Input Validation Issues- Part 1

Aakash R CB.SC.P2CYS23011

We explore the application by entering values in the search EditText field

7. Input Validation Issues - Part 1 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. Admin

User: (admin) pass: (passwd123) Credit card: (1234567812345678)

We observe the decompiled source code in the JADX.

```
diva-beta.apk
                                    ## AndroidManifest.xml ×
                                                                                                       MainActivity
                                                                # res/layout-v21/content_main.xml ×
  In Inputs
                                                      Intent i3 = new Intent(this, (Class<?>) InsecureDataStorage1Activity.class);
startActivity(i3);
   I Files
                                    100

diva-beta.apk

    M Scripts
                                                  if (view == findViewById(R.id.d4button)) {
Source code
                                                      Intent i4 = new Intent(this, (Class<?>) InsecureDataStorage2Activity.class);
                                     61
  > 🖿 android.support
                                                      startActivity(i4);
                                    100
                                                      return;
  v 🖿 jakhar.aseem.diva
    AccessControl1Activity
                                                  if (view == findViewById(R.id.d5button)) (
   Intent i5 = new Intent(this, (Class<?>) InsecureDataStorage3Activity.class);
    S AccessControl2Activity
    AccessControl3Activity
                                                      startActivity(i5);
    AccessControl3NotesAct
                                    100
                                                      return:
    APICreds2Activity
                                     68
                                                  if (view == findViewById(R.id.d6button)) (
    APICredsActivity
                                                      Intent i6 = new Intent(this, (Class<?>) InsecureDataStorage4Activity.class);
     > 🧠 BuildConfig
                                                      startActivity(i6);
                                    100
                                                      return;
    🗦 🕵 DivaJni
    S Hardcode2Activity
                                                  if (view == findViewById(R.id.d7button)) {
    S HardcodeActivity
                                                      Intent i7 = new Intent(this, (Class<?>) SQLInjectionActivity.class);
                                                      startActivity(i7);
    S InputValidation2URISch
    S InputValidation3Activi
                                                      view == findViewById(R.id.d8button)) {
Intent i8 = new Intent(this, {Class<?>} InputValidation2URISchemeActivity.class);
    S InsecureDataStorage1Ac
    > 🕵 InsecureDataStorage2Ac
                                                      startActivity(i8);
    S InsecureDataStorage3Ac
    See InsecureDataStorage4Ac
     S LogActivity
                                                  if (view == findViewById(R.id.d9button)) {
                                                      Intent i9 = new Intent(this, (Class<?>) AccessControllActivity.class);
startActivity(i0);
      MainActivity
                                     81
```

We understand that for this Task an activity called **SQLInjectionActivity** is used.

```
diva-beta.apk
                                                                                             I Inputs
                                                            package jakhar.aseem.diva;
    □ Files
                                                           import android.database.Cursor;
import android.database.sqlite.SQLiteOatabase;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.vitl.l.og;
import android.view.View;
import android.view.teitText;
import android.widget.Totat;
          diva-beta.apk
      N⊓ Scripts
  Source code

≥ android.support

    jakhar.aseem.diva

           AccessControllActivity
         AccessControl2Activity
         ## AccessControl3Activity
## AccessControl3NotesAct
### AccessControl3NotesAct
### Public class SQLInjectionActivity extends AppCompatActivity (
### private SQLInjectionActivity extends AppCompatActivity (
          APICreds2Activity
                                                                 /* JADX INFO: Access modifiers changed from: protected */
##Override // android.support.v4.app.AppCompatActivity, android.support.v4.app.FragmentActivity, android.support.v4.app.BaseFragmentActivityDopublic void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
         APICredsActivity
          BuildConfig
         DivaJni
          Hardcode2Activity
                                                                             (
this.mOB = openOrCreateOatabase("sqli", 0, null);
this.mOB.execSQL("DADO TABLE IF EXISTS sqliurer;");
this.mOB.execSQL("DADO TABLE IF EXISTS sqliurer;");
this.mOB.execSQL("INSERT INTO sqliuser VALUES ("admin", "password23", "1234587812345878");");
this.mOB.execSQL("INSERT INTO sqliuser VALUES ("idmin", "password23", "1234587812345878");");
this.mOB.execSQL("INSERT INTO sqliuser VALUES ("john", "password22", "5555666677778888");");
         % HardcodeActivity 21
% InputValidation2URISch 22
% InputValidation3Activi 24
         InsecureDataStorage1Ac 25
InsecureDataStorage2Ac 26
         InsecureDataStorage3Ac
InsecureDataStorage4Ac
                                                                     catch (Exception e) (
  Log.d("Diva-sqli", "Error occurred while creating database for SQLI: " * e.getMessage());
          LogActivity
                                                  32
                                                                       setContentView(R.layout.activity_sqlinjection);
         MainActivity
                                                               NotesProvider
% mD8 SQLiteDatabase
% onCreate(Bundle) voi
% search(View) void
42
```

