22/12/2021

Oyun Manipülasyonu

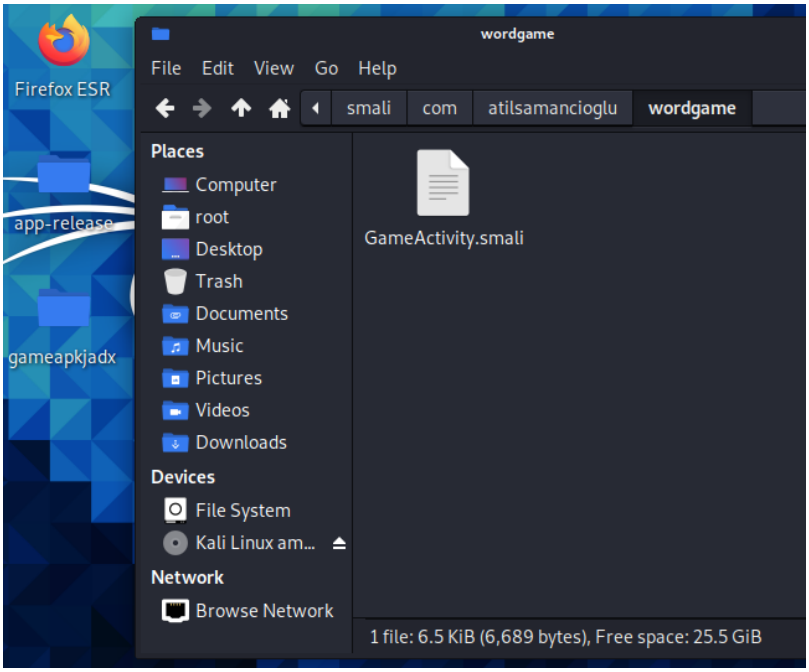
Hazırlayan

Ümmü Derya Çelik

Pratik ve mantığını anlamak için geliştirilmiş basit bir kelime oyununu manipüle etmeyi deneyeceğiz.

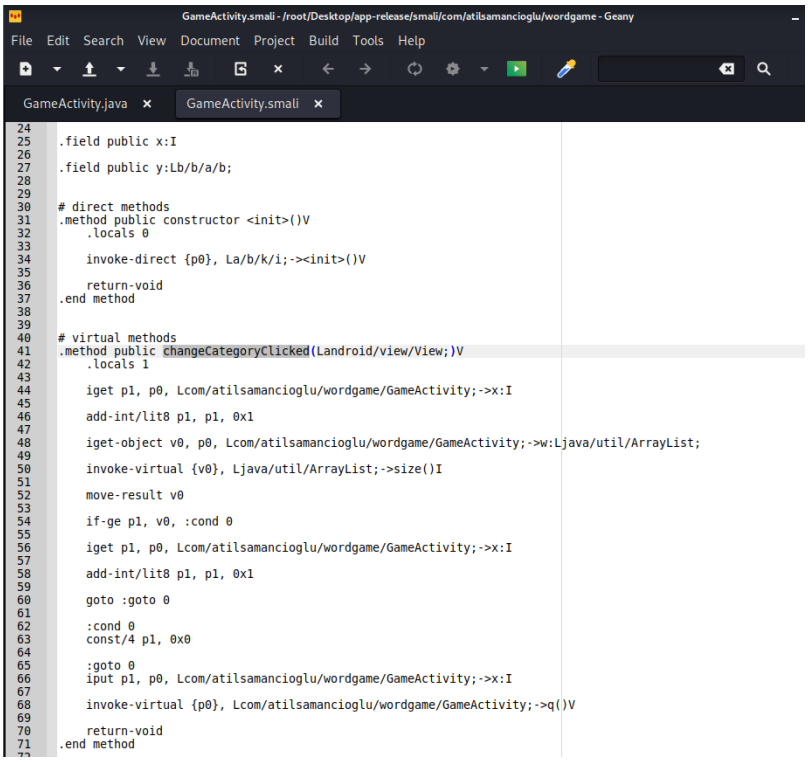
Oyunun genel görünümü görselde görüldüğü gibi olup 3 kategoriye ve basit bir veri tabanına sahip sahiptir. Her veri tabanında bulunan kategorisine uygun kelime girildiğinde Score birer birer arttırılmaktadır.

