

بسم الله الرحمن الرحيم

شرح خطوات حل ctf

عند تحميل الملف من الرابط التالي: <https://t.co/9GWxdwORhZ>
تحصل على ملف مضغوط بصيغه rar
قم بفك الملف واستخدام كلمة السر : opcde2018

بعد فك الملف نحصل على الملفات التالية



ctf.apk

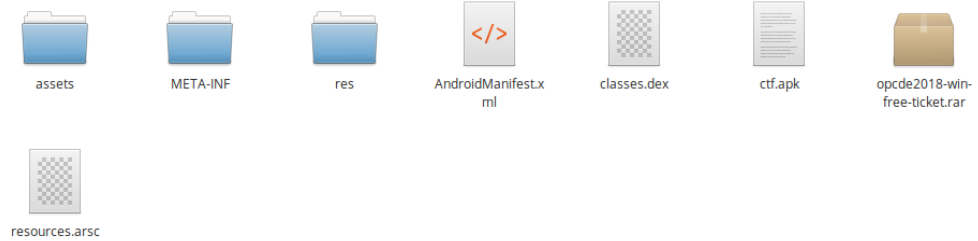


opcde2018-win-
free-ticket.rar

يتبين لنا ان هناك ملف تطبيق اندرويد apk
أول خطوه نحو هذا الملف نقوم بفكه عن طريق unzip خطوه مبدئية – هناك طرق أخرى اذا احتجنا)
لفتح الملف قم بالضغط باليمين على نفس الملف والضغط على استخراج او بالامر التالي عن طريق التريمنال

```
opcde2018-win-free-ticket.rar  
~/Downloads/ccana/untitled folder$ unzip ctf.apk  
ctf.apk  
ng: META-INF/MANIFEST.MF  
ng: META-INF/APKKEY.SF  
ng: META-INF/APKKEY.DSA  
ng: AndroidManifest.xml  
ng: assets/ctf.png
```

بعد ذلك نجد ملفات اكثر كما هو موضح



نقوم بفحص واستكشاف هذه الملفات وفتحها والتعرف على مافي داخلها

بعد البحث نجد ان مجلد assets يحتوي على صوره من نوع png

كما هو موضح بالصورة نقوم بفحص الملف والتعرف عليه

```
abby@abby:~/Downloads/ccana/untitled folder/assets$ ls
ctf.png
abby@abby:~/Downloads/ccana/untitled folder/assets$ file ctf.png
ctf.png: data
abby@abby:~/Downloads/ccana/untitled folder/assets$ hexdump ctf.png | head
00000000 429b 555c 181f 1808 1212 1f12 5a5b 4056
00000010 1212 4313 1212 8f12 101a 1212 9012 ce5b
00000020 127c 1212 6113 5540 1250 dcbc fb0e 1212
00000030 1612 5375 535f 1212 9da3 ee19 1773 1212
00000040 1b12 5a62 614b 1212 6600 1212 6600 cc13
00000050 0d74 126a ed12 5bb7 5356 6a46 6e4c 15ef
00000060 36a6 67d5 9837 48d4 8787 aff7 cce5 fd69
00000070 adaf c9cc cfac 6fa5 7a11 002a 169c 025b
00000080 3226 0813 930a 07b0 c21b 3601 c353 8690
00000090 780b df36 85de cc74 e9de 9dfd 5483 3a74
abby@abby:~/Downloads/ccana/untitled folder/assets$
```

الامر file يعرض نوع الملف ولكن لم يتعرف على نوع الملف فقمنا بعرض hexdump للملف للتعرف على PE للملف

بداية الملف 429b لاتعني شي حاليا ويبدو ان الملف مشفر

في هذه الحالة نقوم بمحاولة كسر الملف عن طريق XOR باستخدام أداة xortool

```

root@kali:/media/sf_Windows7/assets/test# ls
ctf.png
root@kali:/media/sf_Windows7/assets/test# xortool ctf.png -l 10 -c 00
1 possible key(s) of length 10:
\x12\x12\x12\x12\x12\x12\x12\x12\x12\x12
Found 0 plaintexts with 95.0%+ printable characters
See files filename-key.csv, filename-char_used-perc_printable.csv
root@kali:/media/sf_Windows7/assets/test# ls
ctf.png xortool_out
root@kali:/media/sf_Windows7/assets/test# cd xortool_out/
root@kali:/media/sf_Windows7/assets/test/xortool_out# ls
0.out filename-char_used-perc_printable.csv filename-key.csv
root@kali:/media/sf_Windows7/assets/test/xortool_out#