We enter 1' OR 1=1-- so that the above SQL query becomes

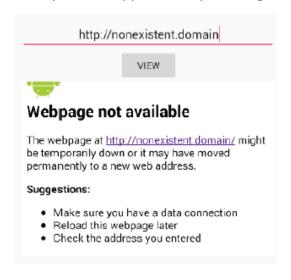
SELECT * FROM sqliuser WHERE user ='1' OR 1=1--'

The WHERE clause condition gets evaluated to FALSE or TRUE which is equivalent to TRUE, hence all the records in the database are displayed in the Toast message.



Input Validation Issues- Part 2

We explore the application by entering values in the URL EditText field.



We enter a sensitive path like

file:///data/data/jakhar.aseem.diva/shared_prefs/jakhar.aseem.diva_preferences.xml

Which only the application has access to and normal user of the device does not have access to. We observe that the file contents are displayed in the WebView.

```
root@generic_x86:/ # su
data/data/jakhar.aseem.diva/
root@generic_x86:/data/data/jakhar.aseem.diva # ls
app_webview
cache
databases
lib
root@generic_x86:/data/data/jakhar.aseem.diva # id
uid=0(root) gid=0(root)
root@generic_x86:/data/data/jakhar.aseem.diva # pwd
/data/data/jakhar.aseem.diva
root@generic_x86:/data/data/jakhar.aseem.diva # ls
app_webview
cache
databases
lib
root@generic_x86:/data/data/jakhar.aseem.diva # ls
app_webview
cache
databases
```

.aseem.diva/shared_prefs/jakhar.aseem.div

VIEW

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<map>
    <string name="user">a</string>
    <string name="password">12345678</string>
    </map>
```

We observe the decompiled source code and open the InputValidation2URISchemeActivity in the JADX.

We observe that the user input value in the EditText field is used directly to load in the WebView without any sanitization or validation.

```
diva-beta.apk
                                          InputValidation2URISchemeActivity
  ™ Inputs
                                               package jakhar.aseem.diva;
    N Files
                                               import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.webkit.WebSettings;
import android.webkit.WebView;
import android.widget.EditText;

    diva-beta.apk

     N Scripts
 Source code
   android.support

  jakhar.aseem.diva

     AccessControl1Activity
                                                /* loaded from: classes.dex */
     > % AccessControl2Activity | 12 | public class InputValidation2URISchemeActivity extends AppCompatActivity {
     > @ AccessControl3Activity
                                                    /* JADX INFO: Access modifiers changed from: protected
     AccessControl3NotesAct
                                                   <code>@Override // android.support.v7.app.AppCompatActivity, android.support.v4.app.FragmentActivity, a public void onCreate(Bundle savedInstanceState) (</code>
     super.onCreate(savedInstanceState);
     setContentView(R.layout.activity_input_validation2_urischeme);
                                                        WebView wview = (WebView) findViewById(R.id.ivi2wview);
WebSettings wset = wview.getSettings();
     🤉 🕵 BuildConfig
     > @ DivaJni
                                                        wset.setJavaScriptEnabled(true);
     > 🧠 HardcodeActivity
                                                 public void get(View view)
     S InputValidation2URISch 23
                                                       EditText uriText = (EditText) findViewById(R.id.ivi2uri);
WebView wview = (WebView) findViewById(R.id.ivi2wview);
wview.loadUrl(uriText.getText().toString());

get(View) void

           @ onCreate(Bundle) voi
     S InputValidation3Activi
     🦠 🕵 InsecureDataStorage1Ac
```