Modding de aplicaciones de Android P1

Modificando las Shared Preferences de las aplicaciones (sin acceso Root)

Clonar este repositorio de GitHub: https://github.com/m00nbyt3/42_apkmod_1

SELF-MODDING

1. Instalar apktool con "brew install apktool"
2. Descargar el archivo APK desde un portal como APKMirror. También podremos extraer el archivo APK desde nuestro dispositivo (APK extractor o Apktool M). Recomendable que la APK no esté spliteada (APK bundle).
3. Descompilar la APK con "apktool d your_app.apk"
4. Modificar el archivo Manifest.xml y agregar ' android:debuggable="true" ' al final de la etiqueta <application …>

```
<manifest>
    ...
    <application ... android:debuggable="true">
        ...
    </application>
    ...
</manifest>
```

5. Recompilar la APK con "apktool b your_app -o your_app_debug.apk"
6. Descargar uber-apk-signer-1.2.1.jar
7. Firmar la aplicación con "java -jar uber-apk-signer-1.2.1.jar -a your_app_debug.apk"
8. Ejecutar ./adb desde la carpeta *plataform-tools* (.zip en github o aquí, o instalar con brew: "brew install android-platform-tools", o usar WebADB)
9. Instalar la APK en el móvil, con adb: "adb install your_app_debug-aligned-debugSigned.apk"
10. Iniciar una shell con "adb shell"
11. Listar los paquetes instalados con "pm list packages" y utilizar grep para filtrar por el nombre de tu aplicación
12. Ejecutar el comando "run-as" seguido de el nombre del paquete de la aplicación
13. Acceder al directorio *shared_prefs* y ver el contenido de los archivos con las preferencias a modificar.
14. Para modificar las preferencias, es probable que no esté instalado ningún editor de texto, por lo que se recomienda usar sed (sustituye el texto de entrada con el de salida):

```
sed -i 's/"pref_name" value="2"/"pref_name" value="999"/' pref_file.xml
```

15. Las Shared Prefs de la aplicación ya estarán modificadas.

OPEN MODDING

1. Para poder realizar modificaciones en la app, se necesita conocer primero las preferencias a modificar, por lo que se recomienda primero realizar *SELF MODDING*
2. Buscar el nombre de la llamada a la primera actividad en Manifest.xml

```
<application android:banner="@drawable/app_banner" android:icon="@mipmap/
app_icon" android:isGame="true" android:label="@string/app_name" android:
theme="@style/UnityThemeSelector">
    <activity android:configChanges="density|fontScale|keyboard|keyboardHid
    den|layoutDirection|locale|mcc|mnc|navigation|orientation|screenLayout|
    screenSize|smallestScreenSize|touchscreen|uiMode" android:
    hardwareAccelerated="false" android:label="@string/app_name" android:
    launchMode="singleTask" android:name="
    com.unity3d.player.UnityPlayerActivity" android:screenOrientation="
    sensorLandscape">
        <intent-filter>
            <action android:name="android.intent.action.MAIN"/>
            <category android:name="android.intent.category.LAUNCHER"/>
            <category android:name="
            android.intent.category.LEANBACK_LAUNCHER"/>
        </intent-filter>
        <meta-data android:name="unityplayer.UnityActivity" android:value="
        true"/>
    </activity>
```

activity

3. En la carpeta *smali*, acceder a la ruta indicada por el nombre de la actividad: *smali/com/unity3d/player/*
4. Agregar el archivo *CheatClass.smali* en este directorio (este archivo requiere de modificación, hay comentarios indicando las partes a cambiar, busca #)
5. Modificar el archivo UnityPlayerActivity, agregando lo siguiente después de *.method protected onCreate*:

```
.method protected onCreate(Landroid/os/Bundle;)V

    ...

    ##From here (after .locals)

    move-object v0, p0

    check-cast v0, Landroid/app/Activity;

    invoke-static {v0}, Lcom/unity3d/player/CheatClass;->addCoins(Landroid/app/Activity;)V

    ##To here
    ...
.end method
```

.method protected onCreate

6. Recompilar la APK con "apktool b your_app -o your_app_mod.apk"
7. Firmar la aplicación con "java -jar uber-apk-signer-1.2.1.jar -a your_app_mod.apk"
8. Instalar la APK en el móvil (your_app_mod-aligned-debugSigned.apk)
9. La aplicación ya estará modificada

EXTRA: Añadir toast message personalizado al iniciar la aplicación

1. Acceder al archivo *smali/com/unity3d/player/*UnityPlayerActivity
2. Agregar lo siguiente en *.method protected onCreate:*

```
.method protected onCreate(Landroid/os/Bundle;)V
    ...

    ##From here (after .locals)

    const/4 v0, 0x1

    const-string v1, "YOUR MESSAGE HERE"

    invoke-static {p0, v1, v0}, Landroid/widget/Toast;->makeText(Landroid/content/Context;Ljava/lang/CharSequence;I)Landroid/widget/Toast;

    move-result-object v0

    invoke-virtual {v0}, Landroid/widget/Toast;->show()V

    ##To here
    ...
.end method
```