ASEE's 123rd Annual • Conference & Exposition • New Orleans, LA • June 26-29, 2016

New Orleans JAZZED about Engineering Education

Paper ID #15364

Mobile Computing & Security Laboratory Development

Dr. Yujian Fu P.E., Alabama A&M University

Dr. Yujian Fu is an associate professor of computer science department at Alabama A&M University. Her research interests fall in formal verification of cyber physical systems, behavioral analysis of mobile security, software architecture and design analysis of safety-critical and mission-critical systems. Her projects are supported by NSF, Air Force and DoD. She have several publications regarding to the research and educational projects.

Dr. Di Ma, University of Michigan - Dearborn

Dr. Di Ma is currently a visiting professor at UMTRI and an Associate Professor at the Computer and Information Science Department of the University of Michigan, Dearborn. She obtained her PhD degree from the University of California, Irvine in 2009. She also received a B.Eng. degree from Xi'an Jiaotong University, China and a M.Eng. degree from Nanyang Technological University, Singapore. She was with the Institute for Infocomm Research, Singapore (2000-05). She is a recognized expert with 10+ years of research experience in Computer/Network/Wireless and Mobile Computing/Data Storage security and privacy. She has extensive experience in designing mitigation techniques such as authentication, secure delegation and authorization, access control, and secure protocols.

Mobile Computing and Security Laboratory Development with Flip Teaching

Abstract

Android has reached over 1 million devices and occupies 85% of the market in 2014, according to a recent market report. Due to many advantages of the Android platform, such as open source, Google support, advanced software programmable framework in popular programming language Java, Android has been and will remain as the most popular mobile device operating system in market. Due to large popularity of user markets, research study of mobile computing is raising to a challenge level. In addition, the number of malicious applications is increasing continually. All these reasons raise a high challenge to computer science researchers and educators. How to build the next generation of workforce that are equipped with techniques and problem solving capabilities in the mobile pervasive computing and security has drawn attentions to researchers and STEM educators. As a collaborative effort supported by NSF program, this paper presented a laboratory development conducted in computer science program at Alabama A&M University regarding to mobile computing and security issues. In addition, class studies that apply the laboratories to classroom teaching using flip-flop are briefly discussed. Several interesting observed experimental results that demonstrated a relation between project based learning (PBL) and flip teaching through an online environment were presented in the end. These studied were facilitated by the mechanisms provided by Blackboard.

Keywords

Mobile computing; Security; Virtualization; Sniffing; encryption.

Introduction

Android has reached over 1 billion active users, revealed by the senior vice president of Android. Android occupies 85% of market in 2014, according to report [1]. Due to many advantages of the Android platform, such as open source, Google support, advanced software programmable framework in popular programming language Java, Android has been and will remain as the most popular mobile device and operating systems in market.

Due to the large popularity of user market, research study of mobile computing is raising to a challenge level. In addition, the number of malicious applications is increasing continually. All these reasons raise a high challenge to computer science and educators. How to prepare the next generation of workforce that are equipped with techniques and problem solving capabilities in the mobile pervasive computing and security has drawn attentions from both researchers and STEM educators.

This paper presented a research study of the mobile and pervasive computing and security from two angles of view points. One is how to equip the next generation of workforce with strong technical skills and problem solving capabilities. Another one is how to raise students with a motivated and challenging environment so that learning can be effectively and efficiently enforced in the applied topics and laboratories.

Hands-on experience is a key for students to learn concepts and definitions in all areas. Project based learning (PBL) is absolutely an effective approach to improve learning results by motivating students. We use a simplified flipped classroom to test the labs and relations with PBL learning through two courses in 2012 and 2015. This collaborative research work has demonstrated meaningful scientific results on the laboratory development and socio-psychological results on the relations between PBL and FF teaching strategy.

Pedagogical Theory & Teaching Strategy

Hands on experience is good for kids. How to apply labs and projects and instill to the lectures and classroom teaching seamlessly remains silent issue in past decades. In this section, we will discuss the traditional PBL learning and the new FF teaching strategy using hands-on labs.

Project based Learning

The PBL based learning aims at ensuring students to learn better by engaging into real world problems. The umbra domain of mobile and pervasive computing provides overall problem paradigm. Several teaching underpinnings must be considered carefully when to use PBL learning into the curriculum and classroom teaching.

Planning and Evolution. Projects designed and used in the classroom are mostly cross-curricular PBL problems. The thoughtful design with the teaching experience needs to blend into the context aware content. In addition, measurable goals and objectives that are challenge enough to students need to be counted.

Standards. Standards of electronics and computer science curriculum and program goals must be instilled and reflected to the projects. We need to consider the ABET requirement for the computer curriculum and program expectations. Overall, projects need to be developed by ourselves to be authentic to students and local community. The research study is supported by NSF DUE program requirements are considered.

In addition to the above project design issues, PBL carries the characteristics of pedagogical underpinnings.

Innovation. Mobile computing is a new emerging area in computer science. Research result on the mobile security is raising dramatically due to the dramatically increase in the device market. Students are excited to know and learn the technical content about mobile and pervasive computing and security.