Manipülasyon için ilk olarak dalvik kodlarına erişip bir şeyler bulabilecek miyiz bir bakalım.



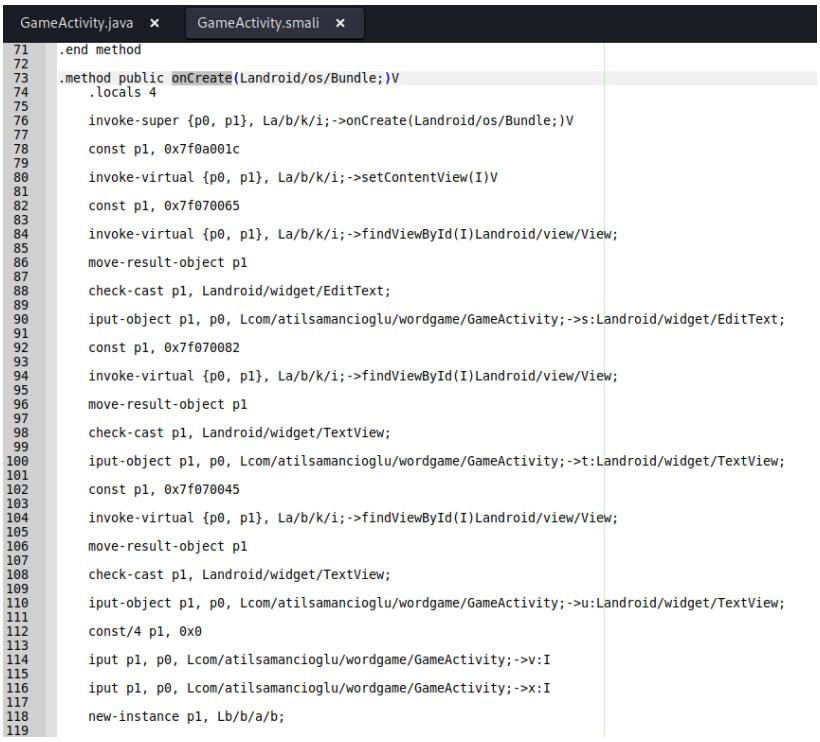
Apktool ile apk uygulama dosyasını decompile etmeyi görmüştük.

GameActivity smali dosyasını açıp bir göz atalım.



Proguard ile korunuyor olsa gerek ki yine anlamsız ve karışık ifadelerle hem dosya adları hem kodlar şifrelenmiş.

Geany ile gameactivity kodlarını açtık.



onCreate ve changeCategoryClicked gibi fonksiyonları görebiliyoruz.

Dalvik kodları anlayamasak da dalvik opcodes sitesinden kodların ne anlama geldiğine bakarak çıkarımlarda bulunabiliriz.

Güvenli değil | pallergabor.uw.hu/androidblog/dalvik_...

Uygulamalar | https://ue.firat.edu.tr/ | Merkezi Kimlik Doğ... | WhatsApp | Netflix | Okuma listes

Dalvik opcodes

Author: Gabor Paller

Vx values in the table denote a Dalvik register. Depending on the instruction, 16, 256 or 64k registers can be accessed. Operations on long and double values use two registers, e.g. a double value addressed in the V0 register occupies the V0 and V1 registers.

Boolean values are stored as 1 for true and 0 for false. Operations on booleans are translated into integer operations.

All the examples are in hig-endian format, e.g. 0F00 0A00 is coded as 0F, 00, 0A, 00 sequence.

Note there are no explanation/example at some instructions. This means that I have not seen that instruction "in the wild" and its presence/name is only known from Android opcode constant list.

Consider using the dedexer tool to observe the Dalvik opcodes in real-life dex files!

