**Secure HTTP request**

Description:

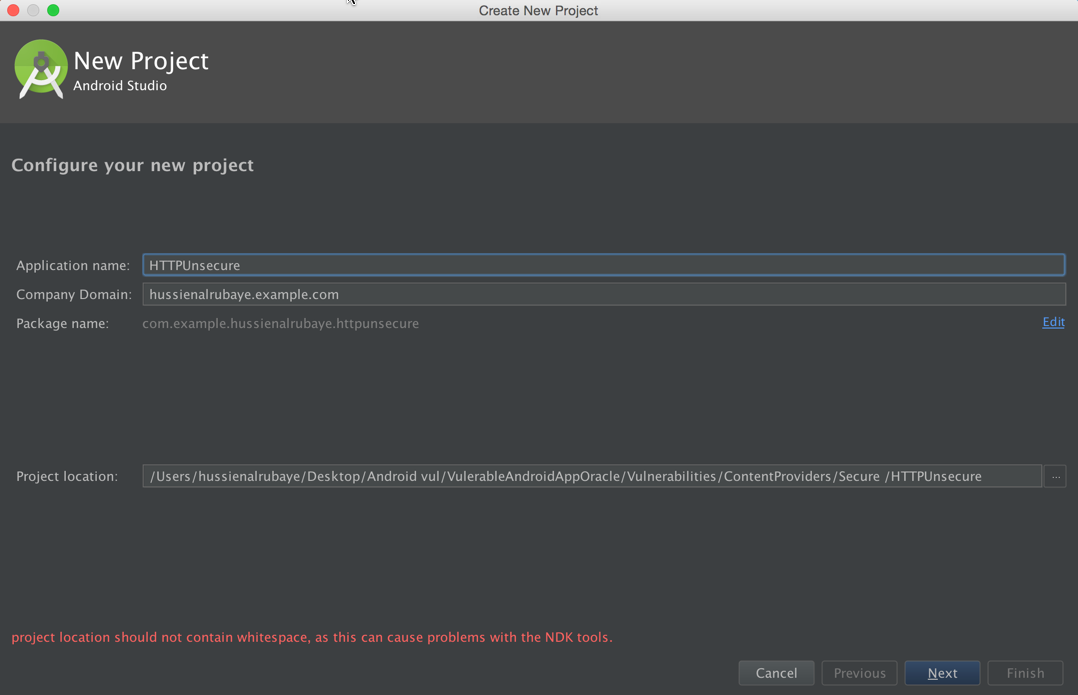
In Android, we use HTTP to exchange data between client and server. We use HTTP to send deferent data sometimes We send sensitive data like user name and password or user location from client to server over HTTP. We have to know that this Request will move over the network, When this data is moving over network it very easy to hacker to interrupt the request and read the data that HTTP handle, and many people could interrupt and read this request data, like internet service provides, so we have to protect it

We will demonstrate example how to send sensitive data between client to server over HTTP, then we will explain how hackers could read this data then we protect it.