Student activity centered. The designed context aware projects are well polished regarding to the course concepts and software development processing cycle [2]. Each development phase is managed by tasks that correlated to the student's role and timeline.

Behaviorist. All projects need to promote student learning outcomes to reinforce the association of actions between project goal, expectations and design tasks. Project presentation and progress reports during the project performance period will identify the contributions on the part and expectations towards the results.

Collaborative learning. Collaborative learning is emphasized by the social interaction during the team up process. The theory behind this is rooted from Vygotsky's socio-cultural psychology [2].

FF Approach: Flip classroom vs. Flip laboratory

Flipped classroom is defined as using technology to provide lectures outside the classroom, while assignments with concepts are provided inside the class with learning activities [4]. Herreid and Schiller [5] asserted that a flipped classroom engages and focuses students' learning by combining active, student-centered learning with content mastery that can be applied in the real world. According to Clark [4], activities with real-world scenarios could be implemented by hands-on and project-based learning activities during class time to enhance students' understanding and comprehension of the content and to encourage them to verbalize their engagement with such activities. There are some challenges and problems that must be faced by the lecturer and the students using a flipped classroom to promote active learning as a means of enhancing student engagement.

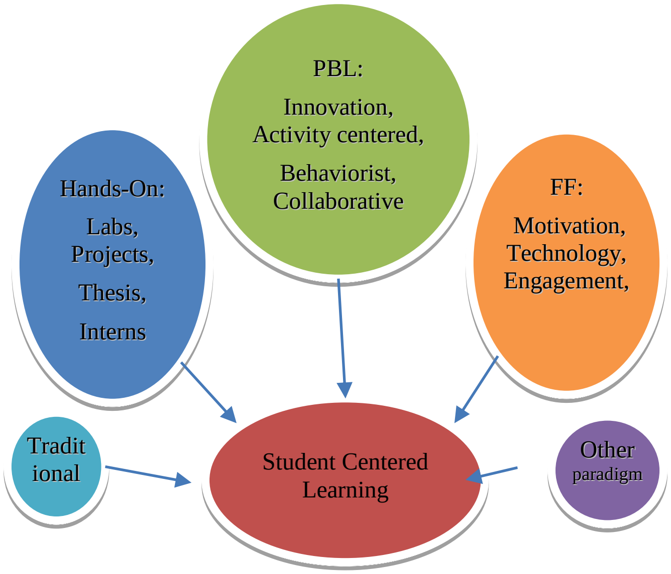


Figure 1. Student centered learning framework.

The main challenge issue of flipped classroom teaching is the lack of effective models [6]. For formal settings of computer science and STEM areas, classrooms, computer labs equipped with Internet access and access administration are well suited for flipped classroom. At this time, based on our current statistics at AAMU, informal environment statistical data indicated that one out 36 sophomore (2.7%) student does not have laptop and 1 out of 51 sophomore (~2%) or juniors does not have Internet at home.

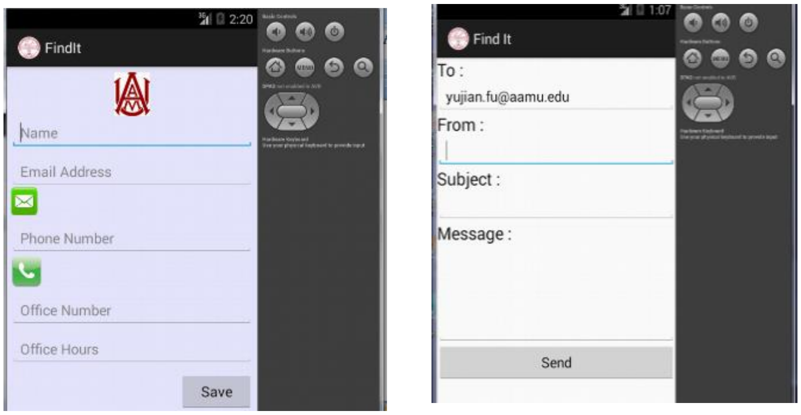
Due to limited credit hours, we do not have separate lab sessions. In addition, these labs are mainly used in the junior and senior classes, which do not provide lab hours. If we can provide the lectures online, we may be able to allocate some sessions for our mobile computing labs. Here we introduce the idea of flipping to expose students to the mobile computing laboratory. Allow students to study before the lecture and complete the laboratory.

Student-Centered Framework

The main theme of these laboratory development focus on the student centered activities to improve learning outcomes by a project based learning paradigm. The designed labs will be used through the class content and flow into the course context through the semester so that an active learning and engagement of the student will be reached.

Mobile Security Laboratory Development

We have initiated the mobile computing and security projects and labs and integrated them to classroom since Spring 2012. Most of the labs were developed and applied in the software engineering and senior design classes, which are senior courses for computer science and electrical engineering students. Later, with new labs developed more courses continued, we applied in wireless computing and graduate courses. The mobile computing and security labs were developed in four main categories –system level design for the mobile platform (Android); API level security analysis (PID recognition); reverse engineering based security analysis including both static and dynamic analysis; traffic engineering. Table I. shows the overall list of current mobile computing labs and security analysis labs related to mobile and pervasive computing in the past two years. Each lab will be introduced in the following subsections. Each lab contains the objectives, description, steps, and some sample code segments. Some labs had been updated based on past implementation.



