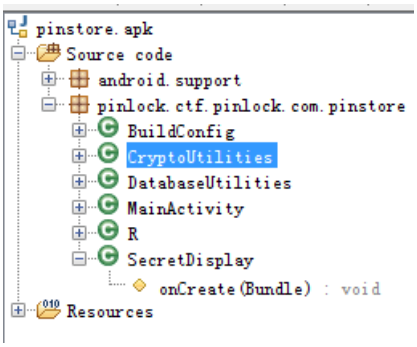
ctf php 读取flag,ctf题： pinstore获取flag

转载

momo呀耶 于 2021-03-11 04:10:43 发布 621 收藏
文章标签： ctf php 读取flag
首先先用jadx反编译出源代码，可以看到目录结构



从这里就可以知道我们只需查看pinlock.ctf.pinlock.com.pinstore目录下的代码即可

先从MainActivity看起

```
    super.onCreate(savedInstanceState);
    setContentView((int) R.layout.activity_main);
    Button button = (Button) findViewById(R.id.loginbutton);
    //public static final int pineditText = 2131492946;
    this.pinEditText = (EditText) findViewById(R.id.pineditText);
    button.setOnClickListener(new OnClickListener() {
        public void onClick(View view) {
            String enteredPin = MainActivity.this.pinEditText.getText().toString();
            String pinFromDB = null;
            String hashOfEnteredPin = null;
            try {
                //从数据库里获取
                pinFromDB = new DatabaseUtilities(MainActivity.this.getApplicationContext()).fetchPin();
            } catch (IOException e) {
                e.printStackTrace();
            }
            try {
                //hashOfEnteredPin为输入的SHA-1加密的内容
                hashOfEnteredPin = CryptoUtilities.getHash(enteredPin);
            } catch (NoSuchAlgorithmException e2) {
                e2.printStackTrace();
            } catch (UnsupportedEncodingException e3) {
                e3.printStackTrace();
            }

            //.equalsIgnoreCase:执行忽略大小写的比较
            if (pinFromDB.equalsIgnoreCase(hashOfEnteredPin)) {
                Intent intent = new Intent(MainActivity.this, SecretDisplay.class);
                intent.putExtra("pin", enteredPin);
                MainActivity.this.startActivity(intent);
                return;
            }
            MainActivity.this.pinEditText.setText("");
            Toast.makeText(MainActivity.this, "Incorrect Pin, try again", 1).show();
        }
    });
}
```

从源代码我们可以知道pinFormDB是数据库中存储的密码、hashOfEnteredPin属于我们输入的密码加密后的密文，在这里我们可以修改if的判断条件

接下来通过apktool工具对apk文件进行反编译得到smail代码

```
const/4 v5, 0x0

.line 30
.local v5, "pinFromDB":Ljava/lang/String;
const/4 v3, 0x0

.line 32
.local v3, "hashOfEnteredPin":Ljava/lang/String;
:try_start_0
new-instance v0, Lpinlock/ctf/pinlock/com/pinstore/DatabaseUti
```

分别找到pinFormDB对应的是v5，hashOfEnteredPin对应的是v3

然后接下来找到两个参数在一起被调用的地方

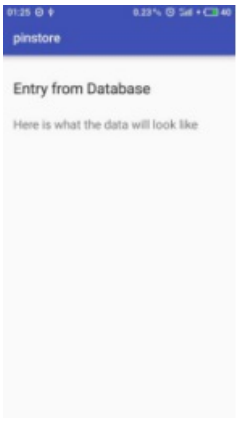
```
.line 33
.local v0, "dbUtil":Lpinlock/ctf/pinlock/com/pinstore/DatabaseUtilities;
invoke-virtual {v0}, Lpinlock/ctf/pinlock/com/pinstore/DatabaseUtilities;->fetchPin()Ljava/lang/String;
:try_end_0
.catch Ljava/io/IOException; {:try_start_0 .. :try_end_0} :catch_0

move-result-object v5

.line 38
.end local v0    # "dbUtil":Lpinlock/ctf/pinlock/com/pinstore/DatabaseUtilities;
:goto_0
:try_start_1
invoke-static {v2}, Lpinlock/ctf/pinlock/com/pinstore/CryptoUtilities;->getHash(Ljava/lang/String;)Ljava/lang/String;
:try_end_1
.catch Ljava/security/NoSuchAlgorithmException; {:try_start_1 .. :try_end_1} :catch_1
.catch Ljava/io/UnsupportedEncodingException; {:try_start_1 .. :try_end_1} :catch_2

move-result-object v3

.line 45
:goto_1
invoke-virtual {v5, v3}, Ljava/lang/String;->equalsIgnoreCase(Ljava/lang/String;)Z

move-result v6

if-eqz v6, :cond_0
```