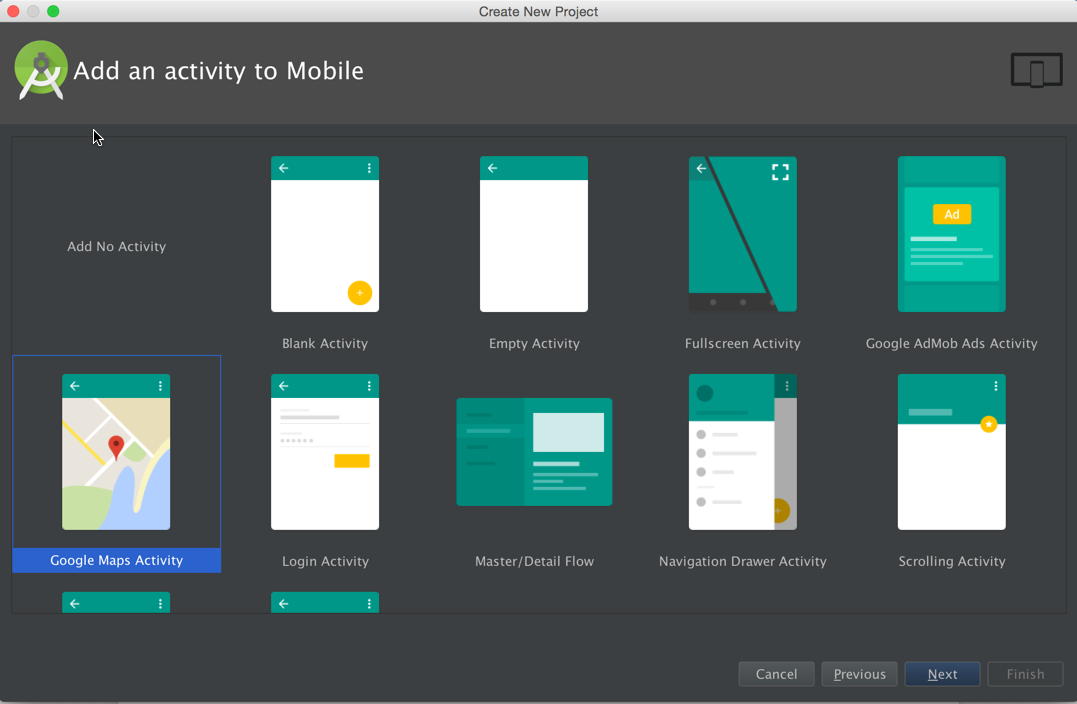


**Steps to work**

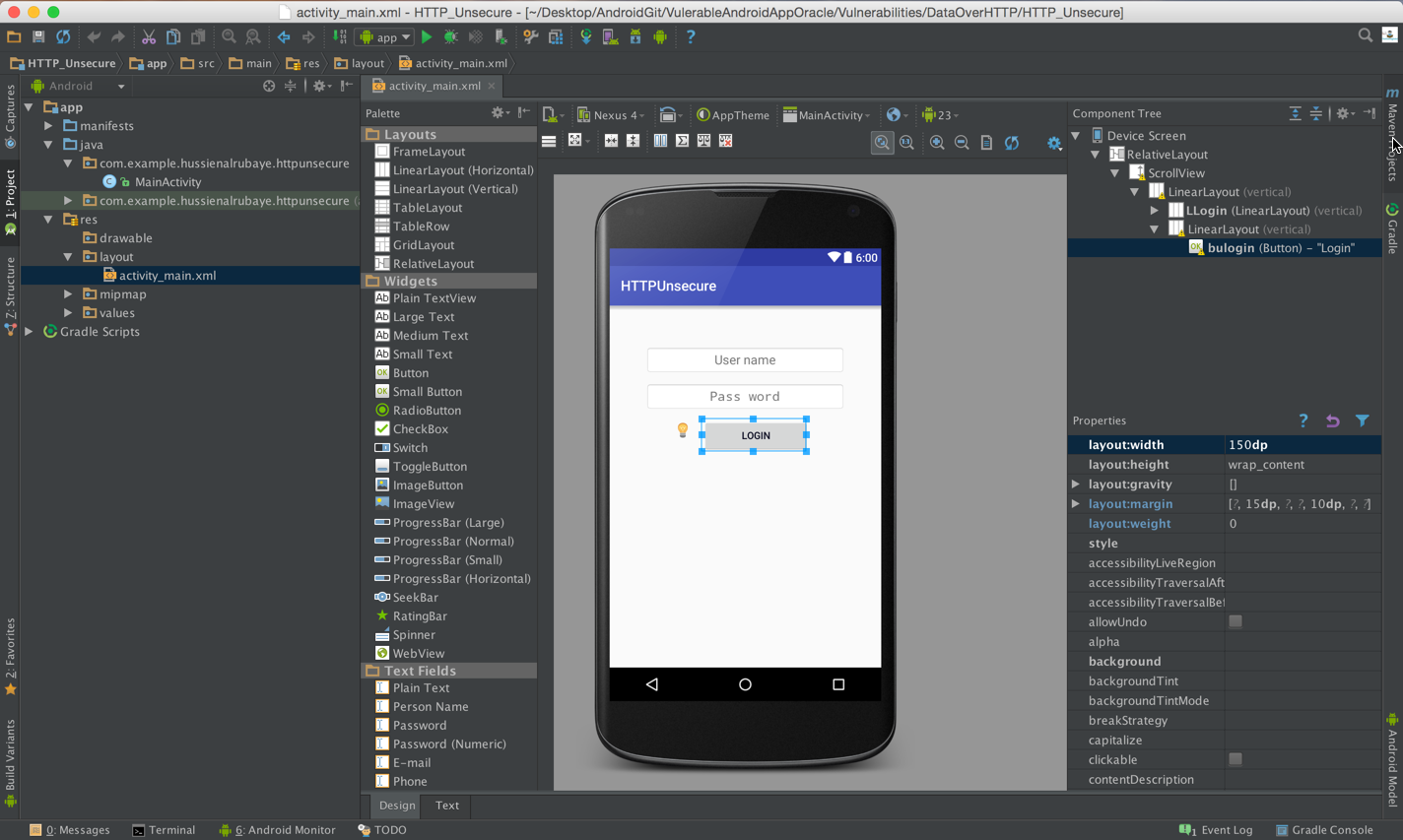
1-Create new project, named “HTTPUnsecure”