To : yujian.fu@aamu.edu

Figure 2. Snapshots for the information collection (a) and phone make lab

Information Collection Android App

This is an Android app for students to be familiar with the Android architecture and APIs, communication, and programming platforms through a design of information app. Extra learning is to build a database that save the collected data. A simple data stream maybe need from the ADT bundle.

```
fname=edit_first.getText().toString().trim();
eaddress=edit_email.getText().toString().trim
();
pnumber=edit_phone.getText().toString().trim(
);
onumber=edit_officeN.getText().toString().tri
m();
ohours=edit_officeH.getText().toString().trim
();
```

(a)

```
if(fname.length()>0 && eaddress.length()>0
&& pnumber.length()>0 && onumber.length()>0
&& ohours.length()>0) {       saveData();}
else{
AlertDialog.Builder alertBuilder=new
AlertDialog.Builder(EditActivity.this);
alertBuilder.setTitle("Invalid Data");
alertBuilder.setMessage("Please, Enter valid
data");
```

(b)

Figure 3. Sample code for the information collection.

Main APIs: onClick, onClickListener, phoneCallListener, TelephonyManager, and some APIs to check phone state change (onCallStateChange)

Code Segment: A sample code for the information collection is shown in Figure 3 (a). Data validation is shown in Figure 3 (b).

Software Testing on Encryption

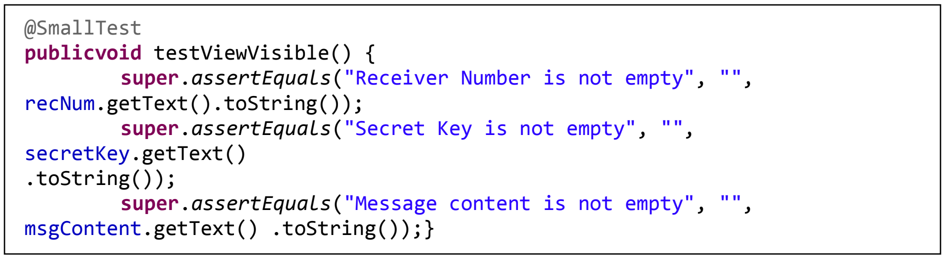
Software quality assurance costs a lot due to the complexity of the large scale software intensive systems. Software testing on the Android application remains new and challenge due to various types of paltforms used in the mobile systems. Most traditional testing tools are not available for Android application testing. This contains two labs – android testing tool (JUnit, Robotium and Selendroid) and specification based testing on AES algorithm.

The objectives of this lab category are: 1) Understand the software testing and challenge in the mobile application. 2) Understand the testing on the authentication and encryption algorithms in Andriod. AES and AES+ will be introduced in the introduction during lecture time. 3) Master the specification based testing.

Tools. Three software testing tools are introduced to students in lecture video based version.

Specification of cryptographic algorithm. AES algorithm is the most widely accepted algorithm for data encryption. Many SMS applications are developed based on the variant or improved AES [7]. In SDK bundle, javax.crypto.spec is a package that supports the classes and interfaces needed to specify keys and parameter for encryption. Keys maybe specified via algorithm or in a more abstract and general way with ASN.1. For example, *SecretKeySpec*() is the key specification for a SecretKey and also a secret key implementation that is provider-independent. It can be used for raw secret keys that can be specified as byte. Depending on the properties of user requirements, with the above class design, you may specify different types of OCL properties to check is the AES algorithm implemented correctly or not.

Assertion. Assertions will be added based on the above specified properties. In other words, assertions are translated version of OCL properties. An object Assert from junit.Assert pack needs to instantiated. In Assert class, a set of assertion methods useful for writing tests. Only failed assertions are recorded in junit. The main API here is assertEquals(…), where the parameter list can carry String, integer and any type of user defined objects. An example of assertion is shown in Figure 4.



```
@SmallTest
publicvoid testViewVisible() {
        super.assertEquals("Receiver Number is not empty", "",
recNum.getText().toString());
        super.assertEquals("Secret Key is not empty", "",
secretKey.getText()
.toString());
        super.assertEquals("Message content is not empty", "",
msgContent.getText() .toString());}
```

Figure 4. Some assertions for identity test in the algorithm.