通过修改蓝色方框内的内容：if-eqz => if-nez

注：if-eqz:如果vAA为0则跳转

if-nez:如果VAA不为0则跳转

于是我们在只要输不出正确的密码就可以直接读取数据，结果如下图



可惜这不是我们想要的flag，所以恢复原来的smail，继续分析源代码，

```
.line 52
iget-object v6, p0, Lpinlock/ctf/pinlock/com/pinstore/MainActivity$1;->this$0:Lpinlock/ctf/pinlock/com/pinstore/MainActivity;

const-string v7, "Incorrect Pin, try again"

const/4 v8, 0x1

invoke-static {v6, v7, v8}, Landroid/widget/Toast;->makeText(Landroid/content/Context;Ljava/lang/CharSequence;I)Landroid/widget/Toast;

move-result-object v6

.line 53
invoke-virtual {v6}, Landroid/widget/Toast;->show()V

goto :goto_2
d method
```

在这里我们找到一个Toast，Toast是用来显示消息的，我们可以修改smail源码让密码爆出来(v5是从数据库中读取的密文pinFromDB，v7是密码输入错误提示信息"Incorrect Pin, try again")

通过修改蓝色方框内的内容：v7 => v5

结果如下图，

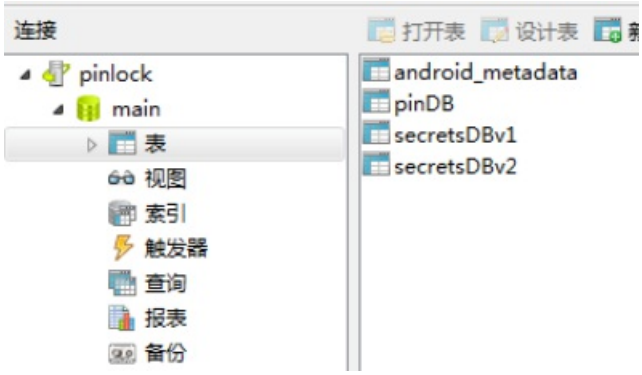
得到加密的密码密文：d8531a519b3d4dfebece0259f90b466a23efc57b

MD5解密得：7498

可惜也不是我们想要的flag。

在这里还是话费了我大量时间来找flag位置，

这里是后知后觉的pinlocak.db，我们用navicat for sqlite打开，这里我们可以发现他有pinDB、secretsDBv1和secretsDBv2



然而在源代码中我们查看，发现我们只读取过secretsDBv1和pinDB

//[MainActivity.java]

pinFromDB = new DatabaseUtilities(MainActivity.this.getApplicationContext()).fetchPin();//**********

Intent intent = new Intent(MainActivity.this, SecretDisplay.class);

intent.putExtra("pin", enteredPin);

MainActivity.this.startActivity(intent);

//[MainActivity.java]

tv.setText(New CryptoUtilities("v1", pin).decrypt(new DatabaseUtilities(getApplicationContext()).fetchSecret()));

```
public String fetchSecret() throws IOException {
    openDB();
    Cursor cursor = this.db.rawQuery("SELECT entry FROM secretsDBv1", null);
    String secret = "";
    if (cursor.moveToFirst()) {
        secret = cursor.getString(0);
    }
    Log.d("secret", secret);
    cursor.close();
    return secret;
}

public String fetchPin() throws IOException {
    openDB();
    Cursor cursor = this.db.rawQuery("SELECT pin FROM pinDB", null);
    String pin = "";
    if (cursor.moveToFirst()) {
        pin = cursor.getString(0);
    }
    cursor.close();
    return pin;
}
```

