculate the complete route is eliminated which makes a very fast and efficient implementation possible unlike other reactive and proactive routing protocols. Routing decisions are based on the knowledge about the existence or lack of information. BATMAN chooses the most reliable route upon the next hop routing decision of individual nodes. This approach has shown in practice that it is reliable because it ensures highly adaptive and loop-free routing while causing only low processing and traffic cost.

Unlike the other routing protocols implemented, such as Ad Hoc On-Demand Distance Vector Routing (AODV), BATMAN is able to provide us the cross compatibility among ARM and x86/x64 architectures [9,10,11,12]. Since mobile devices operate on the ARM architecture and personal computers operate on the x86/x64 architecture; the use of BATMAN as the routing protocol has allowed RemoDroid to be implemented on a wide variety of devices. Also BATMAN routing protocol operates faster comparatively since it makes use of Link Access layer information. In order to ensure cross compatibility, we inserted the same BATMAN protocol version as a kernel module to the Linux kernel of both Android and Linux platforms. The Android Operating System (OS) disables the use of Ad Hoc networks by default. Therefore, a custom OS; CyanogenMod version 7.2 was used in the Android platform as it supports Wi-Fi Ad Hoc networks visibility.

2.1.2. RemoDroid Single Peer Architecture

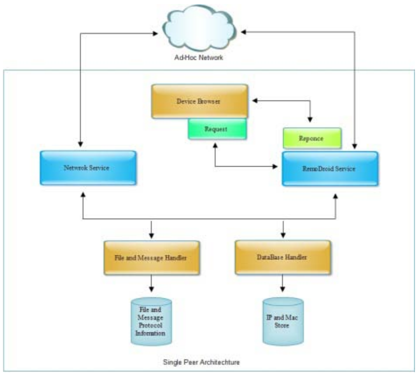


Figure 2: RemoDroid single peer architecture

The overall single peer architecture of RemoDroid illustrated in Figure 2 consists of a RemoDroid Service, which is responsible for managing the local Request and managing requests from the Android Application, a Network Service, which is responsible for managing external requests from the peers of the MANET, an Database Handler which manages the IP address and its own Mac address of the each and every Peers which are connected with the MANET.

2.1.3. RemoDroid Android Service Controller

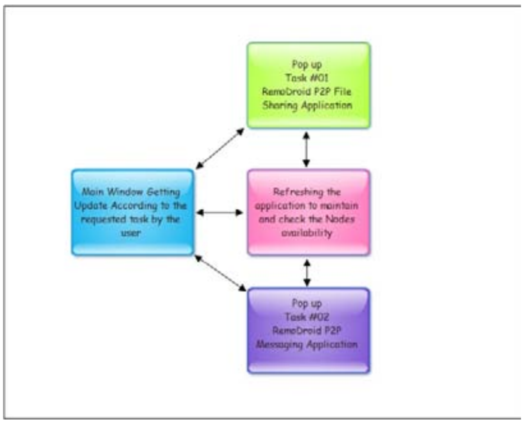

Pop up
Task #01
RemoDroid P2P File Sharing Application
Main Window Getting Update According to the requested task by the user
Refreshing the application to maintain and check the Nodes availability
Pop up
Task #02
RemoDroid P2P Messaging Application

Figure 3: Android service controller

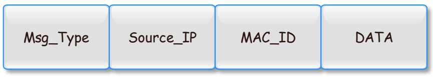
As shown in Figure 3, all the popup is linked to the main window. Each popup suffices a task each. RemoDroid Chat GUI is the control panel that is supplied to the host device user. The user is supplied with high level controls of RemoDroid.

Through the main window following accesses are granted Chat Application access and file sharing access as well as User can discover for active and visible peers to communicate, Can move to device settings for renaming to user profile such as host name, Can be choosing users/peers according to user's choice.

All these user access changes are coded parallel with backend RemoDroid services and all the users/peers data's are stored in SQL lite data base.

2.1.4. RemoDroid Messaging Standards

The RemoDroid system has its own messaging standard protocol in which the messages are not broadcasted over the network but propagated. This enables the devices to consume less power (low battery darning and OS utilization) while providing a way for devices to identify the source of the message and the type of request sent. All the RemoDroid Network Messages have the format and fields as shown in Figure 4.



Msg_Type	Source_IP	MAC_ID	DATA

Figure 4: RemoDroid messaging standards

Reqest_ID – This field is a user defined Name set for the service. This can contain up to 15 ASCII characters.

Msg_Type – This field is used to define which type of message is being sent. This field is a single ASCII character denoting the number of the message type. (Its acting as a identifications for what should need to done by in the user end)

Source_IP – This field is used to indicate the Source IP of the device which initiates the particular request.

Mac_ID – This field is used to indicate the Mac ID of the device which initiates the particular request. (IP address of the device and Mac ID of the will be compare to maintain reliability and integrity of the device. It will overcome the device mismatch connectivity and device conflicts)

Data – Any Data related to the Message type is included in this field.

3. RESULTS

Testing of the RemoDroid Messaging System was conducted using live devices in a controlled environment such as a class room and was conducted to observe MANET Performance when Peer to Messaging while RemoDroid is deployed.

