

# REVERSE ENGINEERING AND MALWARE ANALYSIS

## **Team Members :**

LANKE NARASIMHASWAMY (120134204028).

MARADANA VENKAT NAIDU (120134204029).

MUKTHIPUDI MANVITHA (120134204030).

NEELAPU TARUSH REDDY (120134204031).

**Project Register Number : SBAP0001869**

## **REVERSING MOBILE APPLICATION**

We took a simple mobile application which will print whether the given user is VIP user or not. In this application there is only one VIP user whose username is "sabin". We going to reverse engineer it and make it available for everyone.

"Dex-2-jar" is a opensource tool to covert Dex file into jar file.

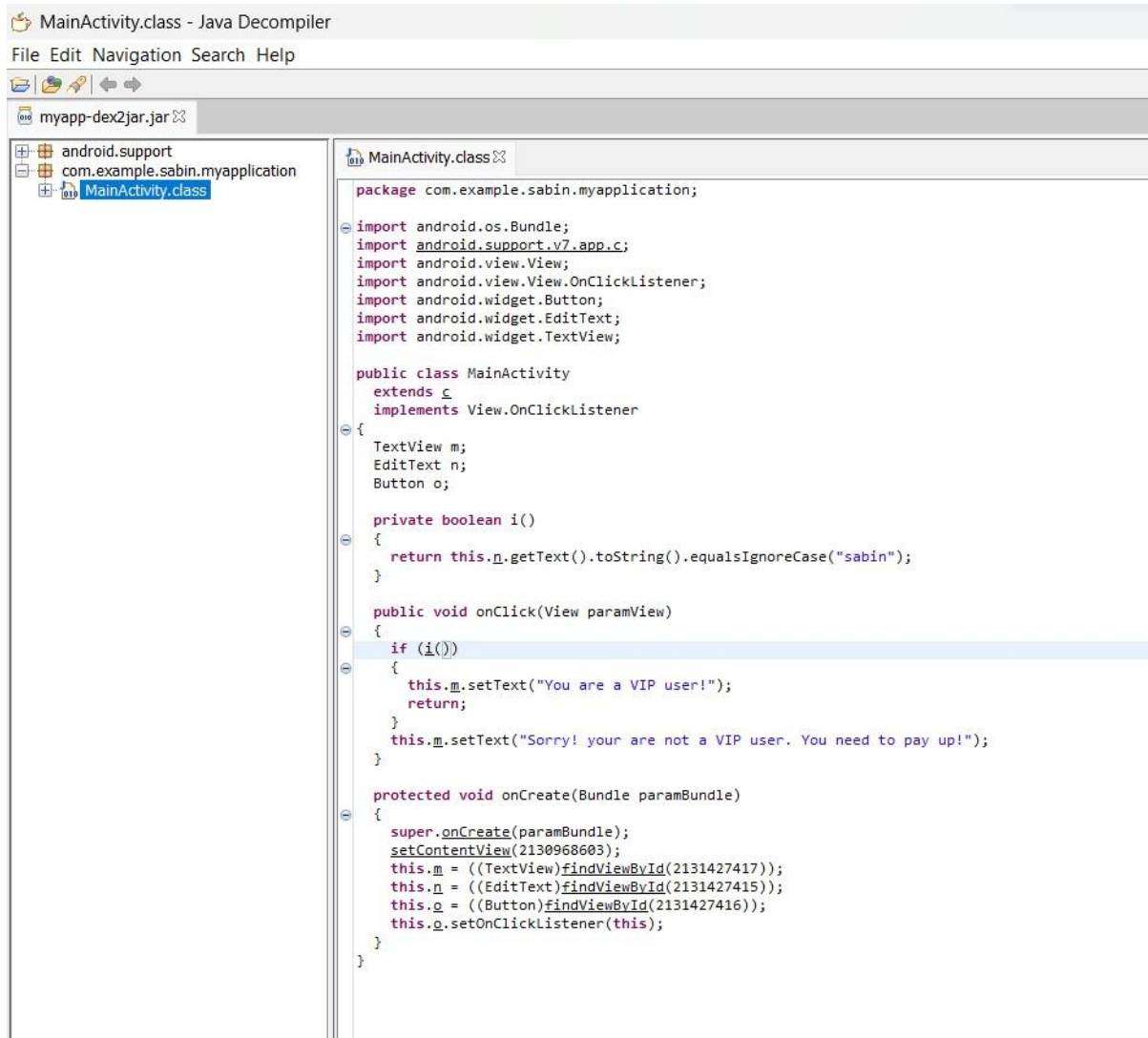
Name	Date modified	Type	Size
lib	27-10-2014 05:32 PM	File folder	
d2j_invoke.bat	27-10-2014 05:32 PM	Windows Batch File	1 KB
d2j_invoke.sh	27-10-2014 05:32 PM	Shell Script	2 KB
d2j-baksmali.bat	27-10-2014 05:32 PM	Windows Batch File	1 KB
d2j-baksmali.sh	27-10-2014 05:32 PM	Shell Script	2 KB
d2j-dex2jar.bat	27-10-2014 05:32 PM	Windows Batch File	1 KB
d2j-dex2jar.sh	27-10-2014 05:32 PM	Shell Script	2 KB
d2j-dex2smali.bat	27-10-2014 05:32 PM	Windows Batch File	1 KB
d2j-dex2smali.sh	27-10-2014 05:32 PM	Shell Script	2 KB
d2j-dex-recompute-checksum.bat	27-10-2014 05:32 PM	Windows Batch File	1 KB
d2j-dex-recompute-checksum.sh	27-10-2014 05:32 PM	Shell Script	2 KB
d2j-jar2dex.bat	27-10-2014 05:32 PM	Windows Batch File	1 KB
d2j-jar2dex.sh	27-10-2014 05:32 PM	Shell Script	2 KB
d2j-jar2jasmin.bat	27-10-2014 05:32 PM	Windows Batch File	1 KB
