

Mobil Uygulama Güvenliğine Saldırgan Yaklaşım

Mobil Uygulama Sızma Testi Eğitimi

Ahmet GÜREL
www.gurelahmet.com

Android Application Security

Başlamadan Önce ...

- Open Handset Alliance liderliğinde Google firması tarafından akıllı telefon ve tablet bilgisayarlar gibi mobil cihazlar için geliştirilmiş **Linux** tabanlı işletim sistemi.
- Android cihazlarda uygulamaların çalışabilmesi için .apk uzantılı dosyalar ile uygulamalar yüklenir ve cihazlara dağıtabilir.
- Günümüzde Native ve Hybrid uygulamalardan söz edilmekte.

Neden Android ?

- Açık kaynak kodlu
- Linux tabanlı
- Kullanım yaygınlığı - telefonlar, tabletler, arabalar...
- Gelişmiş ve ücretsiz yazılım geliştirme ortamı sunması
- Açık uygulama marketi

Android Kullanım Alanları

- Cep telefonları, tabletler, akıllı saatler vs....
- Arabalar, akıllı ev sistemleri
- Mobil bankacılık
- Internet of Things (IoT)

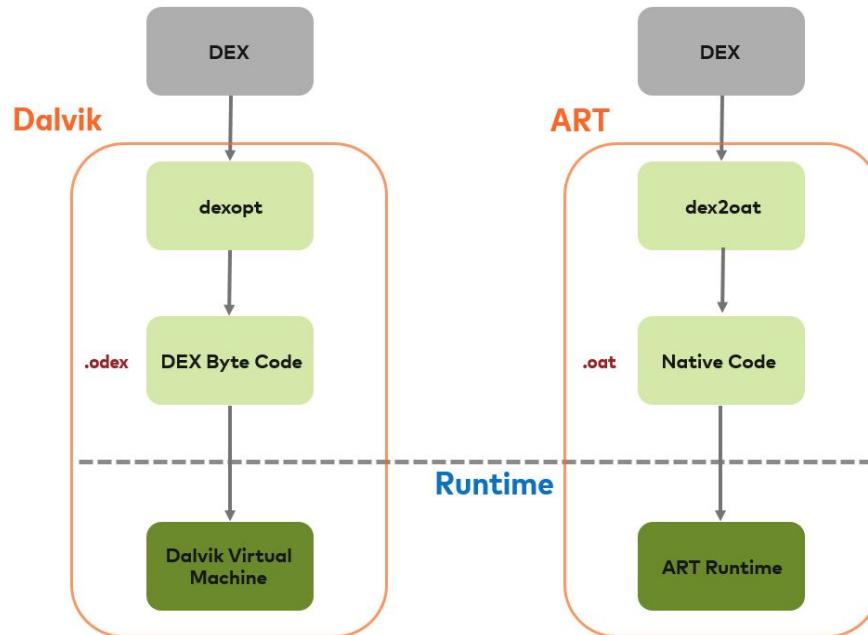
İstatistikler

- Akademik ve iş için kullanılan akıllı telefon ve tablet oranı : **%76**
- 2015-2016 akıllı telefonları hedef alan zararlı yazılım artış oranı : **%250**
- Platformlara göre zararlı yazılımlar : **Android (%73), iOS (%32)**
- Android zararlı yazılımlarının %55'i casus yazılım, %44'ü ise Truva atı.

