攻防世界-app3

原创

bufsnake 于 2019-07-24 18:33:02 发布

分类专栏： android打怪升级 文章标签： 攻防世界 app3 ctf Android逆向 writeup

本文链接：https://blog.csdn.net/qq_40640243/article/details/97142658

所需工具

> jadx
> android-backup-extractor
> DB Browser for SQLite

考察考点

> 如何提取或解压缩.ab文件（Android备份文件）
> sqlite3数据库加密以及解密

艰辛的过程

> 拿到程序，发现是ab结尾的文件，就上网搜了一下，发现是android应用的备份文件，需要用android-backup-extractor里的abe.jar将数据提取出来

```
java -jar abe.jar unpack app3.ab app3.tar
```

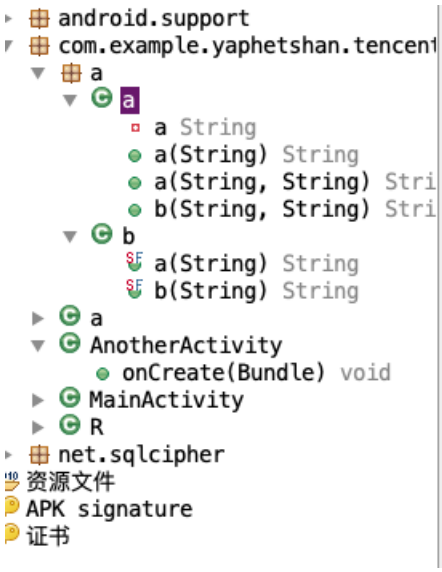
接下来解压app3.tar，一番搜寻后，发现了base.apk，也就是我们需要分析的程序

```
Desktop  cd apps/com.example.yaphetshan.tencentwelcome
com.example.yaphetshan.tencentwelcome  ls
Demo.db     Encryto.db _manifest  a          db
com.example.yaphetshan.tencentwelcome  tree
.
├── Demo.db
├── Encryto.db
├── _manifest
├── a
│   └── base.apk
└── db
```

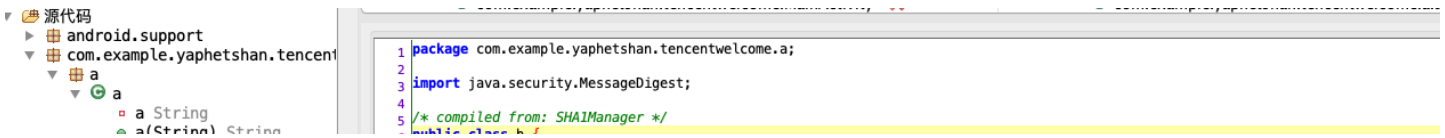
```
└── Demo.db

2 directories, 5 files
```


用jadx打开分析程序的逻辑结构，找到主要代码

```java
import net.sqlcipher.database.SQLiteDatabase;

public class MainActivity extends AppCompatActivity implements OnClickListener {
    private SQLiteDatabase a;
    private a b;
    private Button c;

    /* access modifiers changed from: protected */
    public void onCreate(Bundle bundle) {
        super.onCreate(bundle);
        setContentView((int) R.layout.activity_main);
        this.c = (Button) findViewById(R.id.add_data);
        this.c.setOnClickListener(this);
        Editor edit = getSharedPreferences("test", 0).edit();
        edit.putString("Is_Encroty", "1");
        edit.putString("Encryto", "SqlCipher");
        edit.putString("ver_sion", "3_4_0");
        edit.apply();
        a();
    }

    private void a() {
        SQLiteDatabase.loadLibs(this);
        this.b = new a(this, "Demo.db", null, 1);
        ContentValues contentValues = new ContentValues();
        contentValues.put("name", "Stranger");
        contentValues.put("password", Integer.valueOf(123456));
        a aVar = new a();
        String a2 = aVar.a(contentValues.getAsString("name"), contentValues.getAsString("password"));
        this.a = this.b.getWritableDatabase(aVar.a(a2 + aVar.b(a2, contentValues.getAsString("password"))).substring(0, 7));
        this.a.insert("TencentMicrMsg", null, contentValues);
    }

    public void onClick(View view) {
        if (view == this.c) {
            Intent intent = new Intent();
            intent.putExtra("name", "name");
            intent.putExtra("password", "pass");
            intent.setClass(this, AnotherActivity.class);
            startActivity(intent);
        }
    }
}
```


```java
package com.example.yaphetshan.tencentwelcome.a;

/* compiled from: Cipher */
public class a {
    private String a = "yaphetshan";

    public String a(String str, String str2) {
        String substring = str.substring(0, 4);
        return substring + str2.substring(0, 4);
    }

    public String b(String str, String str2) {
        new b();
        return b.a(str);
    }

    public String a(String str) {
        new b();
        return b.b(str + this.a);
    }
}
```

