Automated Diagnosis For Efficiency And Energy In Smartphone Using Android Applications

M. Veronica Arokia Mary
Department of computer science and engineering
Indra Ganesan College of Engineering Trichy
veronmariadass@gmail.com

D. Thenmozhi
Department of computer science and engineering
Indra Ganesan College of Engineering Trichy
thenser14@gmail.com

Mr. S. Vimalathithan
Department of computer science and engineering
Indra Ganesan College of Engineering Trichy
aathisvimal@gmail.com

Abstract- In "Smartphone applications" energy efficiency is vital, but many Android applications suffer from serious energy inefficiency problems.The project entitled is "CD App: Call Detection in Android".It is used to detect the android mobile call type whether its voice or data call.The notification is displayed when the user receive the phone calls.These application may produce energy efficient problems and reduce battery usages.So provide Greendroid tool to analyze state space search.These application may produce energy efficient problems and reduce battery usages.So provide Greendroid tool to analyze state space for each application using model checker concept.

***Index terms*-JPF, CD app, model checker, state space search technique, green droid tool.**

I. INTRODUCTION

To detect the android mobile call type whether its voice or data call User find the voice call, background download processing to send the notification for the user whether its download or not.. The main objective is to design and improved Greendroid tool .The call detection app is used to implementing in the android mobile and boot the CD APP. By this app it can be detecting whether the incoming call is voice or data call, then process is carried out and the notification is given to the user who receives the voice call or data call. if the user need to download the background process they can press the notification button as "YES or NO" .if yes the process is continued.By saving the energy consumption tools to be used called greendroid.The state space explosion problem is known as one of the most challenging issues in model checking.To reduce these issues in model checking go with the JPF Technique. The state space explosion problem is known as one of the most challenging issues in model checking.To reduce these issues in model checking and go with JPF Technique.Software testing, the most commonly used technique for validating the quality of software, is a labor intensive process, and typically accounts for about half the total cost of software development and maintenance .The state space explosion problem is known as one of the most challenging issues in model checking.

II. EXISTING METHODOLOGY

The existing methodology works under the energy saving analyze sensory data, once received by an application, would be transformed into various forms and used by different application components .Using existing method it is difficult to scheduling event handlers in android programming in android applications . Manual instrumentation is undesirable because it is labor-intensive and error-prone.There is still no well-defined metric for judging ineffective utilization of sensory data automatically.Existing Greendroid tool analyze GPS and Sensory based application in smart phones. our approach can generate detailed reports with actionable information to assist developers in validating detected energy problems. We built our approach as a tool, GreenDroid, on top of JPF. Technically, we addressed the challenges of generating user interaction events and scheduling event handlers in extending JPF for analysing Android applications. We evaluated GreenDroid using real-world popular Android applications. GreenDroid completed energy efficiency diagnosis for these applications in a few minutes. JPF was originally designed for analysing conventional Java programs with explicit control flows . It executes the byte code of a target Java program in its virtual machine. However, Android applications are event-driven and depend greatly on user interactions. Their program code comprises many loosely coupled event handlers, among which no explicit control flow is specified. At runtime, these event handlers are called by the Android framework, which builds on hundreds of native library classes. As such, applying JPF to analyse Android applications requires: (1) generating valid user interaction events, and (2) correctly scheduling event handlers. To address the first technical issue, we propose to analyse an Android application's GUI layout configuration files, and systematically enumerate all possible user interaction event sequences with a bounded length at runtime. We show that such a bounded length does not impair the effectiveness of our analysis, but instead helps quickly explore different application states and identify energy problems. To address the second technical issue, we present an s application execution model derived from Android specifications. This model captures application-generic temporal rules that specify calling relationships between event handlers. With this model, we are able to ensure an Android application to be exercised with correct control flows, rather than being randomly scheduled on its event handlers. As we will show in our later evaluation, the latter brings almost no benefit to the identification

III. PROPOSED METHODOLOGY

I. EMPIRICAL STUDY

We report our findings from an archival study of real energy problems in Android applications. For ease of presentation, we may use "energy problems" and "energy bugs" interchangeably in subsequent discussions. Our study aims to answer the following three re-search questions:

A. RQ1 (Problem magnitude):

Are energy problems in An-droid applications serious? Do the problems have a severe impact on smart phone users?