Android Güvenlik Modeli

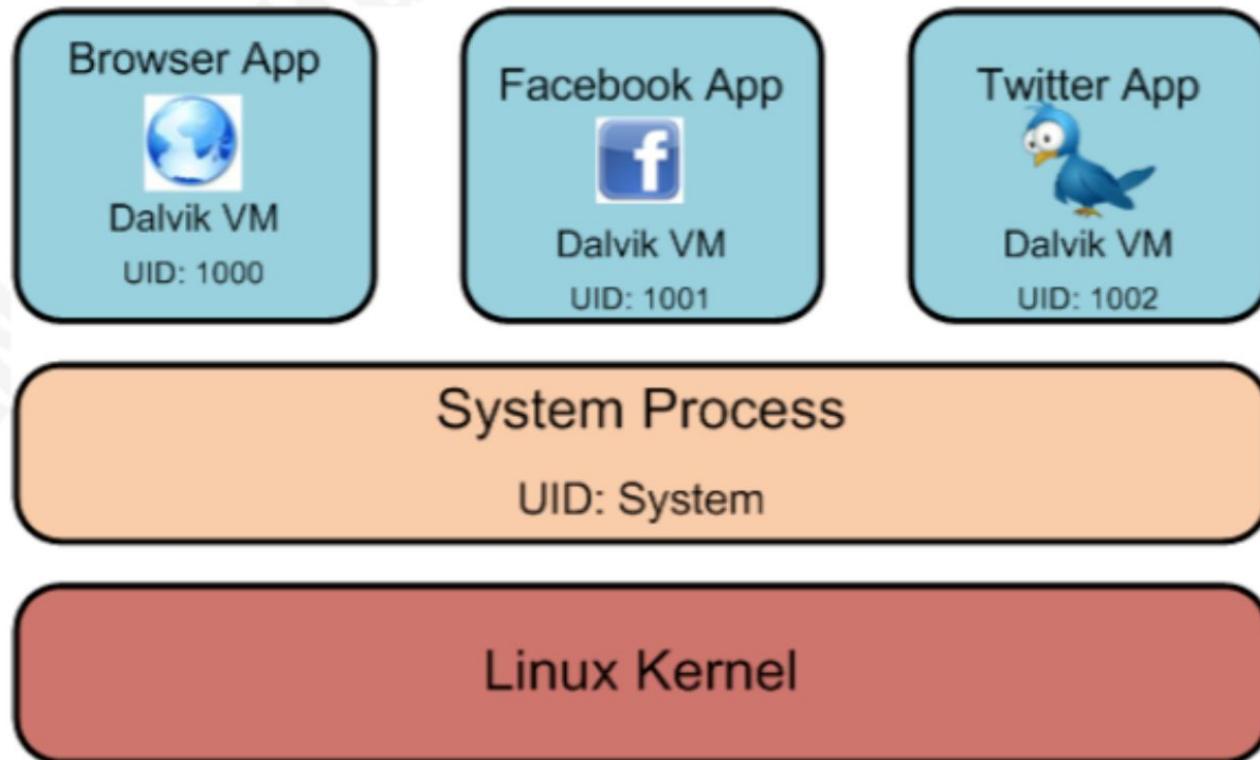
- Linux güvenlik modeli baz alınmıştır (UID/GUID).
- Uygulama bazlı izinler kullanılmaktadır.
- Uygulama izinleri, **AndroidManifest.xml** dosyasında tanımlanmaktadır.
- Uygulama kurulumu için uygulamanın sertifika ile imzalanmış olması gerekmektedir.
- Her bir uygulama farklı bir DVM(Dalvik Virtual Machine) içerisinde çalışmaktadır. Lollipop ve sonrası sürümlerde Android işletim sisteminin içinde ART (Android Run Time) sanal makinesi bulunmaya başladı.
- Sistem güvenliği açısından **kullanıcı** kilit rol oynamaktadır.
- Rootlanmamış bir cihaz için root erişimi mümkün değildir. “su” uygulaması sistemde bulunmaz.