Opcode (hex)	Opcode name	Explanation	Example
00	nop	No operation	0000 - nop
01	move vx,vy	Moves the content of vy into vx. Both registers must be in the first 256 register range.	0110 - move v0, v1 Moves v1 into v0.
02	move/from16 vx,vy	Moves the content of vy into vx. vy may be in the 64k register range while vx is one of the first 256 registers.	0200 1900 - move/from16 v0, v25 Moves v25 into v0.
03	move/16		
04	move-wide		
05	move-wide/from16 vx,vy	Moves a long/double value from vy to vx. vy may be in the 64k register range while wx is one of the first 256 registers.	0516 0000 - move-wide/from16 v22, v0 Moves v0 into v22.
06	move-wide/16		
07	move-object vx,vy	Moves the object reference from vy to vx.	0781 - move-object v1, v8 Moves the object reference in v8 to v1.
08	move-object/from16 vx,vy	Moves the object reference from vy to vx, vy can address 64k registers and vx can address 256 registers.	0801 1500 - move-object/from16 v1, v21 Move the object reference in v21 to v1.
09	move-object/16		
0A	move-result vx	Move the result value of the previous method invocation into vx.	0A00 - move-result v0 Move the return value of a previous method invocation into v0.
0B	move-result-wide vx	Move the long/double result value of the previous method invocation into vx,vx+1.	0B02 - move-result-wide v2 Move the long/double result value of the previous method invocation into v2,v3.
0C	move-result-object vx	Move the result object reference of	0C00 - move-result-object v0

Sayfa görünümü şekilde ki gibidir.

Örneğin move result ifadesi ne demekmiş ona bakalım.

0A	move-result vx	Move the result value of the previous method invocation into vx.	0A00 - move-result v0 Move the return value of a previous method invocation into v0.

Bir önceki metodun sonucunu vx yerinde ne yazıyorsa oraya gönder anlamı taşımaktadır.

```
invoke-super {p0, p1}, La/b/k/i;->onCreate(Landroid/os/Bundle;)V
const p1, 0x7f0a001c
invoke-virtual {p0, p1}, La/b/k/i;->setContentView(I)V
const p1, 0x7f070065
invoke-virtual {p0, p1}, La/b/k/i;->findViewById(I)Landroid/view/View;
move-result-object p1
check-cast p1, Landroid/widget/EditText;
iput-object p1, p0, Lcom/atilsamancioglu/wordgame/GameActivity;->s:Landroid/widget/EditText;
const p1, 0x7f070082
invoke-virtual {p0, p1}, La/b/k/i;->findViewById(I)Landroid/view/View;
move-result-object p1
check-cast p1, Landroid/widget/TextView;
iput-object p1, p0, Lcom/atilsamancioglu/wordgame/GameActivity;->t:Landroid/widget/TextView;
const p1, 0x7f070045
invoke-virtual {p0, p1}, La/b/k/i;->findViewById(I)Landroid/view/View;
move-result-object p1
```

Move result p1 olduğuna göre takip edeceğimiz yer p1.

Bu çıktıların p1 adlı değişken de toplandığını görebiliriz.

Line 2 ye bakalım.

```
.line 2
iget v0, p0, Lcom/atilsamancioglu/wordgame/GameActivity;->v:I
add-int/2addr v0, p1
iput v0, p0, Lcom/atilsamancioglu/wordgame/GameActivity;->v:I
iget-object p1, p0, Lcom/atilsamancioglu/wordgame/GameActivity;->t:Landroid/widget/TextView;
const-string v0, "Score: "
```

Add int diye birşey görüyoruz. Dalvik opcodes sitesinden bakarsak;

90	add-int vx,vy,vz	Calculates vy+vz and puts the result into vx.	9000 0203 - add-int v0, v2, v3 Adds v3 to v2 and puts the result into v0[4].

2 değeri topladığı anlamına gelir yani p1 değeri ile v0 değerini topladığını anlayabiliriz.

Bir alt satıra da bakarsak bu işlemin bize score u veren işlem olduğunu görebiliriz. p1 değeri yerine biz ne değer verirsek verelim o değeri döndürecektir.

```
.line 2
iget v0, p0, Lcom/atilsamancioglu/wordgame/GameActivity;->v:I
add-int/2addr v0, p1
iput v0, p0, Lcom/atilsamancioglu/wordgame/GameActivity;->v:I
iget-object p1, p0, Lcom/atilsamancioglu/wordgame/GameActivity;->t:Landroid/widget/TextView;
const-string v0, "Score: "
```

Eğer bir değer oluşturup o değere 1 verirsek score hep 1 olacaktır. Dalvik opcodes sayfasından dalvik te nasıl integer tanımlanır bulalım.

14	const vx, lit32	Puts the integer constant into vx	1400 4E61 BC00 - const v0, #12345678 // #00BC614E Moves literal 12345678 into v0.

lit32 yerine ne yazarsak vx değerine tanımlanacak anlamına gelmektedir.

