Vulnerability in *Easy Filter* for Android

Daoyuan Wu*, Xiapu Luo* and Rocky K. C. Chang
The Hong Kong Polytechnic University
{csdwu, csxluo, csrchang}@comp.polyu.edu.hk
December 2, 2011

Abstract

We found that *Easy Filter* 2.0.8 and 2.0.10 have a vulnerability that allows a malicious application to access and manipulate the user's call records, sms and the blacklists set by the user.

1 Application Information

Package Name	com.phoneblocker.android
Full Name	Easy Filter
Version	1.1 and 1.2 (the latest version in the Android market)
Category	Communication
Installs	50,000 - 100,000
Average Rating	3.5/5.0 from 402 users

CVE Reference	CVE-2011-4698
Vendor	*AndroidAppTools*, https://market.android.com/developer?pub=AndroidAppTools
Vendor Response	Null

2 Description

Easy Filter (EF) exposes the following content provider in the AndroidManifest.xml file, which are not properly protected:

```
<provider
android:name="com.phoneblocker.android.provider.EasyFilterProvider"
android:authorities="com.easyfilter.provider.EasyFilterProvider" />
```

Through the provider com.phoneblocker.android.provider.EasyFilterProvider, a malicious application on the same device can query, modify and delete the call records, sms and the blacklists set by the user, and insert new records, sms, and blacklists.

*** authors with equal contributions**

3 Impact

This vulnerability enables an adversary to access and manipulate the user's call records, sms and the blacklists. For example, a malicious application on the same device can obtain a contact's sms, as shown in Figure 1, without being noticed by the user. The vulnerability also allows an attacker to access the record of the user's phone call and other information, as shown in Figure 2. Although this application helps the user filter unwanted calls using blacklist that is protected by password, an attacker can bypass this protection and manipulate the blacklist, as shown in Figure 3. For example, she can delete a record so that spam could come or add a record to launching the DoS attack.

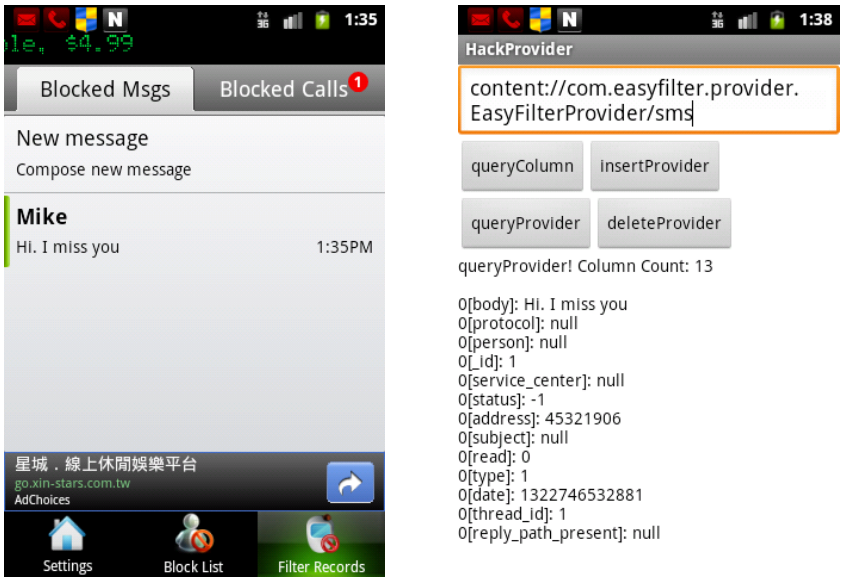

Figure 1: User's sms are exposed.

