International Journal of Smart Home
Vol. 11, No. 1 (2017), pp. 105-116
http://dx.doi.org/10.14257/ijsh/2017.11.1.07

Research and Realization of Security Software for Smart Phone Based on Android Platform

Jiping Li[1], Yaoming Ding[2*], Yongjun Xiao[3] and Shouyin Liu[4]

[1,2]School of Computer and Information Science, Hubei Engineering University, Xiaogan 432000, China
[3]School of Physics and Electronic Information Engineering, Hubei Engineering University, Xiaogan 432000, China
[4]College of Physical Science and Technology, Central China Normal University, Wuhan 430079, China
[1]oucljp@aliyun.com, [2]xgdym2015@aliyun.com, [3]xiaoyj2016@hust.edu.cn, [4]syliu@phy.ccnu.edu.cn

Abstract

In order to reduce the security threat caused by the leakage of personal private information due to the stolen or lost smart phone, design scheme of security software for smart phone based on Android platform is proposed in this paper. In addition, realizing method and corresponding key codes of each module are also presented in detail. Technologies of background monitoring and secure phone number binding are used to realize remote control over the stolen smart phone by sending short controlling message from the phone bound with security number to the stolen smart phone. The security of the stolen or lost smart phone can be protected by performing remote control functions such as anti-theft tracking, voice warning, screen locking and private information wiping. Testing results in both simulation and real machine environment show that the security software is valuable in practice.

Keywords*: privacy leaking; anti-theft tracking; data wiping; voice warning; screen locking*

1. Introduction

With the advent of mobile internet, intelligent mobile terminals such as smart phone have become the main devices used to obtain information. Smart phone based on Android has been widely used in our daily life due to its open source, easy to use and powerful function [1-3]. Smart phone is used not only to give somebody a call, to send text messages, to refresh micro-blog, to chat on the Internet, to perform network shopping, to watch video and to realize route navigation, but also used to deal with their personal businesses, such as storing telephone directory, bank account passwords, online payment account, important reminders, video, photos and other personal information and privacy data. Once the smart phone is stolen or lost, criminals maybe use the contact information stored in the phone, posing as bank staff, court functionary, prosecution staff and public security police, defrauding the owner's friends and family members under various pretexts, which will cause some economic losses to the owner and his/her close relatives and friends, and result in major bad impact on their work and daily life [4-6]. Therefore, how to develop security software to protect the privacy information stored in the smart phone from leaking and help to find the lost smart phone becomes particularly important and meaningful [7,8,10].

* Corresponding Author

ISSN: 1975-4094 IJSH

2. System Design

In order to achieve smart phone's anti-theft feature, the security software should have functions such as SIM card replacement warning, lost phone location tracking, screen locking, voice warning and private data wiping. The function framework of the security software for the smart phone is shown in Figure 1.

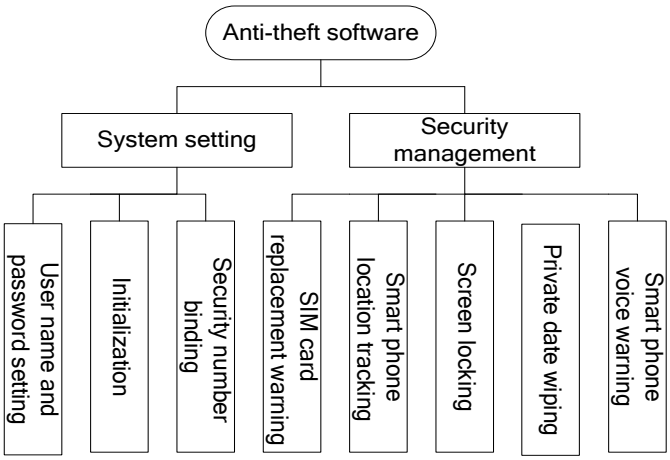


Figure 1. Framework of System's Function

2.1. System Composition

The system is mainly composed of two parts, system setting and security management. The system setting is mainly used to achieve the setting of user name and password, initialization and security number binding when running the software at the first time. Security management is mainly used to realize four main functions for the stolen smart phone, *i.e.*, SIM card replacement warning, smart phone's location tracking, smart phone's screen locking, privacy data wiping and voice warning.