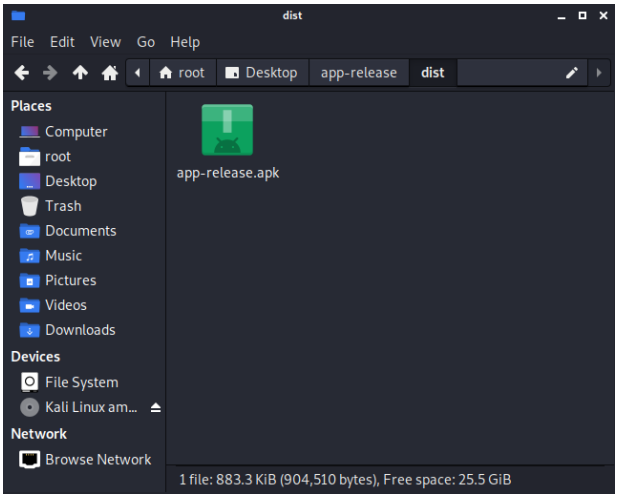
Yani Move result yerine şekilde ki gibi p1 e değer verirsek artık sabit bir değeri,1, var demektir.

```
invoke-virtual {p1}, Ljava/lang/Object;->toString()Ljava/lang/String;
const p1, 1
iget-object v0, p0, Lcom/atilsamancioglu/wordgame/GameActivity;->y:Lb/b/a/b;
iget-object v0, v0, Lb/b/a/b;->b:Ljava/util/ArrayList;
```

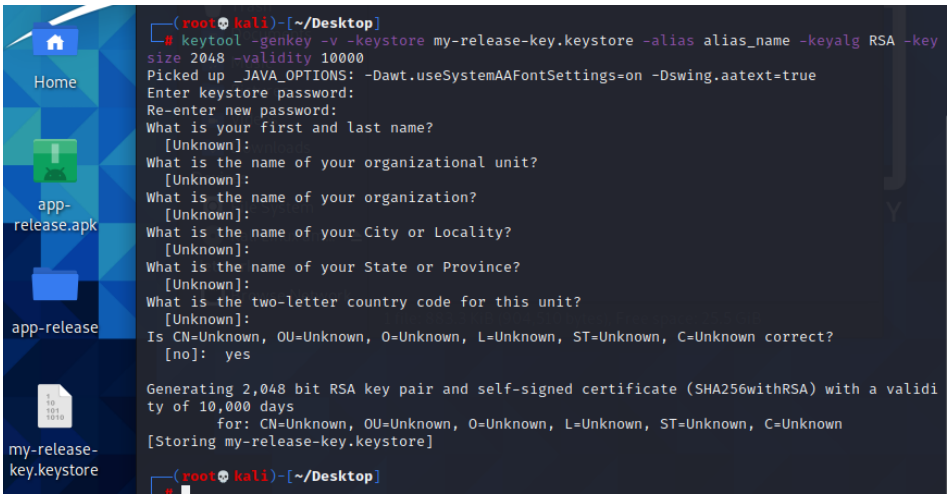
Şu an da manipüle ettik diyebiliriz. Düşünürsek score hesaplayan bir fonksiyon olmalı eğer p1 i direk skorun üstüne eklerse direk score artı 1 olacak bu da ne yazarsak yazalım score değerinin artması gerekir.

```
┌──(root💀kali)-[~/Desktop]
└─# apktool b app-release
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
I: Using Apktool 2.5.0-dirty
I: Checking whether sources has changed ...
I: Smaling smali folder into classes.dex ...
I: Checking whether resources has changed ...
I: Building resources ...
W: aapt: brut.common.BrutException: brut.common.BrutException: Could not extr
act resource: /prebuilt/linux/aapt_64 (defaulting to $PATH binary)
W: res/drawable-v24/$ic_launcher_foreground__0.xml: Invalid file name: must c
ontain only [a-z0-9_.]
brut.androlib.AndrolibException: brut.common.BrutException: could not exec (e
xit code = 1): [aapt, p, --min-sdk-version, 23, --target-sdk-version, 29, --v
ersion-code, 1, --version-name, 1.0, --no-version-vectors, -F, /tmp/APKTOOL14
500583348545824260.tmp, -0, resources.arsc, -0, png, -0, arsc, -I, /root/.loc
al/share/apktool/framework/1.apk, -S, /root/Desktop/app-release/res, -M, /roo
t/Desktop/app-release/AndroidManifest.xml]
```