主要逻辑如下：

> 将Stranger和123456两个字符串取前四个，并拼接到一起，得到Stra1234将Stra1234进行md5加密后用base16加密，得到的字符串前面加上Stra1234，在拼接上字符串yaphetshan后用SHA-1加密，在用base16加密，最后取得到字符串的前七位，即是最后需要解数据库密码的密码

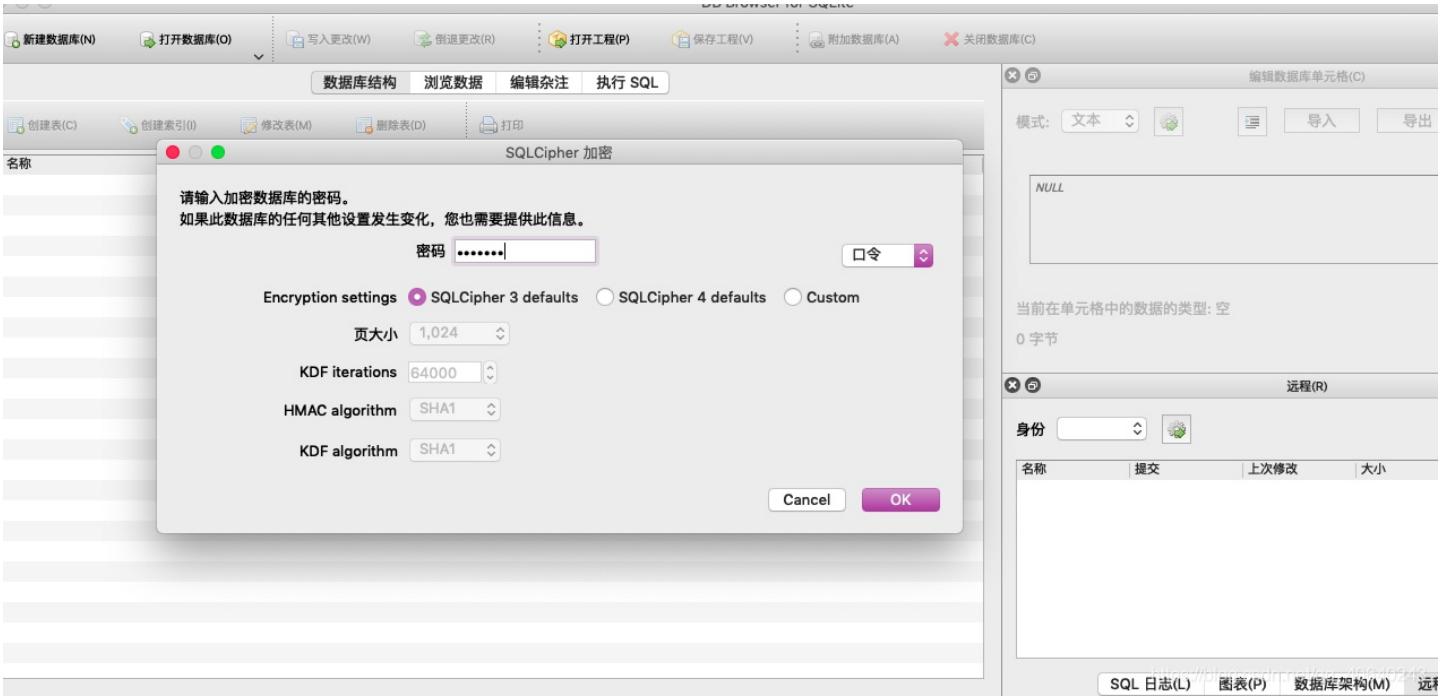
解密代码如下：

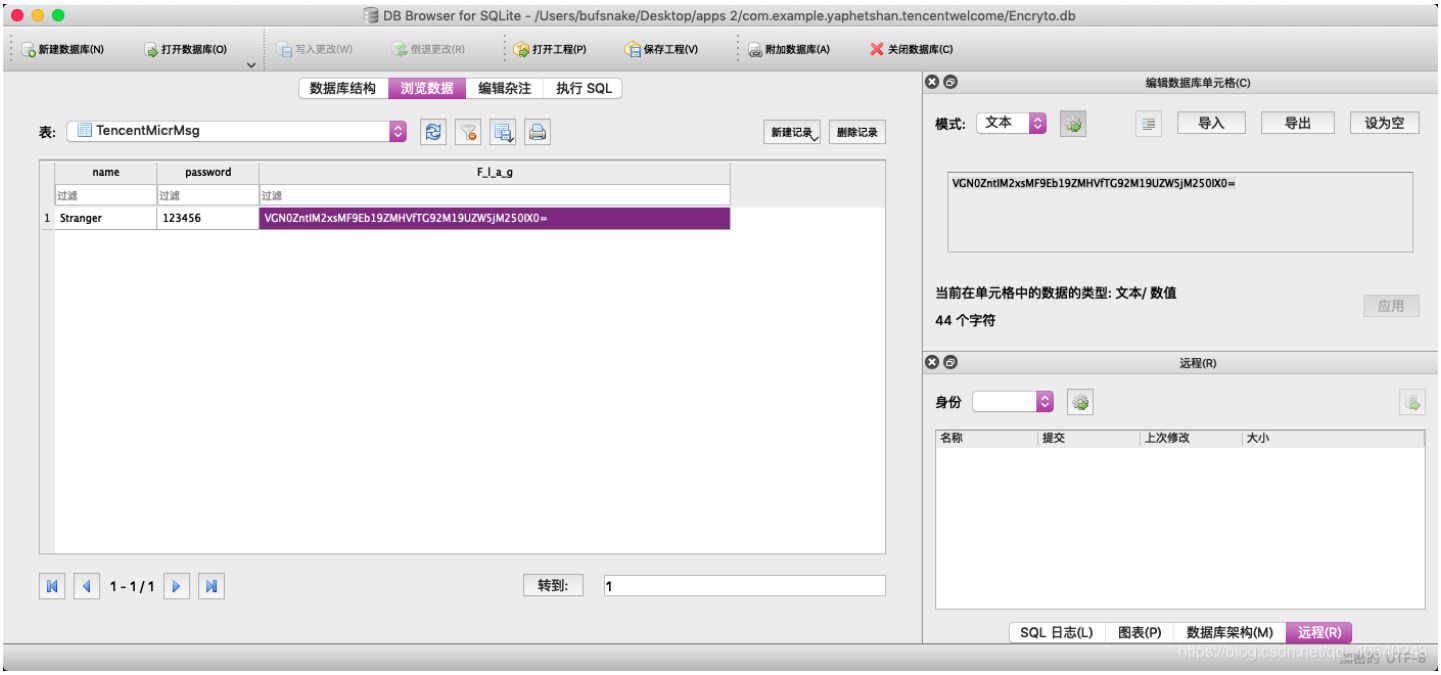
```java
import java.security.MessageDigest;

public class b {

    public static String a(String str) {
        byte[] digest;
        char[] cArr = {'0', '1', '2', '3', '4', '5', '6', '7', '8', '9', 'a', 'b', 'c', 'd', 'e', 'f'};
        try {
            byte[] bytes = str.getBytes();
            MessageDigest instance = MessageDigest.getInstance("MD5");
            instance.update(bytes);

            char[] cArr2 = new char[(16 * 2)];
            int i = 0;
            for (byte b : instance.digest()) {
                int i2 = i + 1;
                cArr2[i] = cArr[(b >>> 4) & 15];
                i = i2 + 1;
                cArr2[i2] = cArr[b & 15];
            }
            return new String(cArr2);
        } catch (Exception e) {
            return null;
        }
    }