2- select project type Empty Activity



3- we build project like this



4- update **activity\_main.xml** to be like this

|  |
| --- |
| Java |
| <?xml version="1.0" encoding="utf-8"?> <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:paddingBottom="@dimen/activity\_vertical\_margin"  android:paddingLeft="@dimen/activity\_horizontal\_margin"  android:paddingRight="@dimen/activity\_horizontal\_margin"  android:paddingTop="@dimen/activity\_vertical\_margin"  tools:context="com.example.hussienalrubaye.httpunsecure.MainActivity">   <ScrollView  android:layout\_width="fill\_parent"  android:layout\_height="wrap\_content"  android:layout\_weight="1">   <LinearLayout   android:orientation="vertical"  android:layout\_width="fill\_parent"  android:layout\_height="fill\_parent"  android:paddingTop="30dp"  android:paddingLeft="15dp"  android:paddingRight="15dp">   <LinearLayout  android:orientation="vertical"  android:layout\_width="fill\_parent"  android:layout\_height="wrap\_content"  android:id="@+id/LLogin"  android:layout\_marginTop="7dp"  android:paddingLeft="20dp"  android:paddingTop="4dp"  android:paddingRight="20dp">   <LinearLayout  android:orientation="vertical"  android:layout\_width="fill\_parent"  android:layout\_height="wrap\_content"  android:touchscreenBlocksFocus="false"  android:layout\_marginBottom="2dp">   <EditText  android:gravity="center"  android:maxLength="50"  android:layout\_width="fill\_parent"  android:layout\_height="wrap\_content"  android:inputType="text"  android:ems="10"  android:id="@+id/EDTUserName"  android:layout\_weight="1"  android:textColor="#ff1a102c"  android:background="@android:drawable/editbox\_background"  android:textSize="18dp"  android:hint="User name"  android:paddingBottom="9dp"  android:paddingTop="9dp"  android:layout\_marginBottom="10dp"/>  </LinearLayout>     <LinearLayout  android:orientation="vertical"  android:layout\_width="fill\_parent"  android:layout\_height="wrap\_content"  android:touchscreenBlocksFocus="false"  android:layout\_marginBottom="2dp">   <EditText  android:gravity="center"  android:maxLength="50"  android:layout\_width="fill\_parent"  android:layout\_height="wrap\_content"  android:ems="10"  android:id="@+id/EDTpassword"  android:layout\_weight="1"  android:textColor="#ff1a102c"  android:background="@android:drawable/editbox\_background"  android:textSize="18dp"  android:hint="Pass word"  android:paddingBottom="9dp"  android:paddingTop="9dp"  android:layout\_marginBottom="10dp"  android:inputType="textPassword" />  </LinearLayout>     </LinearLayout>   <LinearLayout  android:textAlignment="center"  android:gravity="center"  android:orientation="vertical"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent">  <Button  android:layout\_width="150dp"  android:layout\_height="wrap\_content"  android:text="Login"  android:id="@+id/bulogin"  android:drawablePadding="4dp"  android:layout\_marginBottom="10dp"  android:textColor="#ff06071c"  android:onClick="buloginckic"  android:layout\_weight="0"  android:layout\_marginLeft="15dp" />     </LinearLayout>   </LinearLayout>  </ScrollView> </RelativeLayout> |