WiFi Traffic Sniffing

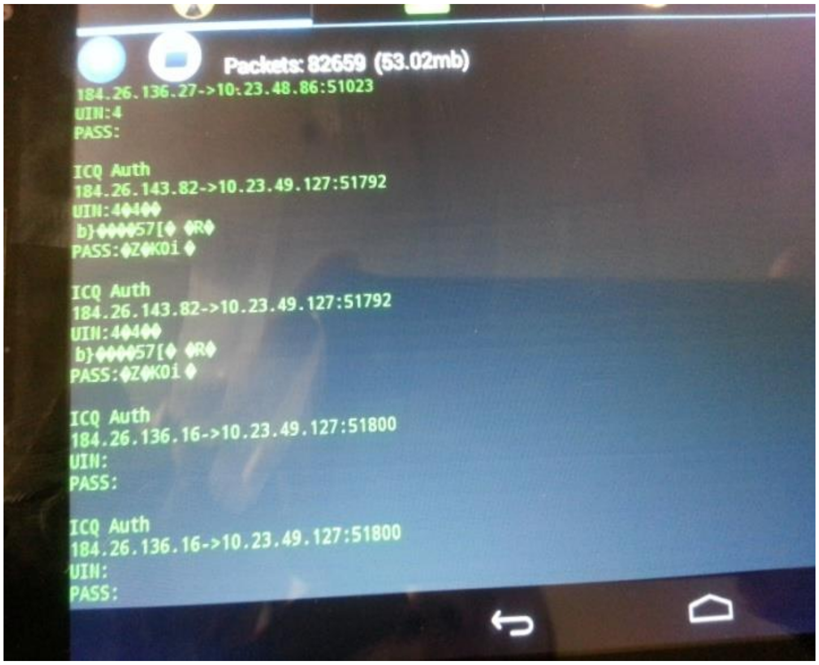
Traffic engineering [8]is a method of adapting and optimizing transmit performance of a telecommunication network by analyzing, predicting and regulating the behavior of data that transmitted over that network. The methods are applied to all types of network including PSTN (public switched telephone network), LANs (Local area networks), WANs (wide area networks), cellular telephone networks, and Internet. The main ideas behind traffic engineering are good user performance and efficient use of network resources by finding an optimized routing path based on existing or adapted configuration.

The objectives of this lab are (1) being familiar with the concepts of mobile and cellular networking diagnosis, packet sniffing in both traditional and cellular networks. (2) being familiar with current mobile packet sniffing tools and being able to use one tool to get the data. (3) being able to collect packet information from existing network and generate analysis results.

This lab is composed of two sublabs – one is the tools and the other is the data analysis.

Tools. We introduce two tools – Intercepter-NG and Wiresharks – for students to start with traffic engineering in the mobile IP, celluar networks on Android apps. Intercepter-NG is a powerfull Andorid packet sniffing App that can sniff both wired and wireless traffic, steal cookies, view usernames/passwords, URL of sites visited and so on. A step based instructions for student to be familiar with this tool after class is provided. Wireshark is a typical network analysis tool designed to capture packets in the real time. The current wireshark Android version can provide the mobile network diagnosis includes filters, color-coding and other network traffic and packet inspection functions. Both tools work on root device.

A snapshots of network packet sniffing are shown in Figure 5.



```
Packets: 82659 (53.02mb)
184.26.136.27->10:23.48.86:51023
UIN:4
PASS:

ICQ Auth
184.26.143.82->10.23.49.127:51792
UIN:4◆4◆◆
 b}◆◆◆◆57[◆ ◆R◆
PASS:◆Z◆KOi ◆

ICQ Auth
184.26.143.82->10.23.49.127:51792
UIN:4◆4◆◆
 b}◆◆◆◆57[◆ ◆R◆
PASS:◆Z◆KOi ◆

ICQ Auth
184.26.136.16->10.23.49.127:51800
UIN:
PASS:

ICQ Auth
184.26.136.16->10.23.49.127:51800
UIN:
PASS:
```

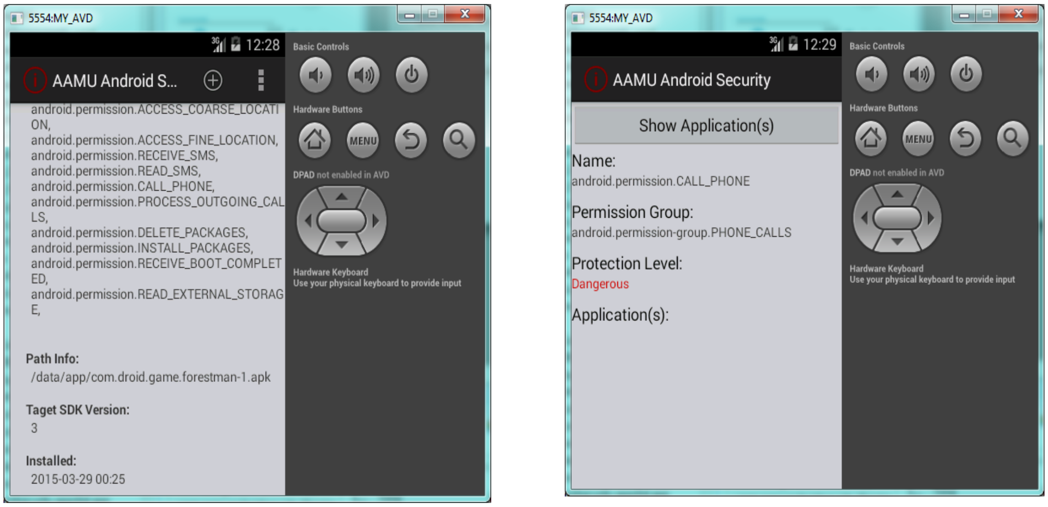
Figure 5. Snapshots of the sniffing on AAMU wifi using Intercepter-NG.