B. RQ2 (Diagnosis and fixing efforts):

Are energy problems relatively more difficult to diagnose and fix than non-energy problems? What information do developers need in the energy problem diagnosis and fixing process?

C. RQ3 (Common causes and patterns):

What are common causes of energy problems? What patterns can we distill from them to enable automated diagnosis of these problems*?*

It analyzes sensory data, once received by an application, would be transformed into various forms and used by different application. Using existing method difficult scheduling event handling is done.

GREENDROID TILE IN SMARTPHONE APPLICATIONS

45 nm process
1.5 GHz
30k instances
Blank space is filled with
a collection of c-cores
Each tile contains different c-cores
Where do the energy savings come from?
Supporting Software Changes
Software may change HW must remain usable
C-cores unaffected by changes to cold regions
Can support any changes, through patching
Arbitrary insertion of code software exception
Mechanism
Changes to program constants configurable registers

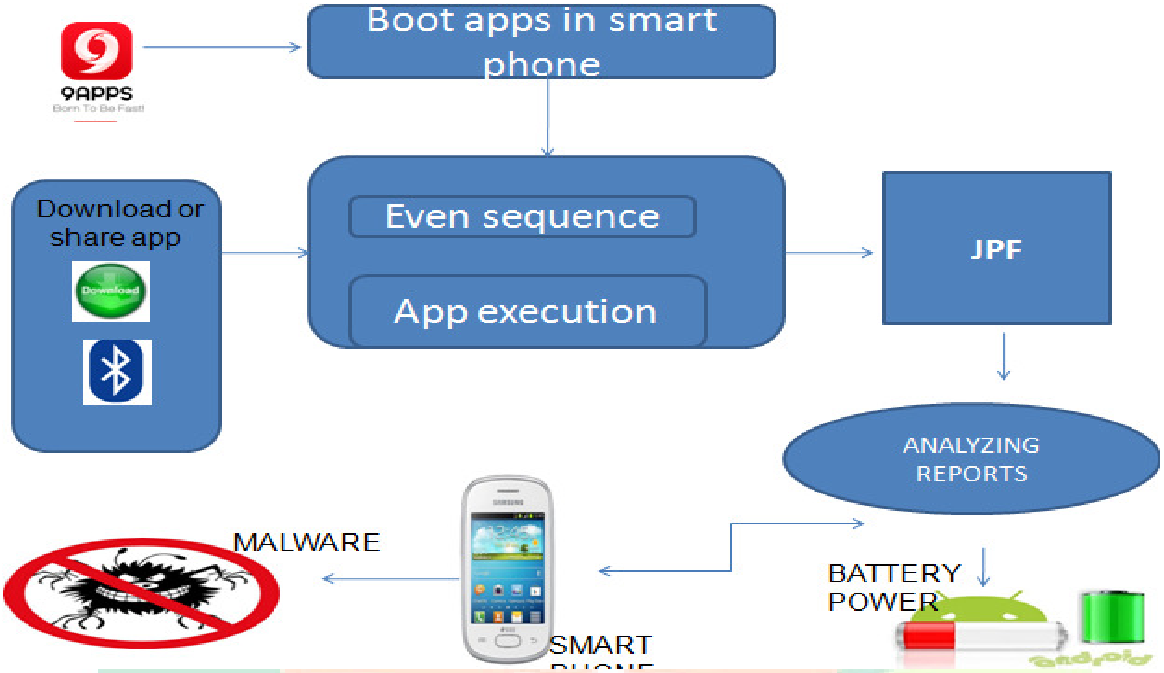
GREENDROID PROPOSED RELATED WORK IN ANDROID

Improve Green droid tool find application executiowhile Sharing application at time of blue tooth sharing.

Implement Java Path Finder (JPF) which is an state model checker that directly works with Java byte code instructions.