Android Güvenlik Modeli



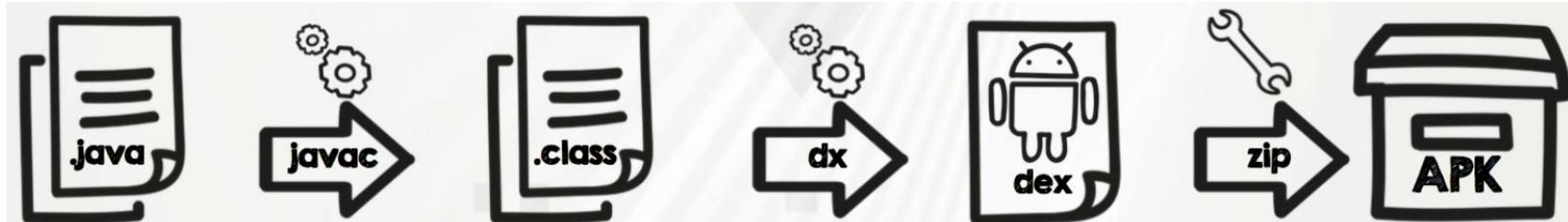
Dalvik vs ART

Android Güvenlik Modeli

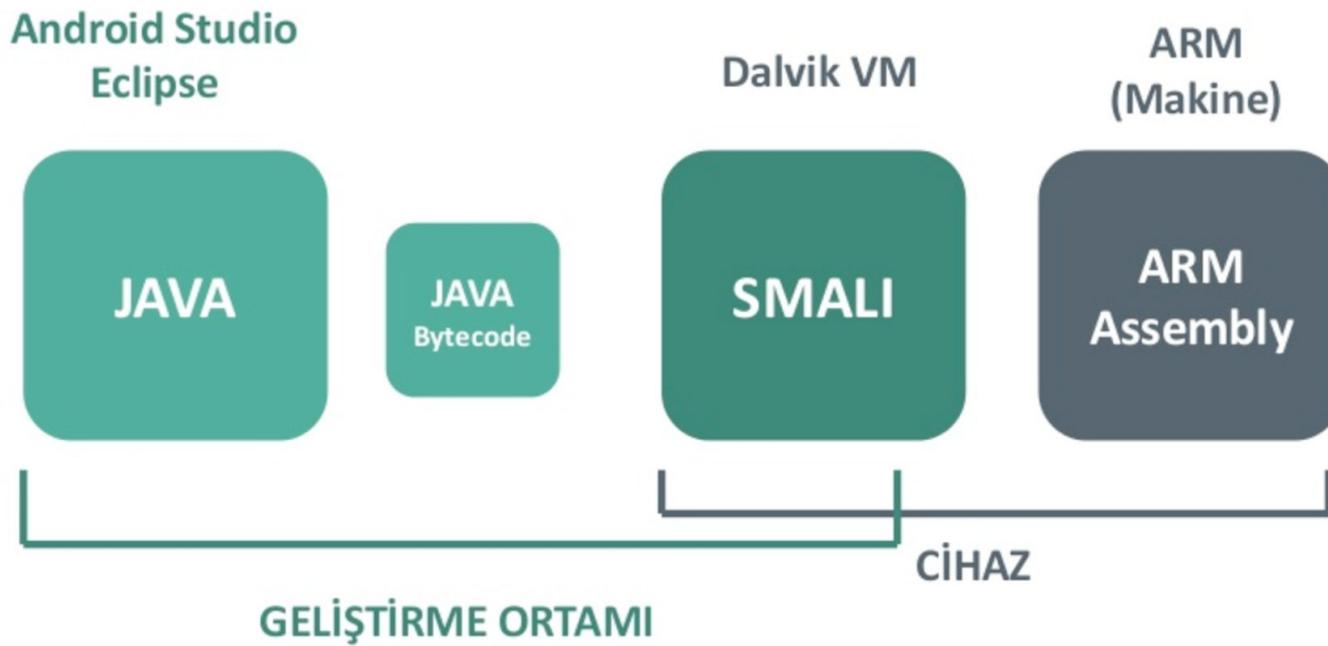


Android Uygulamaları

- Java + Android SDK ile geliştirilir.
- Android Dalvik VM ile çalıştırılır.
- JAVA -> .class -> .dex