2.2. System Function

When running the security software, the system checks whether it is the first time to run or not. If it is the first time to run, the software goes to the initial setup, and the user should set user name and password for login, and then set a security number. When finishing the phase of initialization setting, the security software automatically begins to function. Meanwhile, the security software saves the user's SIM card information, user name and password. If it isn't the first time to run, user authentication process is needed if somebody hopes to perform software setting. Only when the inputting user name and password are both correct, the user can go to interface of software settings, achieving closing security features, modifying the login user name and password. If the entered user name or password is not correct, the user can re-input them, however the re-input operation cannot exceed three times. If the number of user's logging failure is more than three, the security software will exit from the logging interface and the security software will run in the background and keep monitoring status. According to the received short messages such as "location", "lockscreen", "alarm" and "wipedata" sent by the mobile phone bound with preset security number, the stolen smart phone will perform positioning and tracking, screen locking, voice warning and privacy wiping operations, respectively.

When the stolen smart phone's SIM card is replaced by others, the stolen smart phone will send SIM changing information and the new SIM card information to the mobile phone bound with the security number. When the phone bound with security number send short message (SM) "location" to the stolen smart phone, the stolen smart phone

will send its latitude and longitude information to mobile phone bound with the security number. Similar to this, when receiving the SM “lockscreen”, the stolen smart phone will lock its screen to prevent unauthorized users from viewing or processing related information; When receiving the SM “alarm”, the stolen smart phone will automatically play voice warning messages pre-stored in the stolen smart phone; When receiving the SM “wipedata”, the stolen smart phone will wipe the private data stored in the stolen smart phone. Specific processes are shown in Figure 2.

