

Code Engn SmartApp 2

4.Z320

elttzero@gmail.com

Challenges : SmartApp 02

Author : 보안프로젝트 / [Link](#)


Korean :

키값을 찾으시오!

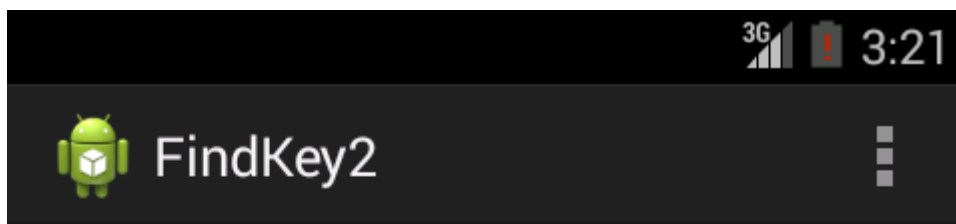
English :

Find a key

 [Download](#)

 [Like](#) [Share](#) [Sign Up to see what your friends like.](#)

키값을 찾으라는 문제입니다.



FindKey :)

다운로드 후 에뮬레이터로 돌려 보면 키를 찾으라는 메시지만 나옵니다.

이후 dex파일을 이용하여 디컴파일을 하기 위해 apk파일을 압축 해제한 뒤 dex2jar를 이용, jar파

일로 변환시켜 줍니다.

```
D:\wprv_rsrch\android\challenges\02\SmartApp L02_unpak>dex2jar.bat classes.dex
this cmd is deprecated, use the d2j-dex2jar if possible
dex2jar version: translator-0.0.9.15
dex2jar classes.dex -> classes_dex2jar.jar
Done.

D:\wprv_rsrch\android\challenges\02\SmartApp L02_unpak>dir
D 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 50E7-74AF

D:\wprv_rsrch\android\challenges\02\SmartApp L02_unpak 디렉터리

2014-02-11 오후 05:32 <DIR> .
2014-02-11 오후 05:32 <DIR> ..
2013-11-02 오후 12:42 1,648 AndroidManifest.xml
2013-11-02 오후 12:42 599,080 classes.dex
2014-02-11 오후 05:32 318,549 classes_dex2jar.jar
2014-02-11 오후 05:32 <DIR> lib
2014-02-11 오후 05:32 <DIR> res
2013-11-02 오후 12:42 2,244 resources.arsc
4개 파일 921,521 바이트
4개 디렉터리 451,616,780,288 바이트 남음

D:\wprv_rsrch\android\challenges\02\SmartApp L02_unpak>
```

이렇게 변환된 jar파일을 jd-gui를 이용하여 디컴파일을 한 뒤 안의 소스코드를 살펴봅니다.

```
public String keyString()
{
    return Security.DecryptStr("-1c776f3a2fa678cae27879e87a74553c61009ee6b037bbe2abdbd5e4314407e60000000000000000");
}

public String makeDate()
{
    Date localDate = new Date();
    return new SimpleDateFormat("yyyy-MM-dd-hh:mm:ss").format(localDate);
}

protected void onCreate(Bundle paramBundle)
{
    super.onCreate(paramBundle);
    setContentView(2130903040);
    this.aView = ((TextView)findViewById(2131230720));
    if ((makeDate() == "2013-11-02-12:35:03") && (Volume() == 53)) {
        this.aView.setText(keyString());
    }
}
```

소스 중 MainActivity 부분을 보면 암호화된 문구와 if구문을 발견할 수 있습니다.

if구문 중 하나는 현재 날짜가 "2013-11-02-12:35:03"이거나 Volume메서드(33을 리턴함)의 리턴값이 53이면 KeyString메서드를 호출하여 암호문을 복호화 시키는 것을 확인할 수 있습니다.

따라서 if문을 변경하여 True값을 리턴 하도록 설정해 주면 이번 문제의 Key값을 구할 수 있을 것이라 예상할 수 있습니다.