Android Uygulamaları



JAVA Archive

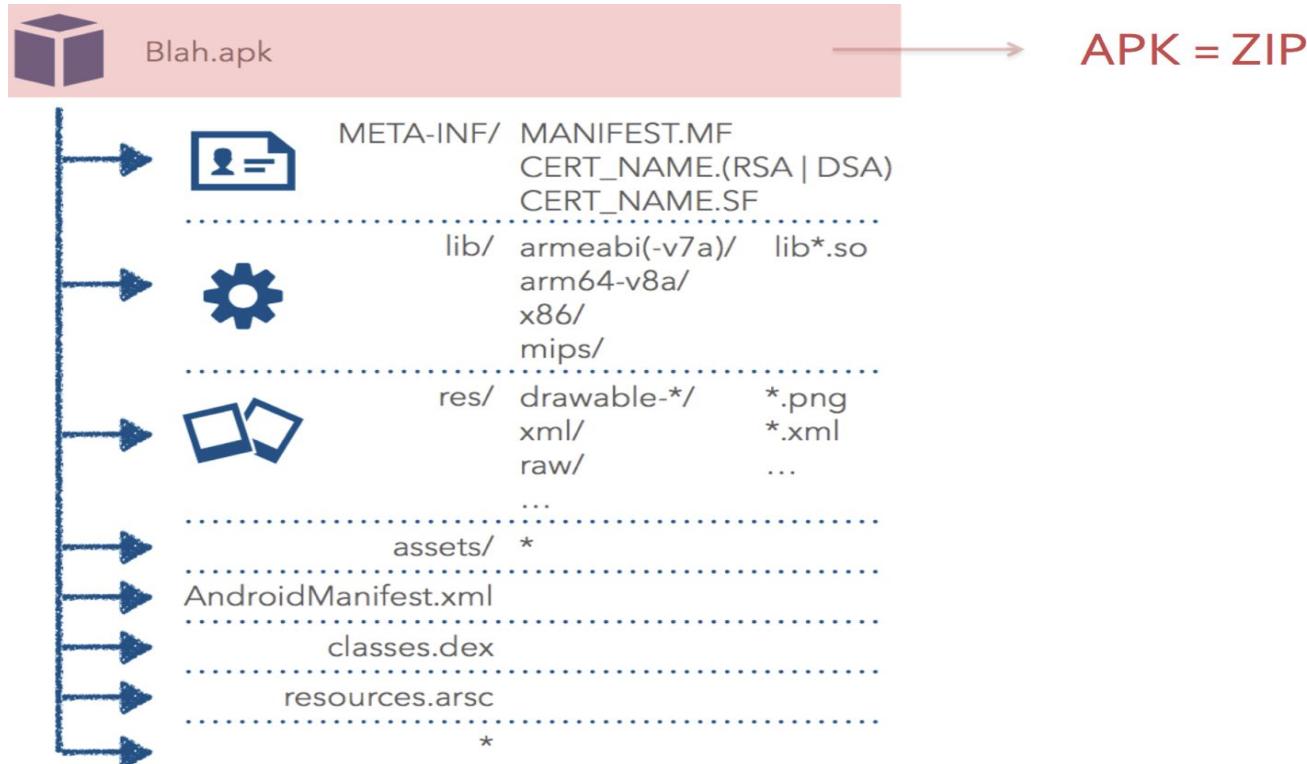
APK = JAR = ZIP

Android Application Package

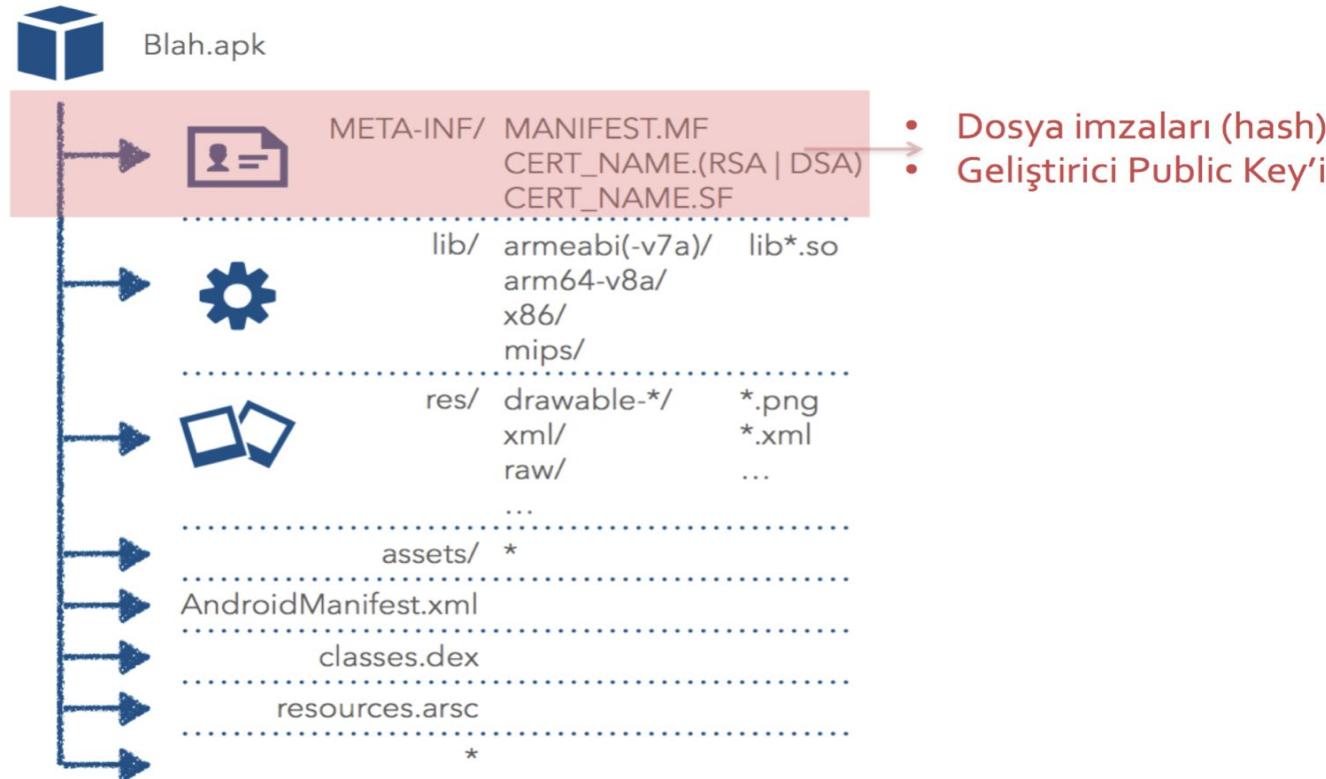
APK

- **Android Application Package File (APK)** dosyası zip dosya formatına sahip .apk uzantılı dosyalardır.
- APK dosyasının uzantısı .zip olarak değiştirildikten sonra WinZip, WinRAR gibi arşiv programları ile dosya içeriği görüntülenebilir.