d2j-jar2jasmin.sh	27-10-2014 05:32 PM	Shell Script	2 KB
d2j-jasmin2jar.bat	27-10-2014 05:32 PM	Windows Batch File	1 KB
d2j-jasmin2jar.sh	27-10-2014 05:32 PM	Shell Script	2 KB
d2j-smali.bat	27-10-2014 05:32 PM	Windows Batch File	1 KB
d2j-smali.sh	27-10-2014 05:32 PM	Shell Script	2 KB
d2j-std-apk.bat	27-10-2014 05:32 PM	Windows Batch File	1 KB

```
PS H:\reverse engineering\experiment\sabin bir\dex2jar-2.0> d2j-dex2jar.bat -f ..\myapp.apk
d2j-dex2jar.bat : The term 'd2j-dex2jar.bat' is not recognized as the name of a cmdlet, function, script file, or
operable program. Check the spelling of the name, or if a path was included, verify that the path is correct and try
again.
At line:1 char:1
+ d2j-dex2jar.bat -f ..\myapp.apk
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (d2j-dex2jar.bat:String) [], CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException

Suggestion [3,General]: The command d2j-dex2jar.bat was not found, but does exist in the current location. Windows PowerShell does not load commands from th
e current location by default. If you trust this command, instead type: ".\d2j-dex2jar.bat". See "get-help about_Command_Precedence" for more details.
PS H:\reverse engineering\experiment\sabin bir\dex2jar-2.0> ./d2j-dex2jar.bat -f ..\myapp.apk
dex2jar ..\myapp.apk -> ..\myapp-dex2jar.jar
```

dex2jar-2.0	30-06-2023 09:09 AM	File folder	
jd-gui-windows-1.4.0	30-06-2023 09:31 AM	File folder	
myapp.apk	29-06-2023 06:26 PM	BlueStacks.Apk	785 KB
myapp-dex2jar.jar	30-06-2023 09:09 AM	Executable Jar File	684 KB