API Level Malware Analysis Approach-Permission ID Analysis

There are three different types of malware detection techniques: attack or invasion detection, misuse detection (signature-based) and anomaly detection (behavior-based) [9].This lab was designed for students to be familiar with the typical malware analysis techniques in different categories.

Android permissions control the access to sensitive resources and functionalities. There are 134 Android-defined permissions are available to third party applications in Android 2.2[10]. Permissions are defined with one of the four different protection levels, which characterize the potential risks implied in the permission and enforce different install-time approval processes. Permissions have associated protection levels: Normal, Dangerous, Signature, and SignatureOrSystem.

Objectives of this lab are (1) Understanding the permission ID assignment is a required process for an Android app. (2) Understanding the different levels of permission ID and risks to the Android system. (3) Being able to capture the requested permissions and display the level of permission ID.



android.permission.ACCESS_COARSE_LOCATION,
android.permission.ACCESS_FINE_LOCATION,
android.permission.RECEIVE_SMS,
android.permission.READ_SMS,
android.permission.CALL_PHONE,
android.permission.PROCESS_OUTGOING_CALLS,
android.permission.DELETE_PACKAGES,
android.permission.INSTALL_PACKAGES,
android.permission.RECEIVE_BOOT_COMPLETED,
android.permission.READ_EXTERNAL_STORAGE,

Path Info:
/data/app/com.droid.game.forestman-1.apk

Taget SDK Version:
3

Installed:
2015-03-29 00:25

Name:
android.permission.CALL_PHONE

Permission Group:
android.permission-group.PHONE_CALLS

Protection Level:
Dangerous

Application(s):

Figure 6. Display permission ID with different level of securities for an apk file.

Android sandbox does not provide mechanisms for users to filter and secure the permission ID request from third parties, it will be important for students to understand that reveal the permission with levels of security and to decide if the app can be allowed, understand the level of potential damages by assigning the permissions. Four steps are required for the permission ID analysis: (1) load the apk file. (2) filter the permissions requested by the apk file. (3) display all permissions requested. (d) highlight the dangerous permissions, and allow user to reject the request.

Analyzing the permission requests with the level information and risk assessment is a complex research issue and out of the laboratory development scope. Some of the most popular available methods in the machine learning will be introduced to students. Those content are not required for learners to conduct the above labs.

Malware Analysis - Dexter

Dexter is a web based free Android application analysis tool supported by DEXlab from Bluebox [11], [12]. From our current study, this tool can provide bite code and source code static analysis, even some web page listed some dynamic analysis functions. The information displayed to students includes the overall program statics analysis, class diagram, basic block graph and the project document profile. We adopted iCalendar.apk as an example here to show all the above features, APIs and statistic information.