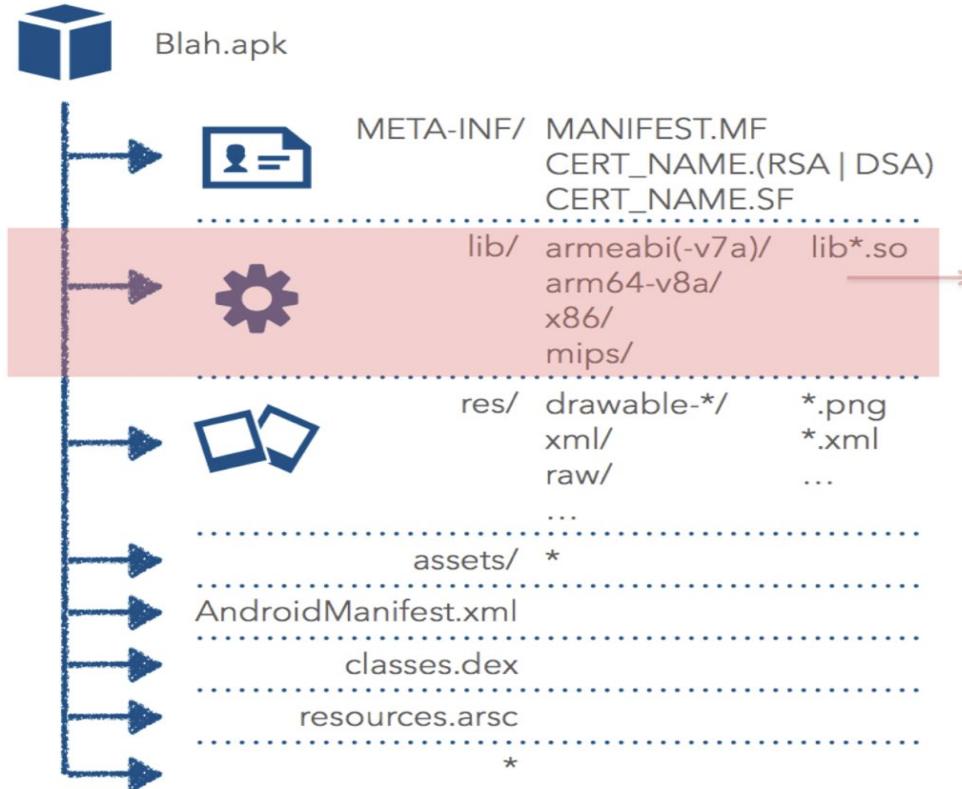
Android Paket İçeriği



Android Paket İçeriği

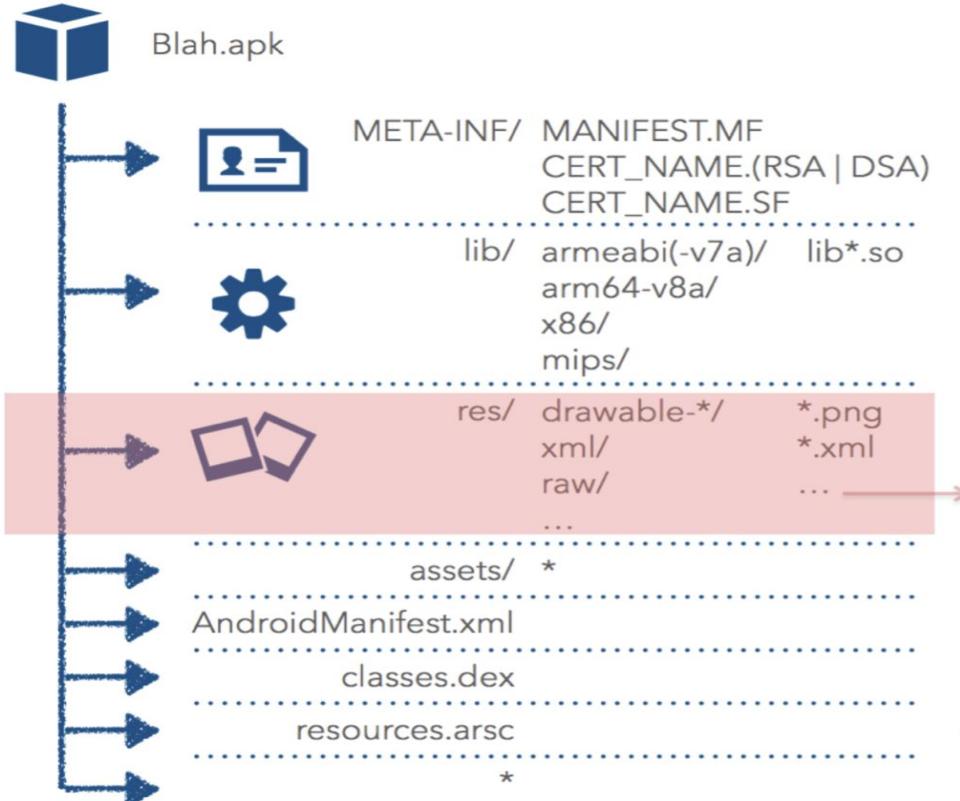


Android Paket İçeriği



- İşlemci mimarisine göre compile edilmiş native kütüphaneler (Native ELF dosyaları)
- JAR Dosyaları (kütüphaneler)

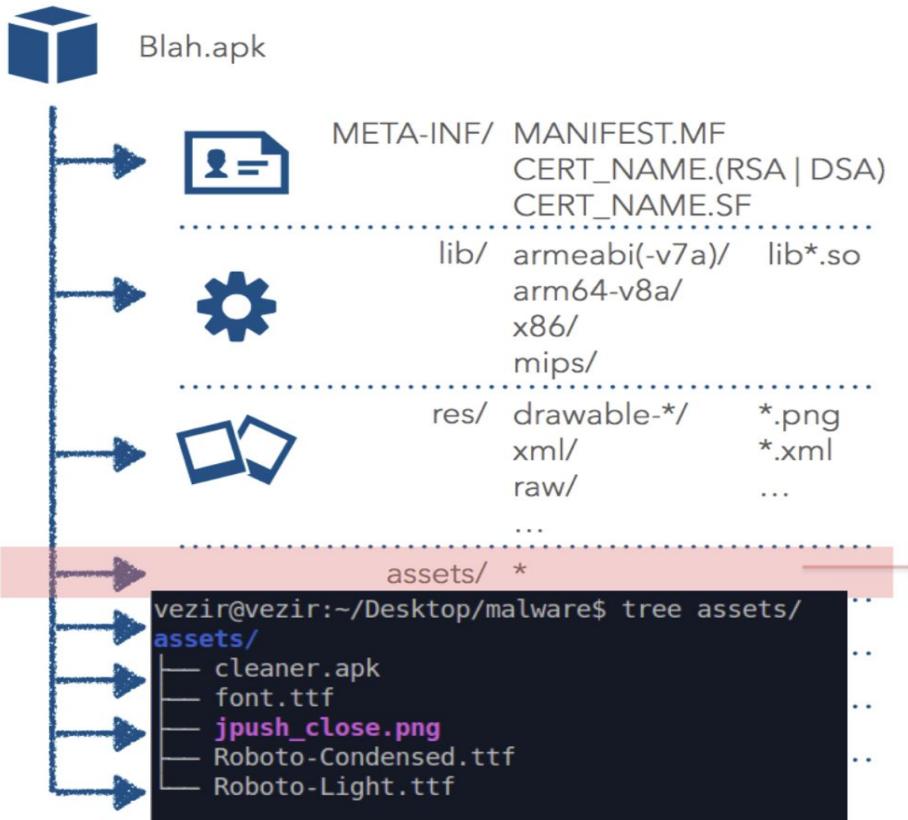
Android Paket İçeriği



- **anim:** Compile edilmiş animasyon dosyaları
- **drawable:** Resim dosyaları
- **layout:** UI/view tanımlamaları
- **values:** Diziler, renkler, style'lar, string'ler dimensions
- **xml:** Compile edilmiş XML dosyaları
- **raw:** Compile edilmemiş raw dosyalar