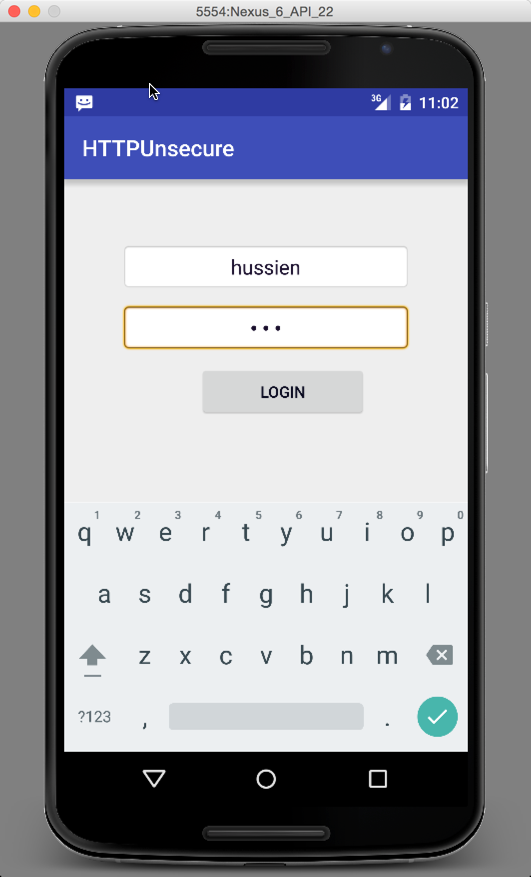
4- update **MainActivity.java** file to be like this

|  |
| --- |
| Java |
| public class MainActivity extends AppCompatActivity {   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_main*);  }   public void buloginckic(View view) {  //get user name and password  EditText UserName=(EditText)findViewById(R.id.*EDTUserName*);  EditText Password=(EditText)findViewById(R.id.*EDTpassword*);  // send user name and password over the http  String url="http://sellingportal.alruabye.net/UsersWS.asmx/Login?UserName="+ UserName.getText().toString() +"&Password="+ Password.getText().toString();  // start background task  new MyAsyncTaskgetNews().execute(url, "news");  }    // get news from server  public class MyAsyncTaskgetNews extends AsyncTask<String, String, String> {  @Override  protected void onPreExecute() {  //before works  }  @Override  protected String doInBackground(String... params) {  // *TODO Auto-generated method stub* try {  String NewsData;  //define the url we have to connect with  URL url = new URL(params[0]);  //make connect with url and send request  HttpURLConnection urlConnection = (HttpURLConnection) url.openConnection();  //waiting for 7000ms for response  urlConnection.setConnectTimeout(7000);//set timeout to 5 seconds   try {  //getting the response data  InputStream in = new BufferedInputStream(urlConnection.getInputStream());  //convert the stream to string  NewsData = *ConvertInputToStringNoChange*(in);  //send to display data  publishProgress(NewsData);  } finally {  //end connection  urlConnection.disconnect();  }   }catch (Exception ex){}  return null;  }  protected void onProgressUpdate(String... progress) {   try {  //display response data  Toast.*makeText*(getApplicationContext(),progress[0],Toast.*LENGTH\_LONG*).show();   } catch (Exception ex) {  }    }   protected void onPostExecute(String result2){    }      }  // this method convert any stream to string  public static String ConvertInputToStringNoChange(InputStream inputStream) {   BufferedReader bureader=new BufferedReader( new InputStreamReader(inputStream));  String line ;  String linereultcal="";   try{  while((line=bureader.readLine())!=null) {   linereultcal+=line;   }  inputStream.close();    }catch (Exception ex){}   return linereultcal;  } } |

4- add access internet permission to Mainfist.xml

|  |
| --- |
| Java |
| <uses-permission android:name="android.permission.INTERNET"></uses-permission> |