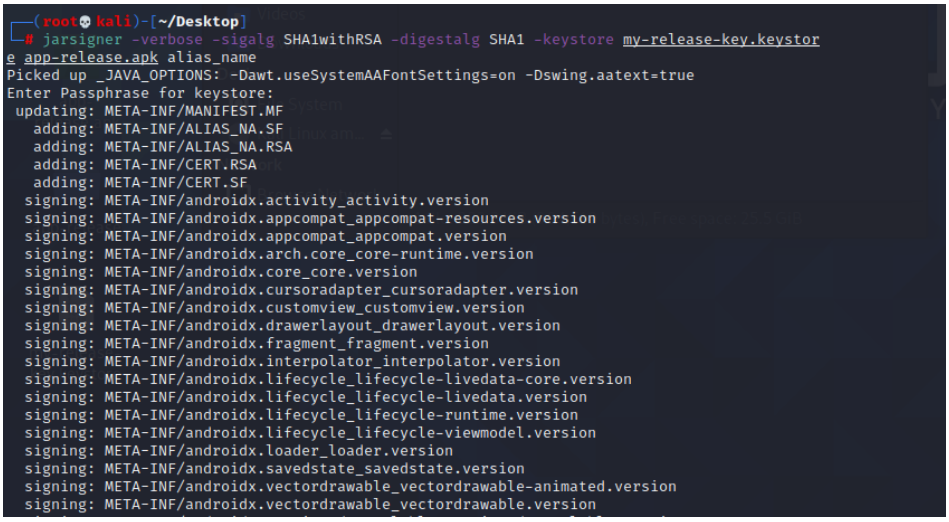
Değiştirdiğimiz kodu da kaydedip decompile ettiğimiz uygulama dosyasını tekrar build edip apk dosyası oluşturuyoruz.



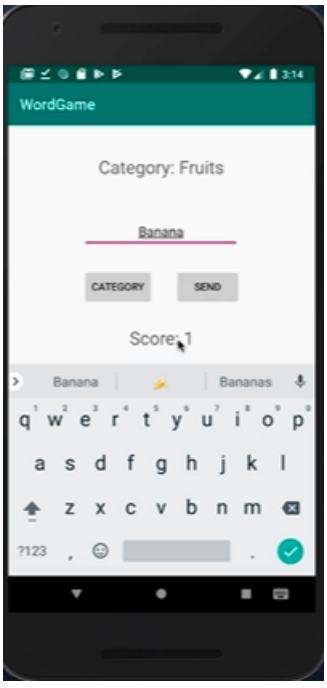
Manipüle ettiğimiz apk dosyasını görmekteyiz. Şimdi anahtar oluşturup sorunsuz yüklenip çalıştırılabilmesi için uygulamamızı imzalayalım.



Bu komutla key oluşturuyoruz. Sol alt köşede keystore görünmektedir.



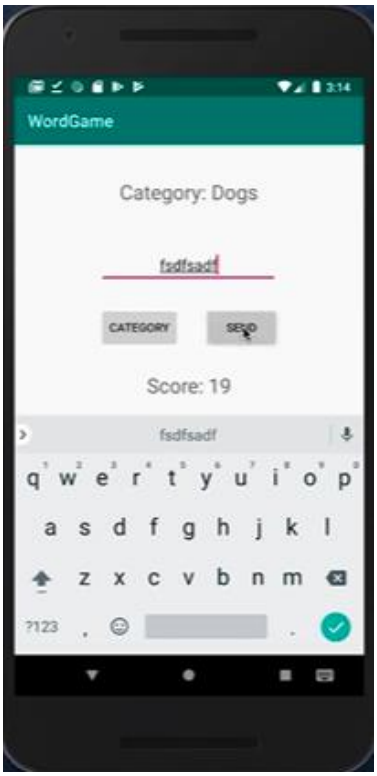
Jarsigner apk imzalayıcısını kullanarak görseldeki komut ile apk dosyamızı da imzaladık.



Şimdi oyunu açıp manipüle edebilmiş miyiz bir bakalım.

Banana yazıp normal çalışma rutinin kontrol ettik ve çalışıyor.

Rastgele bir şeyler deneyelim.



Görüldüğü gibi artık hangi kategori olduğu fark etmeksizin ne yazarsak yazalım score artacaktır.

Oyun manipüle edildi.