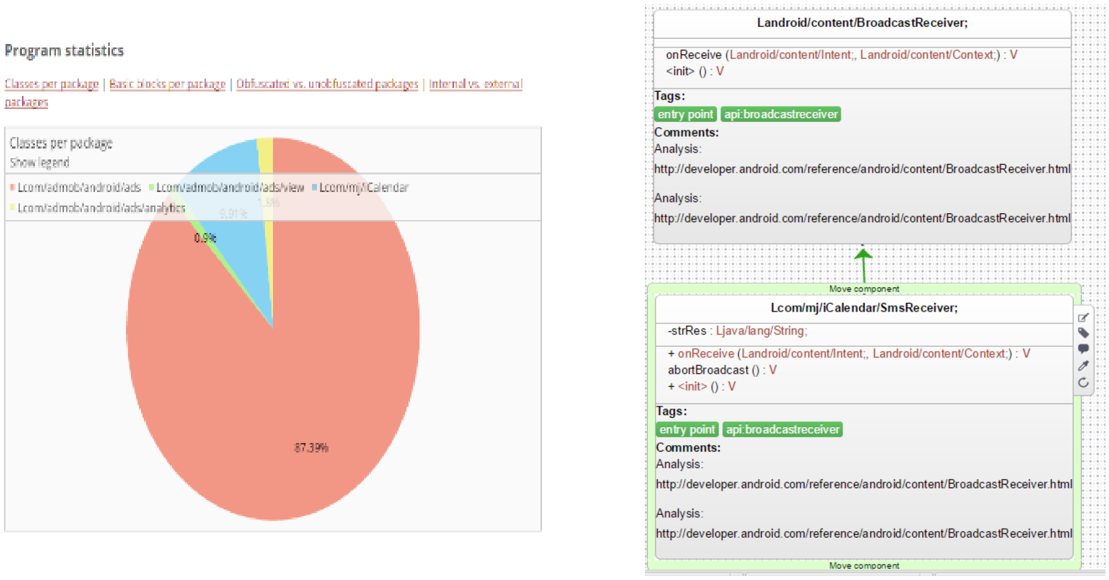

Program statistics
Classes per package | Basic blocks per package | Obfuscated vs. unobfuscated packages | Internal vs. external packages
Classes per package
Show legend
Lcom/admob/android/ads
Lcom/admob/android/ads/view
Lcom/mj/iCalendar
Lcom/admob/android/ads/analytics
9.91%
1.8%
0.9%
87.39%
Landroid/content/BroadcastReceiver;
onReceive (Landroid/content/Intent;, Landroid/content/Context;) : V
<init> () : V
Tags:
entry point
api:broadcastreceiver
Comments:
Analysis:
http://developer.android.com/reference/android/content/BroadcastReceiver.html
Analysis:
http://developer.android.com/reference/android/content/BroadcastReceiver.html
Move component
Lcom/mj/iCalendar/SmsReceiver;
-strRes : Ljava/lang/String;
+ onReceive (Landroid/content/Intent;, Landroid/content/Context;) : V
abortBroadcast () : V
+ <init> () : V
Tags:
entry point
api:broadcastreceiver
Comments:
Analysis:
http://developer.android.com/reference/android/content/BroadcastReceiver.html
Analysis:
http://developer.android.com/reference/android/content/BroadcastReceiver.html
Move component

Figure 7. (a) Program statistics of iCalendar.apk. (b) the class diagram generated from sendSms().

The most advantages of Dexter over other analysis tools are the class diagrams that overall related automatically generated from the source code. In addition Dexter provides a colorful and friendly user interface with the optional agenda, which are not available for many other analysis tools. Figure 6 shows a basic block graph in the assembly code to indicate the call relations for sendSms that is invoked by iCalendar class.

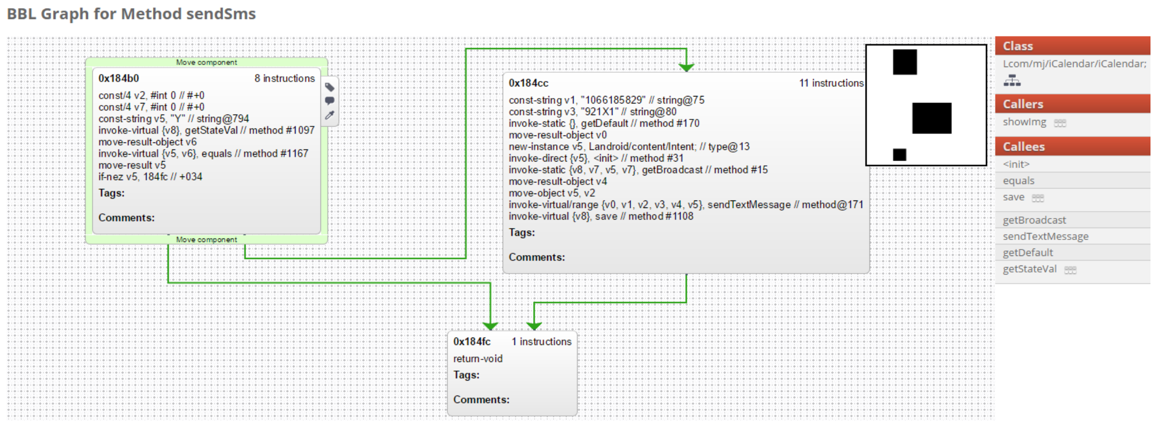
Static Analysis - APK Inspector

APKInspector integrates the static analysis tools and provides graphic features which bring convenience to the malware analysis. APKInspector can provide control flow graph (CFG) and call graph relation, static instrumentation and permission analysis. APKInspector reverse the code with Ded for java analysis. It adds support for odex.

Table 1. A Summary of Mobile Computing and Security Labs

Category	Lab	Status	Applied
Mobile Computing	1 . Info Collection	complete	2013
	2. Email	complete	2013
	3. Phone call	complete	2013
	4. Database	complete	2013
SystemAnalysis – PermissionID	5. Load APK file	complete	Not
	6. Display PID	complete	Not
	7. Remove APK	complete	Not
	8. PID analyzer	On-going	Not
Static Analysis	9. Dexter	complete	2015
	10. ApkTool	complete	Not

Traffic Engineering	11. Intercepter	complete	2015
	12. Wireshark	On-going	Not
Software Testing	13. Cryptographic Algorithm	Complete	2013
Summary	13	C (11) OG(2)	A (7)



```
0x184b0                                         8 instructions
const/4 v2, #int 0 // #+0
const/4 v7, #int 0 // #+0
const-string v5, "Y" // string@794
invoke-virtual {v8}, getStateVal // method #1097
move-result-object v6
invoke-virtual {v5, v6}, equals // method #1167
move-result v5
if-nez v5, 184fc // +034
Tags:

Comments:
```

```
0x184cc                                         11 instructions
const-string v1, "1066185829" // string@75
const-string v3, "921X1" // string@80
invoke-static {}, getDefault // method #170
move-result-object v0
new-instance v5, Landroid/content/Intent; // type@13
invoke-direct {v5}, <init> // method #31
invoke-static {v8, v7, v5, v7}, getBroadcast // method #15
move-result-object v4
move-object v5, v2
invoke-virtual/range {v0, v1, v2, v3, v4, v5}, sendTextMessage // method@171
invoke-virtual {v8}, save // method #1108
Tags:

Comments:
```

```
0x184fc     1 instructions
return-void
Tags:

Comments:
```

Figure 8. The Basic Block Graph (BBG) for method sendSms.