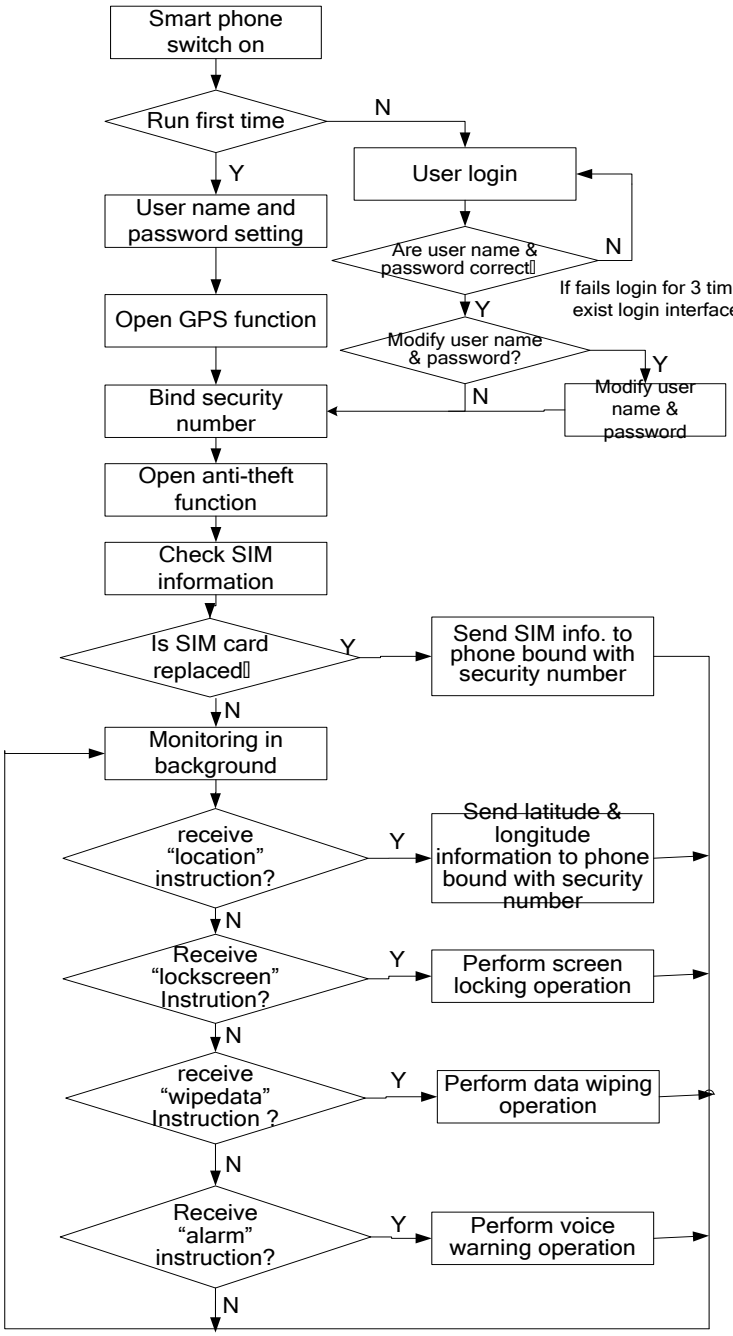


Figure 2. Flow Chart of System Function

3. System Realization

3.1. Initialization Module of the System

The main function of initialization model is used to allow users to set the login name and password when first running the security software, and to allow user to save login name and password. In Android platform, there are mainly three kinds of data storage ways: the first is to use SQLite database, the second to use Shared-Preferences and the third to use file storage [9]. Due to the small amount of data to be stored, the security software uses Shared-Preferences to save user name and password, the security number and SIM card information.

3.2. User Management Module

The function of user management module is mainly used to perform user authentic-cation when user login the system for the first time. If the inputting login name and password match well with the user name and password set in the initialization phase, the user can enter into the interface of anti-theft function and can reset user name as well as password. The core codes of this module are as follows.

```
String savedUsername=sp.getString("username", " ");
String savedPassword=sp.getString("password", " ");
if(username.trim().equals(savedUsername)&&password.trim().equals(savedPassword))
   {  init();  showPasswordDialog.dismiss (); }
 else
    { Toast.makeText(ActivityMain.this,"User name or password error", Toast.LENGTH_LONG).show();
  return; }
```

3.3. SIM Card Bind Module

Once a smart phone is bound with specified SIM card, if the SIM card is replaced by a new one, the smart phone will send SIM card replacement information to the mobile phone bound with security number, and also send the information of the replaced SIM card to the security-number-bound phone. The core codes of SIM card binding are shown as follows.

```
siv_simBind.setOnClickListener(new OnClickListener()
{  public void onClick(View v)
   { String simSerialNumber = tm.getSimSerialNumber();
    if(siv_simBind.isChecked())
        { siv_simBind.setChecked(false);
          sp.edit().putString("sim","").commit();}
    else{siv_simBind.setChecked(true);
          sp.edit().putString("sim",simSerialNumber).commit();}
    }
};
```

3.4. Security Number Binding Module

The main function of security number binding module is enable user to set up a security number. When the smart phone is stolen or lost, the owner can send SM commands such as “location”, “lockscreen”, “alarm” and “wipedata” to the stolen smart phone, realizing remote control and management of the stolen smart phone. The core codes which realize the security number binding are as follows.


```
protected void showNext()
{   String et_safephone=et_phone.getText().toString().trim();
   if(TextUtils.isEmpty(et_safephone))
   {  Toast.makeText(this, "Set security number please!", 0).show();
      return; }
   sp.edit().putString("safephone", et_safephone).commit();
   Intent intent = new Intent (this,Setup4Activity.class);
   startActivity(intent); finish();
   overridePendingTransition(R.anim.tran_in, R.anim.tran_out);
}
```

3.5. SIM Card Checking Module

The function of SIM card checking module is enable the system to check whether the bound SIM card is replaced or not. If the SIM card is replaced, which means the smart phone is stolen or lost, the smart phone will send SIM card replacement information and the new SIM card information to the mobile phone bound with security number. After system booting, Android will send BOOT_COMPLETED broadcasting message to inform various states information after finishing system starting up [11]. With Boot Complete-Receiver listening BOOT_COMPLETED broadcasting events in background, the instance of TelephonyManager derives the information of SIM card when booting, and then compares the derived information with SIM card information bound with the smart phone to determine whether the SIM card is replaced or not and to determine whether or not to send the changed SIM card information to the mobile phone bound with security number. The core codes of obtaining SIM card information when booting are as follows.

```
siv_simBind.setOnClickListener(new OnClickListener()
{  public void onClick(View v)
   {  String simSerialNumber = tm.getSimSerialNumber();
      if(siv_simBind.isChecked())
      { siv_simBind.setChecked(false);
        sp.edit().putString("sim","").commit();}
      else{  siv_simBind.setChecked(true);
             sp.edit().putString("sim",simSerialNumber).commit(); }
   }
};
```