```

باستخدام xor قمنا بفك الملف بنجاح وتم عرض الملف الأصلي 0.out

هذه الملفات تم استخراجها بواسطة اداة xortool

الان نقوم بفحص هذه الملفات – الخطوة الأولى هي التعرف على نوع الملف

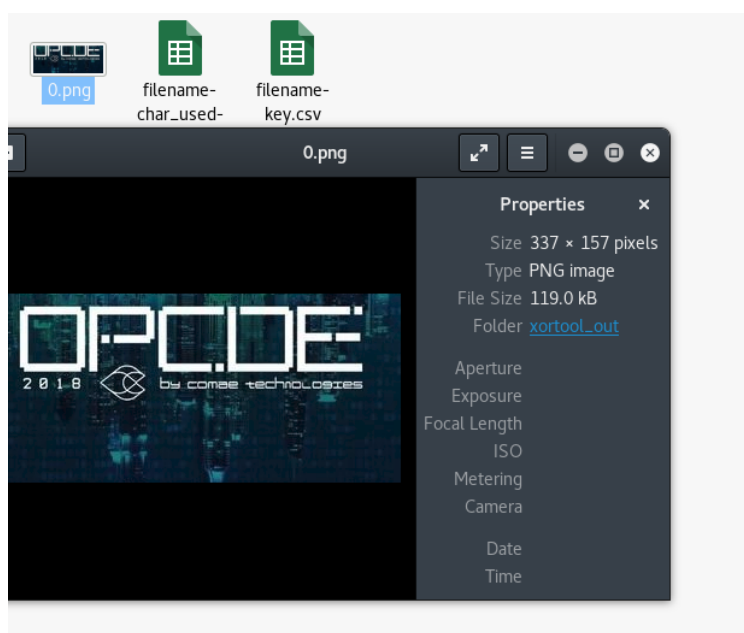
```

root@kali:/media/sf_Windows7/assets/test/xortool_out# file 0.out
0.out: PNG image data, 337 x 157, 8-bit/color RGB, non-interlaced
root@kali:/media/sf_Windows7/assets/test/xortool_out# mv 0.out 0.png
root@kali:/media/sf_Windows7/assets/test/xortool_out# ls
0.png filename-char_used-perc_printable.csv filename-key.csv

```

بعد التعرف على الملف نقوم بتغيير الصيغة من 0.out الى 0.png

بعد ذلك نفتح الصورة من الممكن تكون هي النتيجة لل Flag



بعد فتح الصورة لا يظهر الا شعار المؤتمر ولا يعني شي . الغريب في الامر ان حجم مقاسات الصورة لا يتناسب مع حجم الملف.

الخطوة التالية نتعرف على الملف لانه من الممكن انه يخفي ملفات أخرى بداخل هذه الصورة وهذا ما يعرف بال Steganography

نقوم بفتح الملف عن طريق hexeditor

بعد فتح الملف عن طريق التيرمينال والبحث عن بعض الرموز التي تبين اذا كان هناك ملفات مخفيه قمت بالبحث عن PK وهو الترميز لملفات ZIP - الملفات المضغوطة

بعد ذلك نقوم بحاوله فك واستخراج هذه الملفات من الصورة هناك أداة foremost تقوم باستخراج الملفات بالتعرف على الترميز تبعهم

```
root@kali:/media/sf_Windows7/assets/test/xortool_out# ls
0.png filename-char_used-perc_printable.csv > filename-key.csv
root@kali:/media/sf_Windows7/assets/test/xortool_out# foremost -t zip 0.png -o zip
Processing: 0.png g name="app name">Settings</string>
| foundat=META-INF/MANIFEST.MF000000 URL">https://comae.typeform.com/to/XyWXdS</string>
*| </resources>
root@kali:/media/sf_Windows7/assets/test/xortool_out# ls
xortool_out/zip/zip/test22/res/va
0.png filename-char_used-perc_printable.csv filename-key.csv zip
root@kali:/media/sf_Windows7/assets/test/xortool_out#
```

بعد ذلك نذهب للمجلد zip والذي قمنا كتابة الملفات المستخرجة من الصورة اليه

```
root@kali:/media/sf_Windows7/assets/test/xortool_out# cd zip/1f030001" />
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip# llsets/xortool_out/zip
bash: ll: command not found
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip# ls
audit.txt e zip ces>
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip# cd zip/
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip/zip# lscom/to/XyWXdS</str
00000195.zipources>
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip/zip# file 00000195.zip2/
00000195.zip: Zip archive data, at least v2.0 to extract
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip/zip#
```