The test application was installed in all the mobile nodes. Nodes were situated in a geographically challenging scenario comprising of height deference's lack line of sight and large distances which enabled to construct a multi-hop MANET for the testing purpose RemoDroid used four Peers and single Device used as a master peer (Lecture) and other three peers acting as students (child peers). The Master peer initiates the chat conversation and tested with child and check which child peers received the conversation in the multicast environment and non-multicast environment.

The main objective of the test was to measure the response time for the broadcast message for allowing the file sharing and system availability while it operating. The application measured the time taken from the instance where the message request and the time taken to broadcast the messaging for file sharing within active peer.

Following are the device Specification of mobile phones that we used for the testing process.

Table 1: Mobile device (peers) specifications

Mobile Device #01 (Master/Lecturer Peer)	
Mobile Model	Samsung Galaxy Mini S5570
Processor	600 MHz ARMv6
RAM	384MB
Wireless Module	Wi-Fi 802.11 b/g/n

3.1. Test Case 1

Here we used Android mobile Device #02,#03, and #04 as the nodes. We have disconnected the Mobile Device #01 from the network and broadcast the messages to other alive peers.

3.2. Test Case 1

Here we used Android Mobile #02, #03, and Mobile Device #04 as the nodes. We have disconnected the Mobile Device #01 from the network and broadcast the messages to other alive peers.

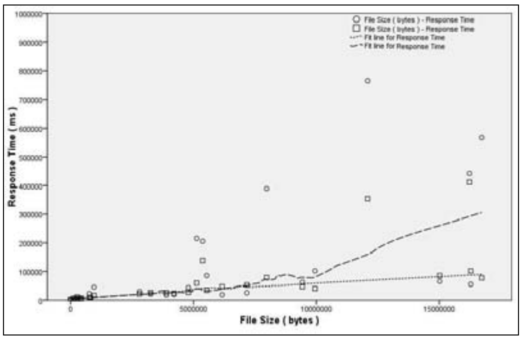


Figure 4: RemoDroid response time for 3 nodes

Based on the information collected from the devices, we were able to measure the average response time messaging and file sharing as shown in Figure 4. These had a very high variance due to the geographically challenging nature of the test bed. Above scenarios shows that the efficiency of the communication (easy to find the best path to communicate) and availability is high when the number of nodes increases Therefore, the performance of RemoDroid Messaging and file sharing are highest in four node test when compared to the three nodes test. There for test results are proved the best method is to messaging in MANET is broadcasting messaging.

3. CONCLUSION

The main advantage of RemoDroid is to provide interactive and efficient communication to the users. RemoDroid will increase the satisfaction of the user to learn a lot. Also the amount a device benefits from RemoDroid increases with the number of Nodes that can participate in the MANET. Therefore, RemoDroid is more preferable to use in an environment where a large MANET node community exists.

In future, cross compatibility can be expanded to other popular platforms also such as iOS, Windows Phone, Symbian OS, BlackBerry OS, etc. Another future extension is develop RemoDroid in different situation where can be apply mobile ad-hoc as a communication media such as a disaster recovery situation.

4. REFERENCES

[1] S. J. Robledo, "*Mobile devices for learning*", 2012.

[2] "*MobileAdHoc Network*", www.saching.com, [Online].Available:www.saching.com/Article/MANET---Mobile-AdhocNETwork--/334 .

[3] "*MANET*", www.ime.uniarlsruhe.de, [Online]. Available:www.ime.unikarlsruhe.de/Uploads/Manets-2on1a.pdf

[4] "*Better approach to mobile Ad hoc networking (B.A.T.M.A.N.)*", www.destestbed.net, [Online]. Available: www.destestbed.net/content/better-approach-mobile-ad-hoc-networking-batman.

[5] "*BATMAN*", www.open-mesh.org, [Online]. Available: www.open-mesh.org/projects/batman-adv/wiki/Wiki [Accessed: June 04, 2014]

[6] S. Brahler, *"Analysis of the android architecture*", KIT – Universitat, (2 June 2010 – 6 October 2010).

[7] A. Neumann, et al., "*Better approach to mobile Adhoc networking (B.A.T.M.A.N.)*", http://www.ietf.org/internet-drafts/draft-wunderlich openmesh-manet-routing-00.txt, Internet Draft, April, 07, 2008.

[8] D. Johnson, N. Ntlatlapa, and C. Aichele, "*Simple pragmatic approach to mesh routing using BATMAN*", 2nd IFIP International Symposium on Wireless Communications and Information Technology in Developing Countries, CSIR, Pretoria, South Africa, pp. 10, 2008.

[9] L. Sicard, Markovics, and G. Manthios, "*An Ad-hoc network of android phones using B.A.T.M.A.N.*", SPVC-E2010, IT-University

Copenhagen, 2010.

[10] A. Neumann, Corinna, “Elektra” Aichele, and Marek Lindner, “*B.A.T.M.A.N Status Report*”, June, 28, 2007.

[11] W. Tsai, “*B.A.T.M.A.N Daemon HowTo-0.2*”, Aug, 08, 2007.

[12] W. Tsai, “*B.A.T.M.A.N Daemon How To 0.3*”, January, 25, 2009.