3.6. Locating and Tracking Module

The main function of locating and tracking module is used to realize the stolen smart phone's locating and tracking service, providing guidance for tracking the stolen smart phone. When the GPS location service runs, the latitude and longitude information of stolen phone can be derived by calling public void onLocationChanged (Location location) method. The core codes of location tracking of stolen smart phone service are as follows.

```
Public void onLocationChanged(Location location) {
String longitude="longitude:"+location.getLongitude() +"\n";
String latitude = " latitude:" + location.getLatitude()+"\n";
String accuracy= "accuracy:" + location.getAccuracy()+"\n";
String lastlocation = longitude+latitude+accuracy;
sp.edit().putString("lastlocation", lastlocation).commit();}
```

3.7. Private Information Wiping Module

The main function of private information wiping module is to enable user to send "wipedate" SM command to the stolen or lost smart phone by using the phone bound with security number. When receiving SM command "wipedata", some private information such as contacts, short message, and photos stored in the smart phone can be deleted. In Android system, the information of contacts is stored in the directory/data/data/com. android.providers.contacts/[11, 12], however, the information cannot be accessed directly. After obtaining permission, contacting information in the address book can be derived by using "insert" interface of ContenResolver, and then the contacting information is wiped by using "delete" interface of ContentRe-solver. However, the storage path of SD card is /mnt/sdcard [11, 12]. If a user hopes to delete the data stored in the SD card, he must get the permission first and then realize deleting operation through file operation.

To delete the contacts of the stolen smart phone, we must first get permission to read and write contacts. If we hope to get the read-write permission, we must add the following codes in configuration file AndroidManifest.xml.

```
<uses-permission andriod:name=android.permission.READ_CONTACTS/>
<uses-persmission android:name=android.permission.WRITE_CONTACTS/>
```

The core codes of deleting the contacts of the stolen smart phone are as follows.

```
ContentResolver cr=getContentResolver();
int n=cr.delete(ContactsContract.Data. CONTENT_URI,null,null);
```

To delete the information stored in SD card of the stolen smart phone, written permission of SD card should be obtained firstly. If we hope to get the written permission, we must add the following code in configuration file AndroidManifest.xml.

```
<uses-permission android:name=android.permission.WRITE_EXTERNAL_STORAGE/>
```

After getting the written permission of SD card, we can derive different kinds of information by performing related operation. The following codes demonstrate deleting all pictures stored in the directory of "Pictures" in smart phone's SD card.

```
String baseDir =Environment.getExternalStorageDirectory().get AbsolutePath();
baseDir=baseDir+"/Pictures";
File[] list=new File(baseDir).listFiles();
for(int i=0;i<list.length;i++) list[i].delete();
```

3.8. Background Short Message Monitoring Module

The main function of the background SM monitoring module is used to constantly monitor the received SM of smart phone in background. If SM sent by mobile phone bound with security number arrives, the stolen smart phone will intercept the message content, and compare the SM content with remote control command. If the SM content matches well with the remote control command, the stolen smart phone will implement some operations such as anti-theft tracking, screen locking, voice warning and private data wiping respectively according to the matched SM content. The core codes of realizing this function are as follows.

```
public class SMSReceiver extends  BroadcastReceiver
{  private static final String TAG = "SMSReceiver";
   private SharedPreferences sp;
   private DevicePolicyManager dpm;
```



```
public void onReceive(Context context, Intent intent) {
sp=context.getSharedPreferences("config",context.MODE_PRIVATE);
dpm=(Devi olicyManager)context.getSystemService(context.DEVICE_POLICY_SERVICE);
Object[] objs = (Object[]) intent.getExtras().get("pdus");
for (Object obj : objs)
{   smsMessage sms=smsMessage.createFromPdu((byte[]) obj);
    String address = sms.getOriginatingAddress();
    String safephone = sp.getString("safephone", "5556");
    if(address.contains(safephone))
    {  String body = sms.getMessageBody();
       if ("location".equals(body))
          {  Intent intentGPSService = new Intent(context,GPSService.class);
             context.startService(intentGPSService);
             String lastlocation = sp.getString("lastlocation", " ");}
       if(TextUtils.isEmpty(lastlocation))
        smsManager.getDefault().sendTextMessage(addr ess, null,"getting location...from zh", null, null);
         else{        SmsManager.getDefault().sendTextMessage(address,  null,  lastlocation,  null,  null);
                abortBroadcast(); }
       elseif("alarm".equals(body)){
                MediaPlayer mp=MediaPlayer.create(context, R.raw.fail);
                mp.setVolume(1.0f, 1.0f); mp.start(); abortBroadcast();}
       elseif("wipedata".equals(body)){ dpm.wipeData(0);abortBroadcast();}
       elseif("lockscreen".equals(body))
            {   dpm.lockNow ();  dpm.resetPassword(safephone, 0);  abortBroadcast();}
}
```

4. Software Testing

The main purpose of software testing is to verify whether the software interface coincides with designing effects, whether or not the running results are good, whether or not the function is perfect and whether or not the performance of the software is stable. In order to verify the running effects of mobile security software, simulator and real machine environment are both established for software testing. The software and hardware environment for testing are shown in Table 1.