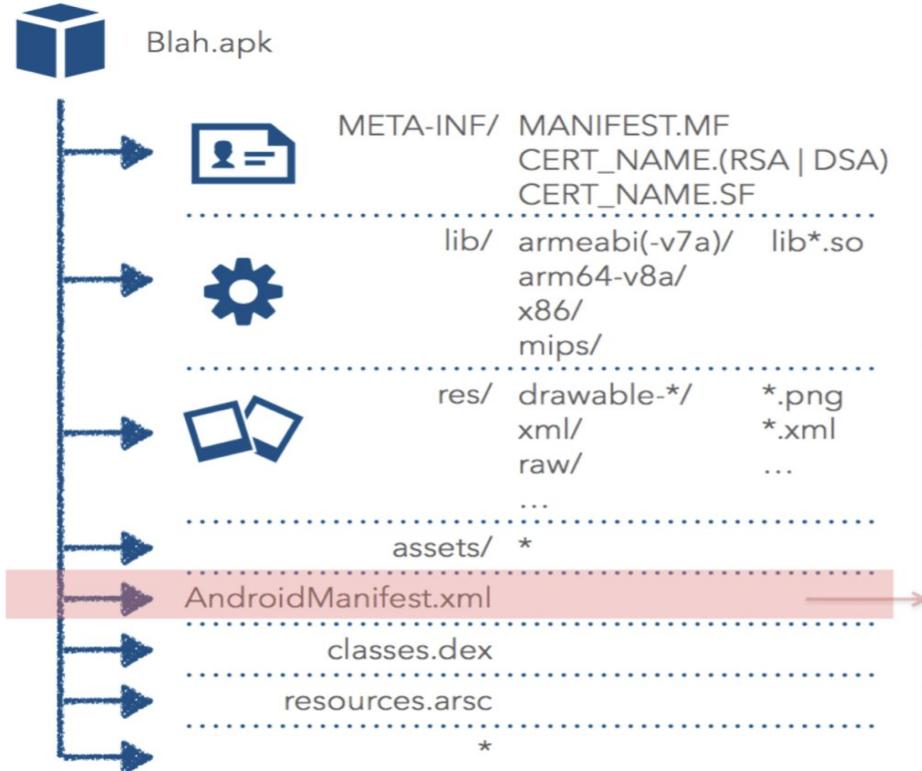
Compile işlemi AAPT (Android Asset Packaging Tool) tarafından yapılır

Android Paket İçeriği



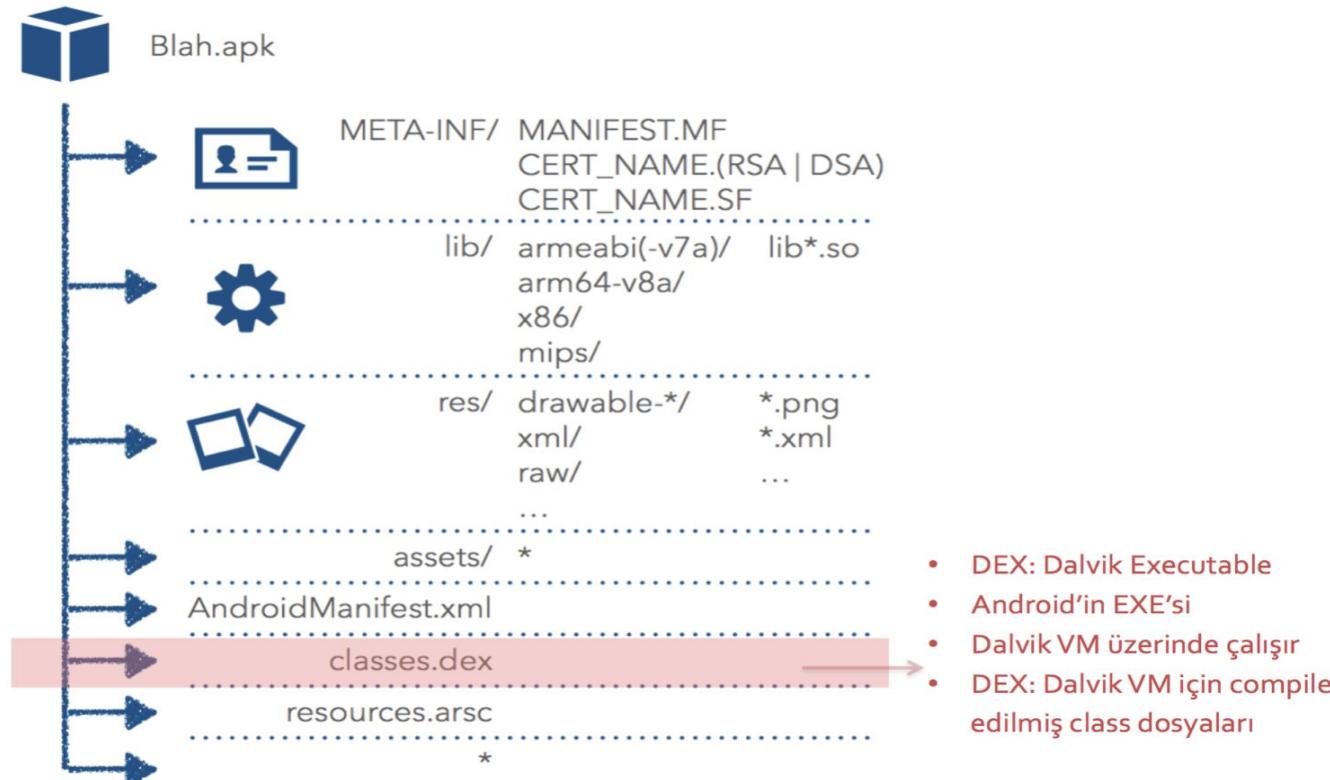
- Çoğu zaman raw dosyalar bulunur.
- Resimler, fontlar, ses dosyaları
- Bazı malware'ler bu dizinde cihaza kurrmak üzere APK dosyaları saklarlar