يتبين لنا انه ملف zip file صحيح

استخدام foremost نقوم باستخراج الملفات وكتابتها بال binary لذلك يجب ان نقوم بإصلاح الملف المضغوط المستخرج باستخدام الامر التالي


```

root@kali:/media/sf_Windows7/assets/test/xortool_out/zip/zip# zip -FF 00000195.zip --out file.apk
Fix archive (-FF) - salvage what can
Found end record (EOCDR) - says expect single disk archive
Scanning for entries...
copying: META-INF/MANIFEST.MF (309 bytes)
copying: META-INF/APKKEY.SF (407 bytes)
copying: META-INF/APKKEY.DSA (762 bytes)
copying: AndroidManifest.xml (1114 bytes)
copying: classes.dex (7287 bytes)
copying: res/drawable/ic_launcher.png (7301 bytes)
copying: resources.arsc (972 bytes)
Central Directory found...
EOCDR found ( 1 19022)...
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip/zip# apktool d file.apk
I: Using Apktool 2.2.1-dirty on file.apk
I: Loading resource table...
I: Decoding AndroidManifest.xml with resources...
I: Loading resource table from file: /root/.local/share/apktool/framework/1.apk
I: Regular manifest package...
I: Decoding file-resources...
I: Decoding values */* XMLs...
I: Baksmaling classes.dex...
I: Copying assets and libs...
I: Copying unknown files...
I: Copying original files...
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip/zip#

```

بعد البحث داخل هذه الملفات . احد الملفات كان يحتوي على رابط لارسال الحل ☺ وهو ملف strings.xml بعد فتحه يظهر بالشكل التالي

```

GNU nano 2.7.4                                File: strings.xml
Desktop Pictures VBoxLinuxAdditions.run
<?xml version="1.0" encoding="utf-8"?> Videos
<resources>ds pythonblackhat WiFi-Pumpkin
  <string name="app_name">Settings</string>tool
  <string name="SubmissionURL">https://comae.typeform.com/to/XyWXdS</string>
</resources>input files
root@kali:~# ld
ld: no input files
root@kali:~# ls
AutoSploit Music untitled-3.py
Desktop Pictures VBoxLinuxAdditions.run

```

ولكن مازال البحث عن ال Flag

ملفات الجافا لا زالت باينري ولكن عندما فتحنا ملف apk باستخدام unzip ظهر لدينا ملف classes.dex هذا النوع من الملفات نستطيع فكه وتحويل ملفات java classes الى الكود الأداة المستخدمه jadx

```

root@kali:/media/sf_Windows7/assets/test/xortool_out/zip/zip# jadx/build/jadx/bin/jadx classes.dex
INFO - output directory: classes
INFO - loading ...
INFO - processing ...
INFO - done
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip/zip# ls
00000195.zip AndroidManifest.xml classes classes.dex file file.apk jadx META-INF res resources.arsc
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip/zip# cd classes/
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip/zip/classes# ls
sources
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip/zip/classes# cd sources/
android/ com/
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip/zip/classes# cd sources/com/android/setting/
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip/zip/classes/sources/com/android/setting# ls
A.java B.java BuildConfig.java C.java D.java E.java F.java R.java Settings.java
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip/zip/classes/sources/com/android/setting# head E.java
package com.android.setting;

import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.os.Environment;
import java.io.ByteArrayOutputStream;
import java.io.File;
import java.io.FileOutputStream;
root@kali:/media/sf_Windows7/assets/test/xortool_out/zip/zip/classes/sources/com/android/setting#

```

يتضح لنا انه تم تحويل الملفات للكود
بعد البحث في الملفات وجدنا شفره

```

GNU nano 2.7.4 File: E.java
context.stopService(new Intent(context, Settings.class));
};
private Context b;
private String p = (Environment.getExternalStorageDirectory() + "/log");
public E(Context context) {
    this.b = context;
    IntentFilter filter = new IntentFilter();
    filter.addAction("android.intent.action.PACKAGE_ADDED");
    filter.addAction("android.intent.action.PACKAGE_CHANGED");
    filter.addDataScheme("package");
    context.registerReceiver(this.a, filter);
}
public void run() {
    IOException e;
    Throwable th;
    File file = new File(this.p);
    InputStream in = null;
    FileOutputStream fileOutputStream = null;
    try {
        in = this.b.getAssets().open("GRDDKMBUGM2DINBVGQZDIMRVGEZEMNCEGQ3TKRRUHE2UMN3XGRDDIRI=");
        if (file.exists()) {
            file.delete();
        }
    }
}

```

بعد ذلك نقوم بنسخ الشفره وفكها وهناك مواقع وأدوات كثيره
نحرب base32
باستخدام موقع decode.fr