소스코드를 수정하기 위하여 smali파일을 추출합니다.

```
D:\prv_rsrch\android\challenges\02>apktool d -f "SmartApp L02.apk"
I: Baksmaling...
I: Loading resource table...
I: Loaded.
I: Decoding AndroidManifest.xml with resources...
I: Loading resource table from file: C:\Users\wtgtzero\apktool\framework\1.apk
I: Loaded.
I: Regular manifest package...
I: Decoding file-resources...
I: Decoding values */* XMLs...
I: Done.
I: Copying assets and libs...
```

smali 추출을 하기 위하여 apktool을 이용, apk 파일을 디코딩 해줍니다.

이후 ./smali/com/namdaehyeon/findkey2 폴더에 있는 MainActivity.smali파일을 열어줍니다.

```
.line 29
invoke-virtual {p0}, Lcom/namdaehyeon/findkey2/MainActivity;->makeDate()Ljava/lang/String;
move-result-object v0
const-string v1, "2013-11-02-12:35:03"
if-ne v0, v1, :cond_0

.line 30
invoke-virtual {p0}, Lcom/namdaehyeon/findkey2/MainActivity;->Volume()I
move-result v0
const/16 v1, 0x35
if-ne v0, v1, :cond_0

.line 31
iget-object v0, p0, Lcom/namdaehyeon/findkey2/MainActivity;->aView:Landroid/widget/TextView;
invoke-virtual {p0}, Lcom/namdaehyeon/findkey2/MainActivity;->keyString()Ljava/lang/String;
move-result-object v1
invoke-virtual {v0, v1}, Landroid/widget/TextView;->setText(Ljava/lang/CharSequence;)V

.line 34
:cond_0
return-void
.end method
```

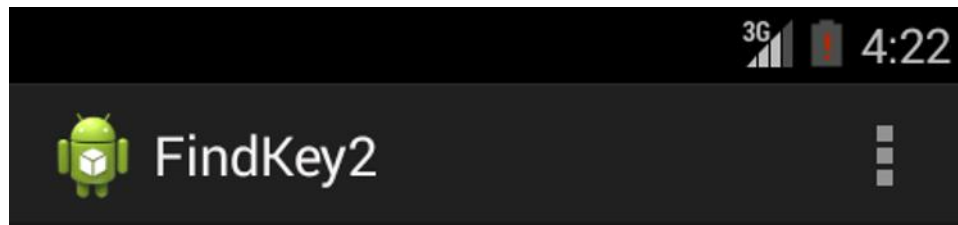
smali파일은 일종의 android의 어셈블리 문법입니다. 이중 if문은 if-ne로 표현되는 구문입니다.

if-[조건] v0, [v1, ...] : cond_0 로 구성이 되어 있으며 조건은 eq(equal), ne(not equal), ge(greater), le(less)가 있으며 인자가 하나일 때는 eqz등으로 비교합니다. 인자값은 v0, v1등으로 표시하며 if문 위에서 각 인자에 해당하는 값이 정의되고 이런 if문의 조건에 일치할 경우 cond_0으로 점프하는 방식입니다.

위 구문에서 처음 if문은 makeDate()메서드의 리턴값과 "2013-11-02-12:35:03"의 값이 다를 경우, 다음 if문은 Volume()메서드의 리턴값과 0x35(53)이 일치하지 않을 경우 cond_0로 점프하는 구문입니다. cond_0는 return-void이기에 cond_0로 점프하지 못하게 if-ne를 if-eq로 수정해 주면 if문을 우회할 수 있게 됩니다.

```
D:\wprv_rsrch\android\challenges\W02>apktool b "SmartApp L02" T2.apk
D:\wprv_rsrch\android\challenges\W02>jarsigner -verbose -keystore my-release-key.keystore T2.apk alias_name_
D:\wprv_rsrch\android\challenges\W02>adb -s emulator-5554 install T2.apk_
```

이렇게 수정된 smali코드를 apktool을 이용하여 다시 빌드한 뒤 sign하고 설치를 해준 뒤에 어플을 실행시켜 주면 키값을 알아낼 수 있습니다.



The Key is [REDACTED]