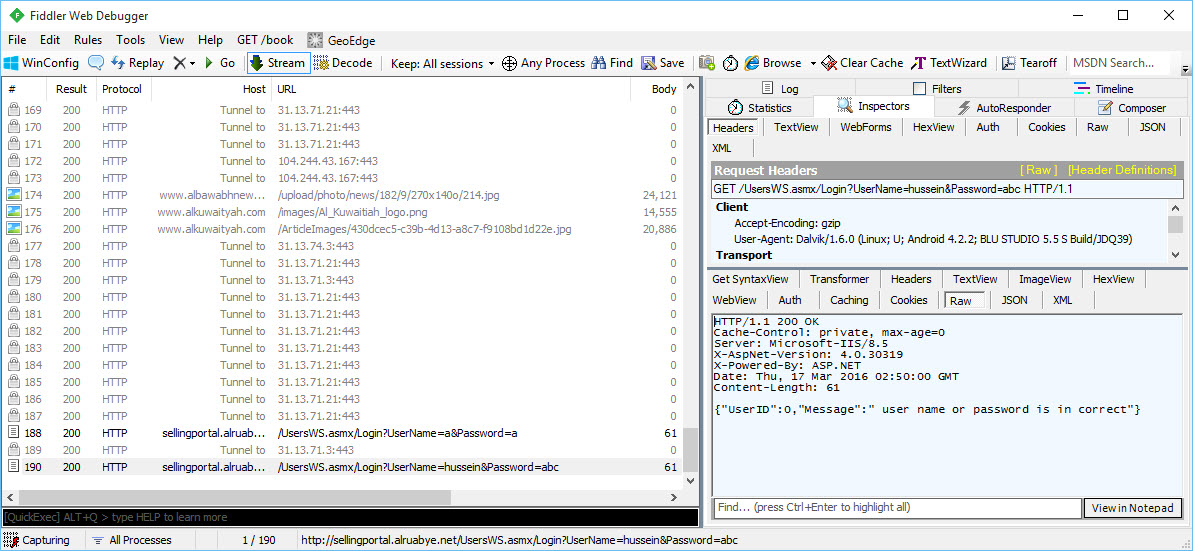
Run the app and enter user name “Hussein” and password “abc”



**What hacker could do,**

When this data is moving over network it very easy to hacker to interrupt the request and read the data that HTTP handle, we will use [Fiddler](http://tech.vg.no/2014/06/04/how-to-monitor-http-traffic-from-your-android-phone-through-fiddler/) app for monitor traffic, click link to install it.

We see that that user name and password is available.



**How to protect our apps,?**

We have to encrypt the the request and use HTTPs when we send the data over the network ,then in the server we decrypt it.

4- update MainActivity.java file to be like this

|  |
| --- |
| Java |
| public class MainActivity extends AppCompatActivity {   @Override  protected void onCreate(Bundle savedInstanceState) {  super.onCreate(savedInstanceState);  setContentView(R.layout.*activity\_main*);  }   public void buloginckic(View view) {  //get user name and password  EditText UserName=(EditText)findViewById(R.id.*EDTUserName*);  EditText Password=(EditText)findViewById(R.id.*EDTpassword*);  // send user name and password over the http  String url="http://sellingportal.alruabye.net/UsersWS.asmx/Login?UserName="+ cipher(UserName.getText().toString(),10) +"&Password="+ cipher(Password.getText().toString(),10);  // start background task  new MyAsyncTaskgetNews().execute(url, "news");  }    // get news from server  public class MyAsyncTaskgetNews extends AsyncTask<String, String, String> {  @Override  protected void onPreExecute() {  //before works  }  @Override  protected String doInBackground(String... params) {  // *TODO Auto-generated method stub* try {  String NewsData;  //define the url we have to connect with  URL url = new URL(params[0]);  //make connect with url and send request  HttpURLConnection urlConnection = (HttpURLConnection) url.openConnection();  //waiting for 7000ms for response  urlConnection.setConnectTimeout(7000);//set timeout to 5 seconds   try {  //getting the response data  InputStream in = new BufferedInputStream(urlConnection.getInputStream());  //convert the stream to string  NewsData = *ConvertInputToStringNoChange*(in);  //send to display data  publishProgress(NewsData);  } finally {  //end connection  urlConnection.disconnect();  }   }catch (Exception ex){}  return null;  }  protected void onProgressUpdate(String... progress) {   try {  //display response data  Toast.*makeText*(getApplicationContext(),progress[0],Toast.*LENGTH\_LONG*).show();   } catch (Exception ex) {  }    }   protected void onPostExecute(String result2){    }      }  // this method convert any stream to string  public static String ConvertInputToStringNoChange(InputStream inputStream) {   BufferedReader bureader=new BufferedReader( new InputStreamReader(inputStream));  String line ;  String linereultcal="";   try{  while((line=bureader.readLine())!=null) {   linereultcal+=line;   }  inputStream.close();    }catch (Exception ex){}   return linereultcal;  }   // cipher encryption add shift for key  public String cipher(String msg, int shift) {  String s = "";  int len = msg.length(); // get string length  for (int x = 0; x < len; x++) {  char c = (char) (msg.charAt(x) + shift); // shift every character  s += c; // append the characters  }  return s;  } } |

**We are Secure now?**

IF we try to login again and run the Fiddler, the request cannot be read.