Table 1. Software and Hardware Environment of Testing

Software and hardware	Configuration
OS of desktop	Windows 7
JDK	1.7
Android SDK	4.0.3
Programming software	Eclipse Kepler 4.3
Phone model	HTC D826w
Phone memory	2GB
OS of Phone	Android 5.0
Phone driver	HTC D826w driver

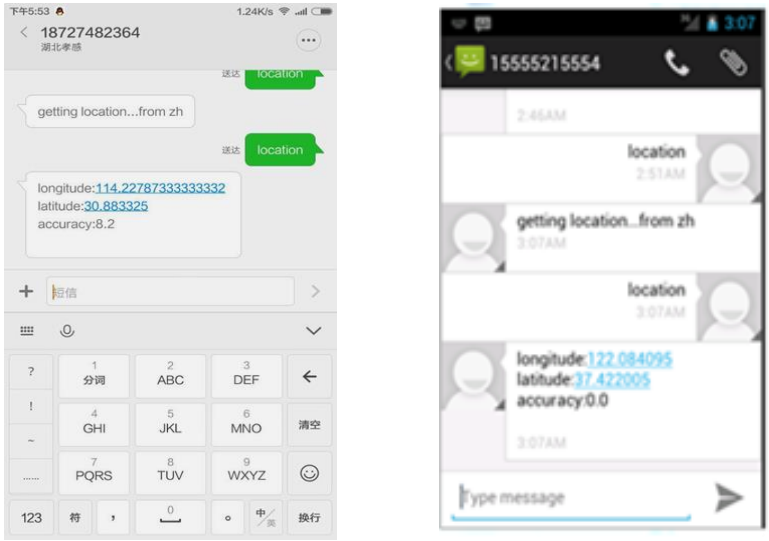
4.1. Testing Platform

The hardware components of the testing platform are as follows. A desktop with a USB2.0 interface, a HTC D826w smart phone based on Android 5.0, a USB data cable, HTC D826w is connected with the desktop by the USB data cable. Through the USB interface, the desktop and the smart phone work synchronously.

The software components of the testing platform are as follows. Eclipse Kepler 4.3, Android ADT 1.7 Integrated Development Environment, Android 4.0.3 SDK, and Driver of HTC D826w smart phone. In the debug module of computer system, the desktop is connected with HTC D826w smart phone and works synchronously. Screenshots of software running can be obtained by device of DDMS in Eclipse. Meanwhile the running results can be analyzed by LogCat of integrated environment.

4.2. Testing Results

Regardless of the testing environments in Android emulator or in real machine, the four main functions proposed in requirements analysis can be achieved. When the smart phone bound with security number (or Android emulator) sends “location” SM command to the stolen smart phone (or Android emulator equipped with this antitheft applications), the phone bound with security number (or Android emulator) will receive the stolen smart phone's location information. The testing result of location and tracking is shown in Figure 3.



a. In real machine testing environment. b. In emulator testing environment

Figure 3. The Testing Results of Anti-theft Tracking

When the mobile phone bound with security number (or Android emulator) sends “lockscreen” SM command to the stolen smart phone (or Android emulator equipped with this anti-theft applications), the screen of the stolen smart phone is locked as shown in Figure 4.

a. In real machine testing environment. b. In emulator testing environment.

Figure 4. Testing Results of Screen Locking

When the mobile phone bound with the security number (or Android emulator) sends "alarm" SM command to the stolen smart phone(or Android emulator equipped with this anti-theft applications), the stolen smart phone plays pre-setting voice file and gives voice warning to the thief.

When the mobile phone bound with the security number (or Android emulator) sends "wipedata" SM command to the stolen smart phone (or Android emulator equipped with this anti-theft applications), the private data such as contacts, photos and messages can be deleted. In order to prevent the data stored in phone and in SD card from being deleted, the data wiping testing is verified in the emulator environment. The test result of private data deleting is shown in Figure 5.