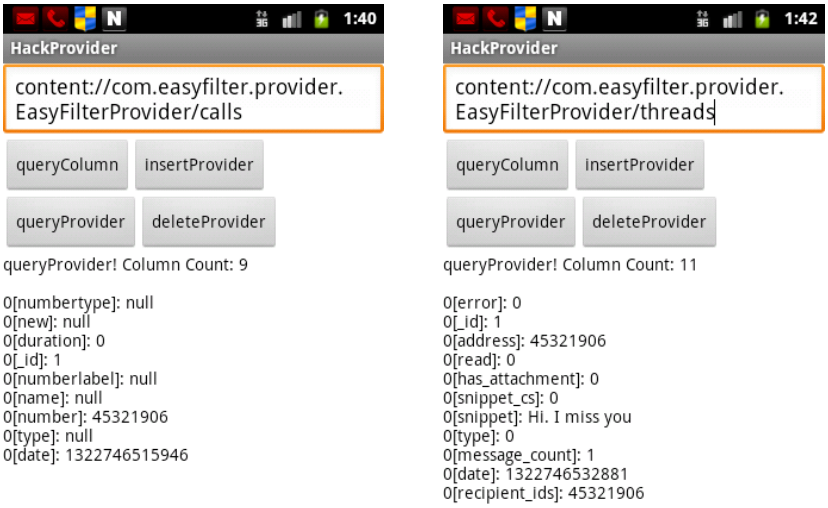

Figure 2: User's phone call and other information are exposed.

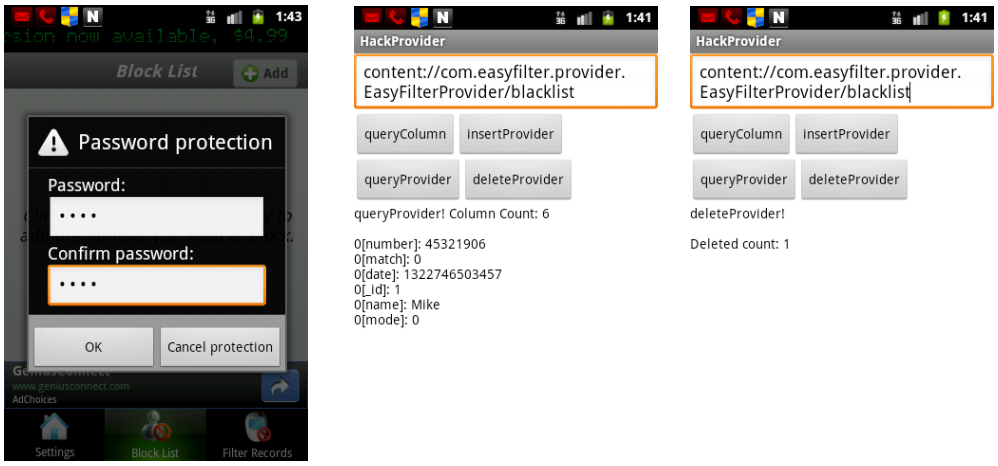

Figure 3: Manipulate the black lists protected by password

4 Solution

We are trying our best to contact *AndroidAppTools* to fix this security issue. Our advice is to set the permission of these application's content providers properly, or avoid exporting these content providers in the AndroidManifest.xml file. Currently, a user could disable the application temporarily and wait for an official update.

5 Technical Description

The vulnerable content provider and the corresponding table i listed as follows:

Content Provider Authority	Table Name
com.phoneblocker.android;com.easyfilter.provider.EasyFilterProvider	blacklist
com.phoneblocker.android;com.easyfilter.provider.EasyFilterProvider	calls
com.phoneblocker.android;com.easyfilter.provider.EasyFilterProvider	sms
com.phoneblocker.android;com.easyfilter.provider.EasyFilterProvider	threads

Sample attack codes:

```
providerUri = Uri.parse("content://
com.phoneblocker.android;com.easyfilter.provider.EasyFilterProvider/sms")
ContentResolver cr  = this.getContentResolver();

//Delete sms
int nCount = cr.delete(providerUri, null, null);

//Query sms
Cursor cursor = cr.query(providerUri, null, null, null, null);

//Insert sms
ContentValues values = new ContentValues();
```

```
....
outUri = cr.insert(providerUri, values);
```