Viewing the jar file in JD decompiler which only readable not writeable

The image shows a Java decompiler window titled 'MainActivity.class - Java Decompiler'. The window has a menu bar with 'File', 'Edit', 'Navigation', 'Search', and 'Help'. Below the menu is a toolbar with icons for file operations and navigation. The left pane shows a project tree with 'myapp-dex2jar.jar' expanded, revealing the package 'com.example.sabin.myapplication' and the file 'MainActivity.class'. The right pane displays the decompiled Java code for MainActivity.class. The code includes imports for Android classes, a class declaration 'public class MainActivity' extending 'android.app.Activity' and implementing 'View.OnClickListener', and methods for 'i()', 'onClick()', and 'onCreate()'. The 'i()' method returns 'this.n.getText().toString().equalsIgnoreCase("sabin")'. The 'onClick()' method calls 'i()' and sets text on a TextView based on the result. The 'onCreate()' method calls 'super.onCreate()', sets the content view, finds UI elements by ID, and sets an onClickListener.

```
package com.example.sabin.myapplication;

import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class MainActivity
    extends AppCompatActivity
    implements View.OnClickListener
{
    TextView m;
    EditText n;
    Button o;

    private boolean i()
    {
        return this.n.getText().toString().equalsIgnoreCase("sabin");
    }

    public void onClick(View paramView)
    {
        if (i())
        {
            this.m.setText("You are a VIP user!");
            return;
        }
        this.m.setText("Sorry! your are not a VIP user. You need to pay up!");
    }

    protected void onCreate(Bundle paramBundle)
    {
        super.onCreate(paramBundle);
        setContentView(2130968603);
        this.m = ((TextView)findViewById(2131427417));
        this.n = ((EditText)findViewById(2131427415));
        this.o = ((Button)findViewById(2131427416));
        this.o.setOnClickListener(this);
    }
}
```

In this code we have to change the return value in the Boolean i() function so that it always return true so that everyone is VIP user.

To modify reversed code we need to use apktool

```
PS H:\reverse engineering\experiment\sabin bir> .\apktool_2.7.0.jar d .\myapp.apk
PS H:\reverse engineering\experiment\sabin bir> |
```

File Explorer view of the directory: This PC > New Volume (H:) > reverse engineering > experiment > sabin bir > myapp >

Name	Date modified	Type	Size
original	30-06-2023 09:14 AM	File folder	
res	30-06-2023 09:14 AM	File folder	
smali	30-06-2023 09:14 AM	File folder	
AndroidManifest.xml	30-06-2023 09:14 AM	Microsoft Edge HT...	1 KB
apktool.yml	30-06-2023 09:14 AM	YML File	1 KB

```
MainActivity.smali
File Edit View

.class public Lcom/example/sabin/myapplication/MainActivity;
.super Landroid/support/v7/app/c;

# interfaces
.implements Landroid/view/View$OnClickListener;

# instance fields
.field m:Landroid/widget/TextView;

.field n:Landroid/widget/EditText;

.field o:Landroid/widget/Button;

# direct methods
.method public constructor <init>()V
    .locals 0

    invoke-direct {p0}, Landroid/support/v7/app/c;-><init>()V

    return-void
.end method

.method private i()Z
    .locals 2

   iget-object v0, p0, Lcom/example/sabin/myapplication/MainActivity;->n:Landroid/widget/EditText;

    invoke-virtual {v0}, Landroid/widget/EditText;->getText()Landroid/text/Editable;

    move-result-object v0

    invoke-virtual {v0}, Ljava/lang/Object;->toString()Ljava/lang/String;

    move-result-object v0

    const-string v1, "sabin"

    invoke-virtual {v0, v1}, Ljava/lang/String;->equalsIgnoreCase(Ljava/lang/String;)Z

    move-result v0
```