Experimental Studies, Current Results & Discussion

Experimental Studies

The labs were started to develop as early as April 2013. Most mobile computing labs were implemented and applied in the Fall 2013. Not all labs were adopted by classes yet. The classes that adopted these developed labs are mainly senior class (CS 401 Software Engineering, and CS 408 Wireless Computing). Students in these classes are fully prepared for the fundamental knowledge of information security, operating systems, and especially Java programming language. As senior standing students, they are mature enough to lead a group by effective communication and working collaboratively on the same topic. Therefore, the developed labs can be connected and continued smoothly to reach the goal of the program.

The only concern is the mobile computing platform, Android APIs and eclipse plugins. However, with some of the help, students are able to connect most of the knowledge and complete the project with satisfactory results. The grade and survey results regarding to the context will be discussed in the follow subsections.

Current Results

The bias exist in the grade expectation and the design flaw of the labs will be in the secondary place. Most of the labs are split into small steps that students are able to handle with relation to the class contents. The class diagram in the software engineering class is required before students started with the implementation. Since these labs are mostly newly developed and not yet applied to many other classes, the current results from the implementation may not be effective wide around. A short description of student grade with the labs and context are shown in Table II. We use scale of 1 to 5 for the student survey satisfaction, where 1 not satisfied, 5 is excellent. In addition, we also code the difficult level of each lab by numbers 1 (Extremely Easy) to 5 (Extremely Hard).

Table 2. Student Class Grade with Lab Surveys

Lab Num	Course Grade	Survey Satisfy (Average value)	Difficult Level
1	A (100%)	5	1
2	A (100%)	5	4
3	A (100%)	5	4
4	A (67%)	>= 4	4.5
9	A (100%)	>= 4	2
11	B (100%)	>= 4	3
13	A (67%)	>= 3	4.5

Discussion

The most important information from Table II is the more difficult of the lab, the lower satisfaction from survey. In addition, we revisit our pedagogical issues considered from the lab development with regarding to teaching strategies.

Learning – PBL

PBL has been proved to motivate students using meaningful projects through the class. The selection of projects are relevant to the class context and teaching strategies. The size of the projects varies based on the class assessment, requirements, and context, which can be as large as a term project, or as small as a large size lab. The developed labs are being used by various courses to validate the teaching strategy as well as motivate students to have a better understand of the concepts and context. The PBL is more effective in the senior courses such as Software Engineering, Senior Design and Wireless Computing. In addition, students are mature and experienced for the program to prepare for the large size of labs and term projects in a team based style. In addition, through these years of teaching in senior and graduate level classes, we have finalized a set of project evaluation process to insure the collaboration, leadership, responsibility and other project management issues.

Teaching – FF

We have a short implementation of FF teaching in the class Wireless Computing. Two opposite sides of the facts have been observed through this short study. We claim that the effectiveness of

FF will still remain unanswered during our current study. First, it works well for the motivated students. Second, it is not effective for those students behind in the class.

Our implementation was done in continuous seven session at the end of the semester. Both lectures, videos and other materials such supplements were provided in the Blackboard (v9).

Summary

The implementation of FF needs more studies in other sessions and classes. Even we have provided types of different videos, and other techniques, due to the short performance time, some bias still exist. The learning outcomes are highly related to the grade and motivation. Even some research has demonstrated that provided quizzes in conjunction with video lectures can improve student motivation [13], [14], the improvement are also tightly bounded to the grade. Finally, student motivation can be improved through bounded project or problems during in class activities that are able to connect to the lectures [15].

Related Works

The literature review of the lab development falls into two main categories – mobile computing lab development, and related security development.

Even though the course of mobile computing has been offered by many institutions, there are not many sets of well-developed labs. Most of courses are offered in the senior or graduate level, thus only a few of them provide the lab assignments and grading policies. University of Wisconsin-Madison offers mobile computing laboratory [16], where a sequence of four labs are defined that covers the initial preparation, program basics, sensor and multimedia programming, and communicating with cloud. University of Florida has offered mobile and pervasive computing courses with labs from 2002 [17]. Two labs of Android platform are offered for context of location based service, mobile service discovery, crowdsourcing which are also related the term project. Payne provided a series of mobile computing laboratories for the class Mobile Computing at University of Liverpool. These cumulative labs are developed on iOS8 platform.

Regarding mobile security, the most popular laboratory repository is the SEED project developed by Kevin Du since 2009, where 21 labs were developed [18]. Several new mobile security and relevant labs were developed later [19]. There are 28 labs regarding to the web browser and mobile security with different level of difficulties [19]. Wireless Network & System Security group at Carnegie Mellon University Silicon Valley offers Mobile Security Cybersecurity Research and wireless Networking Security [20]. The most recent mobile security has four assignments such as steal personal information, exploiting another app. For the later one, bytecode-based analysis are included as part of requirements.