    public static final String b(String str) {
        byte[] digest;
        char[] cArr = {'0', '1', '2', '3', '4', '5', '6', '7', '8', '9', 'a', 'b', 'c', 'd', 'e', 'f'};
        try {
            byte[] bytes = str.getBytes();
            MessageDigest instance = MessageDigest.getInstance("SHA-1");
            instance.update(bytes);
            char[] cArr2 = new char[(32 * 2)];
            int i = 0;
            for (byte b : instance.digest()) {
                int i2 = i + 1;
                cArr2[i] = cArr[(b >>> 4) & 15];
                i = i2 + 1;
                cArr2[i2] = cArr[b & 15];
            }
            return new String(cArr2);
        } catch (Exception e) {
            return null;
        }
    }
    public static void main(String[] args){
        String two = ("Stra1234"+a("Stra1234")+"yaphetshan");
        System.out.println(b(two).substring(0,7));
    }
}
```

得到密码

ae56f99

接下来使用DB Browser for SQLite打开需要解密的数据库





VGN0ZntlM2xsMF9Eb19ZMHVfTG92M19UZW5jM250IX0=

base64解密得到flag

Tctf{H3ll0_Do_Y0u_Lov3_Tenc3nt!}

总结

> 一道题做下来，发现，稳如菜狗