Figure 5. The Testing Result of Private Data Wiping in the Emulator Environment

5. Conclusion

This paper proposes a design scheme based on Android platform for mobile security software, and gives in detailed system functional block diagram of the security software, functional flow diagram and core codes of all functional modules. By using background monitoring technology and by binding security number, the developed security software realizes remote control through short message, realizing the function of anti-theft tracking, voice warning, locking of screen and destroying of private information. The testing

results in both emulator environment and real machine environment show that the security software is valuable in practice. In the next stage, we will focus on the development of anti-theft security software suitable for IOS system, achieving anti-theft tracking, voice warning, private information destroying, screen locking and other related functions.

Acknowledgments

The authors gratefully thank for the helpful and suggestions of reviewers. This work is funded by nature science foundation of Hubei Province, China under Grant No. 2014CFB577, and partly supported by the Natural Science Foundation of China under Grant No. 61370223.

References

[1] Y. Fengsheng, "Android application and development unleashed", China machine press, Beijing, **(2010)**.
[2] D. Jun and Z. Xuesen, "Reseach and Design of the Calls Firewall Based on Android Smart Phone", Journal of Inner Mongolia University of Science and Technology, vol. 31, no. 4, **(2012)**, pp. 356-359.
[3] Y. Mei and P. Xin-guang, "Permissions Detection System Based on Android Security Mechanism", Computer Engineering and Design, vol. 34, no.3, **(2013)**, pp. 854-858.
[4] Z. Hai-jun, "It's Urgent to Solve the Problem of Privacy Leaking of Mobile Phone", China Information World, Feb, 27 (04), **(2012)**.
[5] Anonymity. "Openly and Secretly Strife: Great Battle for the Security of Mobile Phone" [EB/OL]. http: //www.icpcw.com. May **2015**.
[6] L. Chao, L. Xi and L. Jing-lin, "Background Monitoring Technology of Mobile Phone Based on the Android Platform", Computer Knowledge and Technology, vol. 6, no. 33, **(2010)**, pp. 9472-9474.
[7] Z. Hao and C. Sheng-yun, "Realizing the Function of Anti-theft Tracking for Mobile Phone Based on Android Platform", Jiangxi Science, vol. 29, no. 5, **(2011)**, pp. 652-655.
[8] L. Zhong-ping, "Key Technology Analysis of Remote Control Based on Android Mobile Phone", Computer Applications and Software, vol. 30, no. 4, **(2013)**, pp. 113-115.
[9] Y. Feng-sheng, "Inside Story of Android Technology. System Volume", China Machine Press, Beijing, **(2011)**.
[10] W. Yin and L. Wei-yao, "Mobile Phone Security and Data Protection Based on Android Platform", Modern Computer, no. 27, **(2013)**, pp. 62-64.
[11] N. Qin-bo, M. De-jun and H. Yan-yan, "Design and Realization of Anti-burglary Software for Mobile Phone Based on Android Platform", Modern Electronics Technique, vol. 38, no. 4, **(2015)**, pp. 46-49.

Authors

Jiping Li, was born in Hubei Province, China, in 1972. He Received B.S. degree in application of computer from Hubei University in 2011 and M.S. degree in application of computer from Ocean University of China, Qingdao in 2006. He received Ph.D. degree in radio physics from Central China Normal University, Wuhan in 2012. He is presently an associate professor in computer science of Hubei Engineering University. His research interests include network security, wireless resource allocation management and application of internet of things.

Yaoming Ding, was born in Hubei Province, China, in 1963. He Received B.S. and M.S. degree in physic science from Central China Normal University, Wuhan, in 1986 and in 2000 respectively. He received Ph.D. degree in Huazhong University of Science and Technology in 2011. He is presently a professor in physic science in Hubei Engineering University. His research interests include optical communication and security of wireless communication.

Yongjun Xiao, received his Ph.D. degree in communication and information system from Wuhan University. He is postdoc in Wuhan National Laboratory for Optoelectronics (WNLO) now, and his main research interest focusing on fabricating of application for the self-powered sensors.

Shouyin Liu, was born in Henan Province, China, in 1963. He received B.S. degree in physics and M.S. degree in radio electronics from Central China Normal University, Wuhan, China, in 1985 and 1988, respectively. He received the Ph.D. degree from Hanyang University, Korea in 2005 in electronic communication engineering. From 2004, he has been a professor at Central China Normal University. His current research interests include digital communication, wireless sensor network and location techniques.