## Cryptex Apk

Tools used for this challenge:

- -jadx-gui
- -Sublime text

For solving this challenge, I tried to unpack the apk with 7zip and convert the classes.dex to classes.jar, but it didn't work, the text was unreadable. So, I used **jadx-gui**:

```
File View Navigation Tools Help
  i 🕞 | 🖺 🔓 | 🦫 🕪 | 🌽 🧇 | 🧽 🗳
cryptex.apk
                                                       ⊕ com.dvctf.cryptex.LoginActivity ×
 Source code
                                                             import android.os.Bundle;
import android.util.Log;
import android.view.view;
import android.widget.TextView;
import com.dvctf.droid.R;
    android.support.p063v4
    ⊕ ∰ androidx
    □ m com
                                                             import com.dvctt.droid.R;
import java.security.NessageDigest;
import java.util.Arrays;
import javax.crypto.Cipher;
import javax.crypto.spec.SecretKeySpec;
import p000a.p002b.p003c.ActivityC0013e;
import p064b.p065a.p066a.p067a.outline;
           cryptex 🖶
               # # p111ui.login

    f3746q byte[]
    f3747r byte[]

                                                              public class LoginActivity extends ActivityC0013e {
                      onCreate(Bundle)
                       verify(View) void
           m m droid
                                                                   public static byte[] f3745p = {-114, 62, 98, 26, 54, -7, -59, -47, 55, 88, 18, -1, -99, 116, -51, 62};
        ⊕ ∰ google.android.material
                                                                   /* renamed from: q */
public static byte[] f3746q = {-84, 25, 77, -101, -53, -124, -100, 61, 74, 102, 50, -11, -24, 62, -54, -71};

    ⊕ kotlinx.coroutines.android

    ⊕ ⊕ p000a
   ⊕ # p064b
    ⊕ ⊕ p105c
                                                                   public static byte[] f3747r = {11, -35, 55, 10, 62, 79, 125, 62, -28, 115, 77, 4, 73, 0, 11, 121, -126, 85, -83, 109, 1,
 ## p112d.p113a
## Resources
                                                                               ed from: a12dd3a7fd3203a452eb34d91a9be20569d5e337a3384347068895c07f3e0c5a */
                                                                   public void verify(View view) {
   APK signature
                                                                        String str;

byte[] barr;

TextView textView = (TextView) findViewById(R.id.pass);

TextView textView2 = (TextView) findViewBvId(R.id.message)
```

The main class is **LoginActivity**. Copying, renaming and adapting the code, the class results to be like this:

(see the file in my repo called crypt\_algorithm.java)

To summarize the comparison process:

- -The password entered by us is encrypted in a hash
- -The password is then encrypted in aes. The aes key is created from a value stored
- -At the end, a comparison is performed between our password and a stored password, the original password to guess.

So, to retrieve the right password, we have to:

- -Take the password stored, used to perform the comparison
- -Generate the aes key with the value stored
- -Decrypt the password
- -Show the hash
- -Break the hash

The code used to do that is here: (see the file in my repo called reverse.java)