ARCHITECUTRE:

This diagrammatic structure represented as boot the apps that created by own apps and then event sequence and action could be performed. The actions are performed and give to the JPF called Java Path Finder. The JPF works and they analyses the reports about their malware actions are performed and green droid tool make them to perform with the smart phone applications, and saves the battery performance.



Generally, in android mobile have a access permission option for each and every activity such as call, message, website search and so on. In our system the users have permission to access the call whether its voice call. Once the system finds the voice call the remaining is data call. Hence, the system sends the notification to user receiving call. By achieving this we can prevent the data loss. Mobile balance should not be reduced. Identify voice call and data call. it gives the notification .

PERMISSION PROCESS

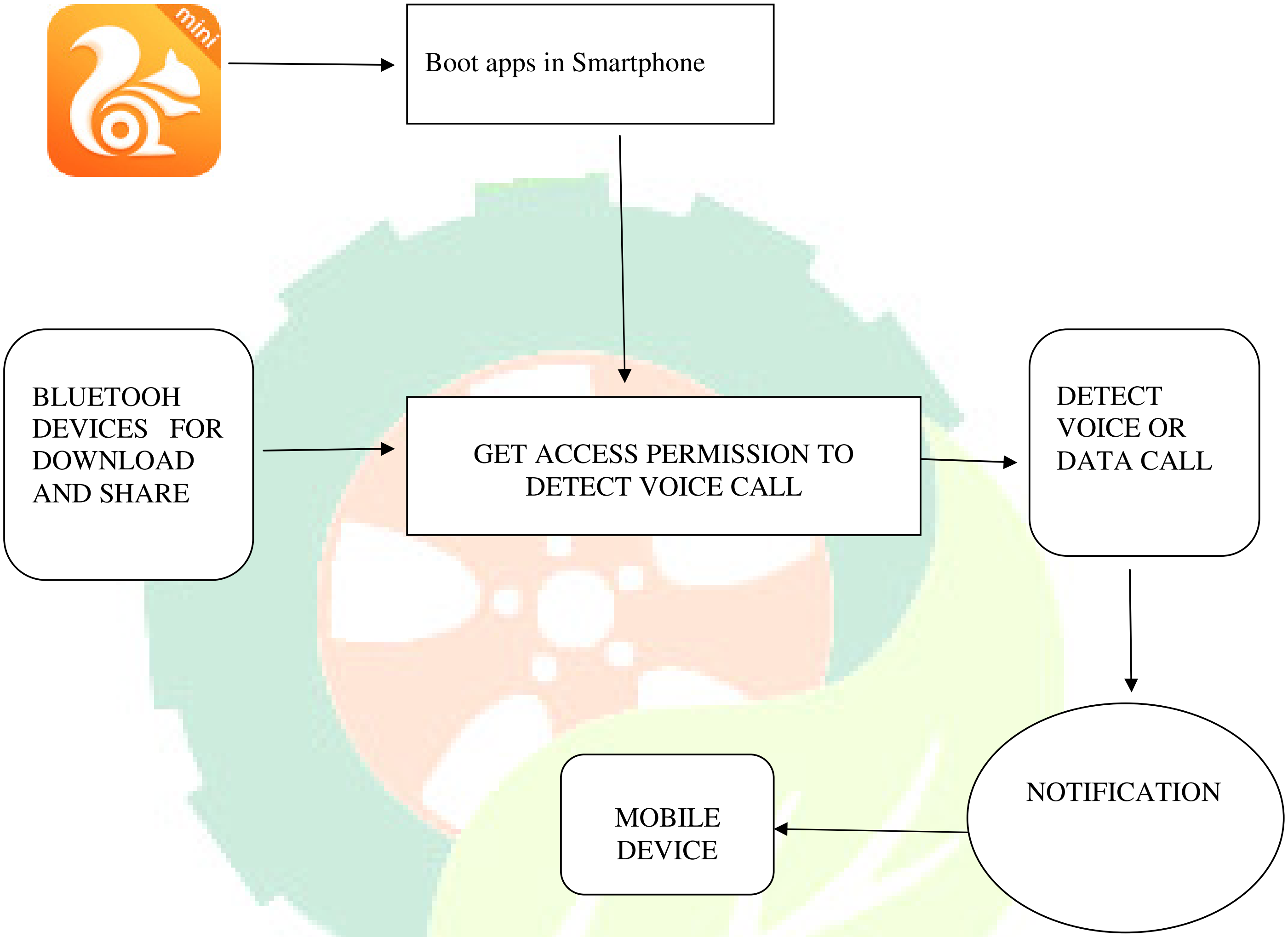
Permission process is the way to get the call detection of android mobile for detecting voice call .once to get the permission it will detect the all voice call using the permission