Conclusion and Future Works

Mobile and pervasive computing and security are still remain very young domain in current research and education. Many more new malwares and analysis approaches and latest techniques to identify novel malwares are expected and highly desired for our classroom and computer curriculum. To build the workforce of the future mille and fulfill the national security needs in the job marketing, developing systematics and constructive laboratories will be the key in the recent computer science and STEM education. Challenge faced by educators and professionals, other than the curriculum, labs and projects, will be new teaching methodologies that are able to be used to instill the classroom context and combine the labs and project to the context and definitions. In this paper, we presented our work in both lab development and experimental study of the FF teaching in the courses. In the future, we will need to (a) develop more labs for student needs and context related. (b) conduct more class studies for the classroom investigation. (c) adopt new methods to integrate the research results into the lectures and labs of various types of courses.

Acknowledgements

This project is supported by NSF under the award DGE-1419295, DUE-1202690, and AAMU Deans Award2013 through Luck Martin Co.. Many thanks to valuable input from reviewers.

References

[1] H. Weber, *Android now has over 1B active users, up from 538M last year,* VB, 2015.

[2] P. Thornton and C. Houser, "Using Mobile Phones in Education," in *The 2nd IEEE International Workshop on Wireless and Mobile Technologies in Edcuation (WMTE'04)*, Washinton, DC, USA, 2004.

[3] L. S. Vygotsky, "Interaction between learning and development," in *Mind in society. The development of higher psychological processes*, M. Cole, V. John-Steiner, S. Scribner and E. Souberman, Eds., Cambridge, MA, Harvard University Press, 1978, pp. 79-91.

[4] K. R. Clark, *Examining the effects of the flipped model of instruction on student engagement and performance in the secondary mathematics classroom: An action research study,* Capella University, 2013.

[5] F. C. Herreid and N. A. Schiller, "Case Study: Case Studies and the Flipped Classroom," *Journal of College Science Teaching,* vol. 42, no. 5, pp. 62-67, 2013.

[6] R. S. Davies, D. L. Dean and N. Ball, "Flipping the classroom and instructional technology integration in a college-level information systems spreadsheet course," *Educational Technology Research and Development,* vol. 61, no. 4, pp. 563-580, June 2013.

[7] P. Pimpale, R. Rayarikar and S. Upadhyay, "Modifications to AES Algorithm for Complex Encryption," *International Journal of Computer Science and Network Security,* vol. 11, no. 10, pp. 183-186, 2011.

[8] W. Stallings, Wireless Communications & Networks, Pearson, 2004.

[9] D. Geneiatakis, N. I. Fovino, I. Kounelis and P. Stirparo, "A Permission verification approach for android mobile applications," *Journal of Computers & Security,* vol. 49, pp. 192-205, 2015.

[10] P. A. Felt, E. Chin, S. Hanna, D. Song and D. Wagner, "Android Permissions Demystified," in *ACM Conference on Computer and Communication Security (CCS)*, 2011.

[11] "Dexter," DEXLab, [Online]. Available: http://dexter.dexlabs.org/. [Accessed 27 8 2015].

[12] "Bluebox," Bluebox, March 2013. [Online]. Available: https://bluebox.com/articles/bluebox-labs-releases-android-malware-analysis-tool/.

[13] C. Talley and S. Scherer, "The Enhanced Flipped Classroom: Increasing Academic Performance with Student-recorded Lectures and Practice Testing in a "Flipped" STEM Course," *The Journal of Negro Education,* vol. 82, no. 3, pp. 339-347, 2013.

[14] J. E. McLaughlin, L. M. Griffin, D. A. Esserman, C. A. Davidson, D. M. Glatt, M. T. Roth, N. Gharkholonarehe and R. J. Mumper, "Pharmacy Student Engagement, Performance, and Perception in a Flipped Satellite Classroom," *American Journal of Pharmaceuticl,* vol. 77, no. 9, p. 196, 2013.

[15] G. S. Wilson, "The Flipped Class: A Method to Address the Challenges of an Undergraduate Statistics Course," *Teaching of*

Psychology, vol. 40, pp. 193-199, July 2013.

[16] X. Zhang, "Mobile Computing Laboratory," [Online]. Available: http://xyzhang.ece.wisc.edu/ece454/#a_assignments. [Accessed 25 August 2015].

[17] S. Helal, "Mobile and Pervasive Computing," [Online]. Available: http://www.cise.ufl.edu/~helal/classes/f14/index.html. [Accessed 25 August 2015].

[18] W. Du, "SEED: Hands-on Lab Exercises for Computer Security Education," in *IEEE SEcurity & Privacy*, 2011.

[19] W. Du, "Hands-on Labs For Security Education," [Online]. Available: www.cis.syr.edu/~wedu/seed. [Accessed 25 August 2015].

[20] P. Tague, "Mobile Security," [Online]. Available: http://wnss.sv.cmu.edu/teaching/14829/f14/. [Accessed 25 August 2015].