In this reversed code we found the i() function

```

.method private i()Z
    .locals 2

    iget-object v0, p0, Lcom/example/sabin/myapplication/MainActivity;->n:Landroid/widget/EditText;

    invoke-virtual {v0}, Landroid/widget/EditText;->getText()Ljava/lang/Editable;

    move-result-object v0

    invoke-virtual {v0}, Ljava/lang/Object;->toString()Ljava/lang/String;

    move-result-object v0

    const-string v1, "sabin"

    invoke-virtual {v0, v1}, Ljava/lang/String;->equalsIgnoreCase(Ljava/lang/String;)Z

    move-result v0

    if-eqz v0, :cond_0

    const/4 v0, 0x1

    :goto_0
    return v0

    :cond_0
    const/4 v0, 0x0

    goto :goto_0
.end method

```

At bottom we found the if-else statement and the variable “V0” is used to return the true/false. So we need to change it as 1 which is always true.

```

if-eqz v0, :cond_0

const/4 v0, 0x1

:goto_0
return v0

:cond_0
const/4 v0, 0x1|

goto :goto_0
end method

```


Now we have made the APK with our modified code

```

PS H:\reverse engineering\experiment\sabin bir> .\apktool_2.7.0.jar b .\myapp
PS H:\reverse engineering\experiment\sabin bir> |

```

This PC > New Volume (H:) > reverse engineering > experiment > sabin bir > myapp > dist

Name	Date modified	Type	Size
 myapp.apk	30-06-2023 09:42 AM	BlueStacks.Apk	786 KB

Now final step we need to sign the APK using keytool and jarsigner,

```

PS H:\reverse engineering\experiment\sabin bir\myapp\dist> keytool -genkey -keystore hacker.keystore -validity 1000 -alias hacker
Enter keystore password:
Re-enter new password:
What is your first and last name?
[Unknown]: hacker 0
What is the name of your organizational unit?
[Unknown]:
What is the name of your organization?
[Unknown]:
What is the name of your City or Locality?
[Unknown]:
What is the name of your State or Province?
[Unknown]:
What is the two-letter country code for this unit?
[Unknown]:
Is CN=hacker 0, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown correct?
[no]: yes

PS H:\reverse engineering\experiment\sabin bir\myapp\dist> |

```

```

PS H:\reverse engineering\experiment\sabin bir\myapp\dist> jarsigner -keystore .\hacker.keystore -verbose myapp.apk hacker
Enter Passphrase for keystore:
adding: META-INF/MANIFEST.MF
adding: META-INF/HACKER.SF
adding: META-INF/HACKER.DSA
signing: AndroidManifest.xml
signing: classes.dex
signing: res/anim/abc_fade_in.xml
signing: res/anim/abc_fade_out.xml
signing: res/anim/abc_grow_fade_in_from_bottom.xml
signing: res/anim/abc_popup_enter.xml
signing: res/anim/abc_popup_exit.xml
signing: res/anim/abc_shrink_fade_out_from_bottom.xml
signing: res/anim/abc_slide_in_bottom.xml
signing: res/anim/abc_slide_in_top.xml
signing: res/anim/abc_slide_out_bottom.xml
signing: res/anim/abc_slide_out_top.xml
signing: res/color/abc_btn_colored_borderless_text_material.xml
signing: res/color/abc_btn_colored_text_material.xml
signing: res/color/abc_hint_foreground_material_dark.xml
signing: res/color/abc_hint_foreground_material_light.xml

```

```

signing: res/mipmap-hdpi-v4/ic_launcher.png
signing: res/mipmap-mdpi-v4/ic_launcher.png
signing: res/mipmap-xhdpi-v4/ic_launcher.png
signing: res/mipmap-xxhdpi-v4/ic_launcher.png
signing: res/mipmap-xxxhdpi-v4/ic_launcher.png
signing: resources.arsc

>>> Signer
X.509, CN=hacker 0, OU=Unknown, O=Unknown, L=Unknown, ST=Unknown, C=Unknown
[trusted certificate]

jar signed.

Warning:
The signer's certificate is self-signed.
PS H:\reverse engineering\experiment\sabin bir\myapp\dist> |

```

## CONCLUSION:

We have successfully reverse engineered and modified the app now every user is VIP user.