Informatics > Character Encoding > Base32

Sponsored ads

Base32 Decoder

★ BASE 32 CIPHERTEXT

GRDDKMBUGM2DINBVGQZDIMRVGE2EMNCEGO3TKRRUHE2UMNIXGRDDI
Bf=

DECRYPT

→ Base64 Coding

Search for a tool

★ SEARCH A TOOL ON DCODE BY KEYWORDS:

e.g. type caesar GO

Results

4F504344454242514F4D475F495F574F4E

Base32 - dCode

Tag(s) : Character Encoding, Informatics

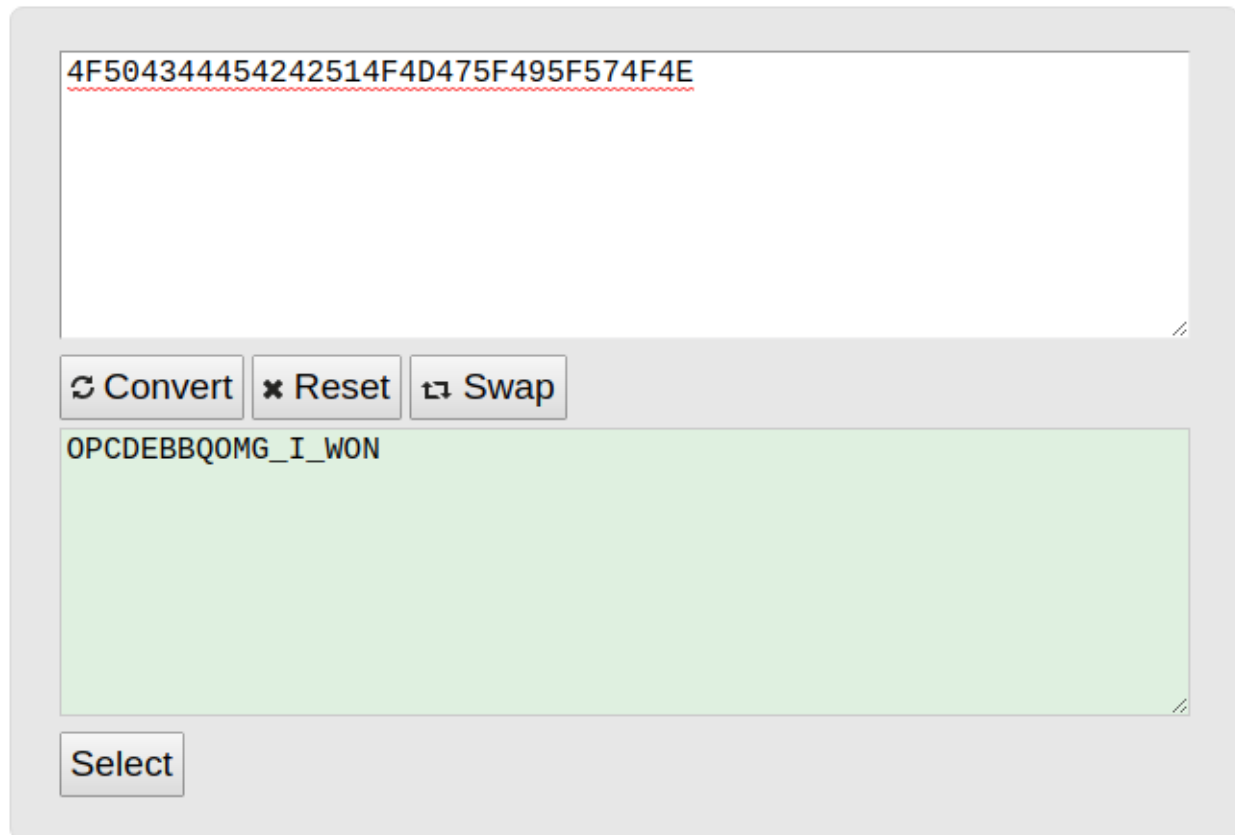
dCode and you

كما هو واضح على اليسار تم فك الشفرة ويتضح لنا ان هناك شفره أخرى ..
بالنظر السريع للشفرة يبدو انها hex
نحاول بتحويل ال hex الى ascii قد يكون هناك امر ما!
باستخدام موقع rapidtables.com
نقوم بتحويل الشفرة من hex الى ascii

Hex to ASCII text converter

Hex to [ASCII text](#) converter.

Enter 2 digits hex numbers with any prefix / postfix / delimiter and press the *Convert* button (e.g. FF 43 5A 7F):



The screenshot shows a web-based hex to ASCII converter. At the top, a text input field contains the hex string "4F504344454242514F4D475F495F574F4E", which is underlined with a red dashed line. Below the input field are three buttons: "Convert" (with a circular arrow icon), "Reset" (with an 'x' icon), and "Swap" (with a double-headed arrow icon). The "Convert" button is highlighted. Below the buttons is a large green text area displaying the converted ASCII string "OPCDEBBQOMG_I_WON". At the bottom left of the interface is a "Select" button.

كما هو موضح بالصورة تبدو ان هذه هي Flag لأنها تعني opcde bbq OMG I won

Flag is: OPCDEBBQOMG_I_WON