所以我们更可以读取secretsDBv2数据看看，但是要注意一个问题

```
String pin = getIntent().getStringExtra("pin");
try {
    tv.setText(new CryptoUtilities("v1", pin).decrypt(new DatabaseUtilities(getApplicationContext()).fetchSecret()));
} catch (Exception e) {
    Log.e("Pinlock", "exception", e);
}
Toast toast = Toast.makeText(context, pin, 1);
```

```
public SecretKeySpec getKey(String version) throws Exception {
    if (version.equalsIgnoreCase("v1")) {
        Log.d("Version", version);
        return new SecretKeySpec(Arrays.copyOf(MessageDigest.getInstance("SHA-1").digest("tOps3kr3ht3y".getBytes("UTF-8")), 16), "AES");
    }
    Log.d("Version", version);
    byte[] salt = "SampleSalt".getBytes();
    return new SecretKeySpec(SecretKeyFactory.getInstance("PBKDF2WithHmacSHA1").generateSecret(new PBEKeySpec(this.pin.toCharArray(), salt, 1000, 128)).getEncoded(), "AES");
}

public String encrypt(String plaintext) throws Exception {
```

我们可以发现从数据库中读取数据的解密方式不一样

所以我们需要修改两处位置：

位置一：在DatabaseUtilities.smail里找到字符串”SELECT entry FROM secretsDBv1”

```
    .end annotation

    .prologue
    .line 30
    invoke-virtual {p0}, Lpinlock/ctf/pinlock/com/pinstore/DatabaseUtilities;->openDB()V

    .line 31
    const-string v1, "SELECT entry FROM secretsDBv1"

    .line 32
    .local v1, "query":Ljava/lang/String;
    iget-object v3, p0, Lpinlock/ctf/pinlock/com/pinstore/DatabaseUtilities;->db:Landroid/database/sqlite/SQLiteDatabase;

    const/4 v4, 0x0
```

通过修改蓝色方框内的内容：

SELECT entry FROM secretsDBv1 -> SELECT entry FROM secretsDBv2

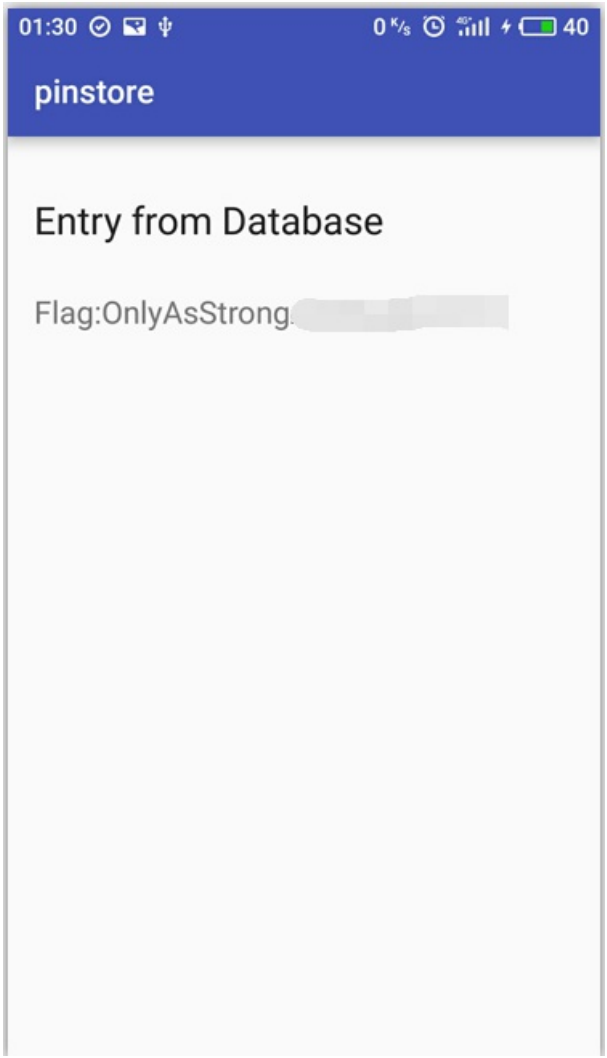
位置二：在SecretDisplay.smail里找到字符串”v1”(这里也可以同上面一样修改if条件吗；"v1"字符串好找，这里就改v1了)

```
    new-instance v1, Lpinlock/ctf/pinlock/com/pinstore/Cryp

    const-string v7, "v1"

    invoke-direct {v1, v7, v4}, Lpinlock/ctf/pinlock/com/p
```

通过修改蓝色方框内的内容：

v1 -> v2 (这里可以是任意非v1内容)

然后打包回编译，签名apk后，我们就可以得到flag了


01:30
40
pinstore
Entry from Database
Flag:OnlyAsStrong