DETECT THE CALL

Get access permission to the call manager easy to detect voice call. Detect voice call then easy to identify the data call. Set of instruction can performed receiving call in android mobile.In this instruction can controlled in our CD application using developing codes

VALIDATION THE PROCESS

Above all process can completed correctly the validation is correct then the process was completed. Finally to obtain correct output solution. Give the input files as APK.files and then boot it .Events and actions can be performed. It can extract user interaction events and event sequence details. In this module to create own application for detecting voice call or data call in android mobile. Create application using android software. After create CD application to initialized in the android mobile. Proper initialization can performed in android mobile. Malware predictions finally we can predict and delete malwares from our smart phones. Perform on destroy operation to delete malwares and preserve energy at the time application usage. We can evaluate the performance of the system. Our proposed approach provide improved prediction rate and minimized energy levels


mini
Boot apps in Smartphone
BLUETOOH DEVICES FOR DOWNLOAD AND SHARE
GET ACCESS PERMISSION TO DETECT VOICE CALL
DETECT VOICE OR DATA CALL
NOTIFICATION
MOBILE DEVICE

ARCHITECTURE FOR NOTIFICATION PROCESS

Creating the own apps that should boot in the Smartphone and it either download or share the app either in Bluetooth device or share it and get the access permission to detect the voice call and after accessing that it will detect that is either voice call or data call. And then detecting the call is data call it will send the notification to the mobile. Above all process the validation is completed. Then correct the process was completed. Finally it obtains the correct output solution.

FUTURE WORK

The future work is in above process is implemented as the notification only perform by events and not by actions. so the work procedure can be done as above mentioned in abstract.

CONCLUSION

The mentioned entitled as project work as saving the energy and it will automatically diagnosis the power in Smartphone using android app. It is very useful for the upcoming generations

REFERENCES

1] J. Babb et al., “Parallelizing Applications into Silicon,” Proc. 7th Annual IEEE Symp. Field-Programmable Custom Computing Machines, 1999.
[2] R. Dennard et al., “Design of Ion-Implanted MOSFET’s with Very Small Physical Dimensions,” IEEE J. SolidState Circuits, Oct. 1974.
[3] N. Goulding et al., “GreenDroid: A Mobile Application Processor for a Future of Dark Silicon,” HotChips, 2010.
[4] V. Kathail et al., “Pico: Automatically Designing Custom Computers,” Computer, vol. 35, Sept. 2002, pp. 39–47.
[5] J. Sampson et al., “Efficient Complex Operators for Irregular Codes,” Proc. Symp. High Perf. Computer Architecture, Feb. 2011.
[6] M. Taylor et al., “The Raw Processor: A Scalable 32-bit Fabric for General Purpose and Embedded Computing,” HotChips, 2001.
[7] G. Venkatesh et al., “Conservation Cores: Reducing the Energy of Mature Computations,” ASPLOS XV: Proc. 15th Int’l. Conf. Architectural Support for Prog. Languages and Op. Sys., Mar. 2

STEVENSWANSONis an assistant professor at the University of California, San Diego, and jointly leads the GreenDroid project. His areas of interest include specialized architectures for low-power computing and system-level applications for non-volatile solid state memories. He received his Ph.D. from the University of Washington in 2006.

MICHAEL BEDFORDTAYLORis an assistant professor at the University of California, San Diego, and jointly leads the GreenDroid project. He was lead architect of the 16-core.