Android Paket İçeriği

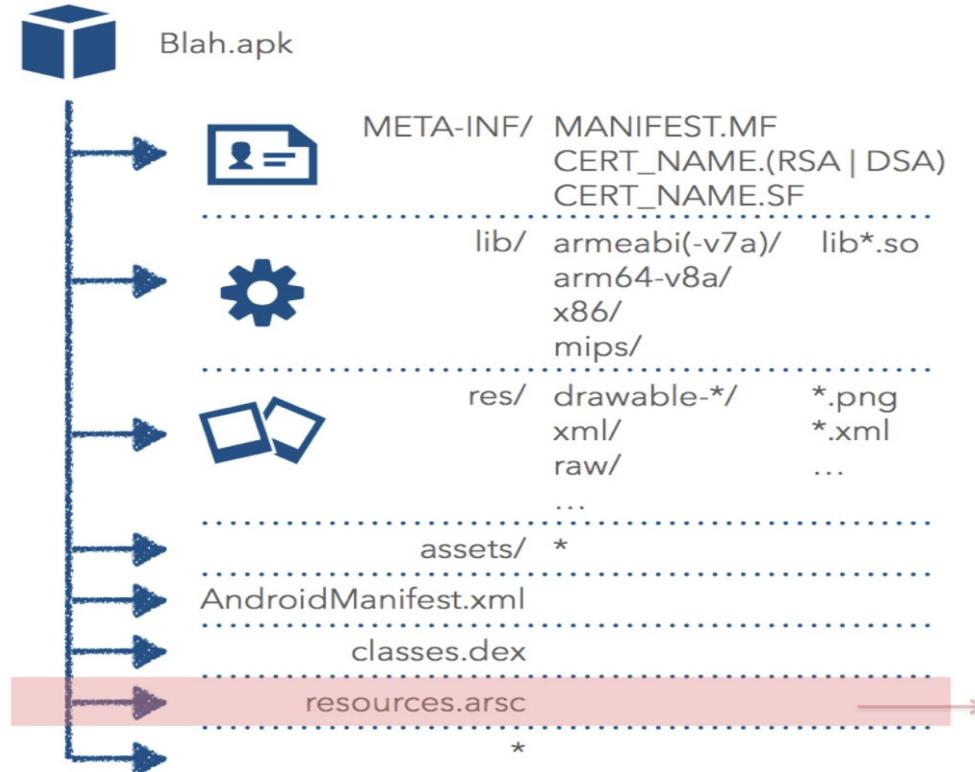


- Uygulama meta-data'ları
 - Paket ismi
 - Versiyon bilgisi
 - ...
- Uygulamanın talep ettiği izinler
- Uygulamada bulunan komponentler
 - Activity
 - Service
 - Broadcast Receiver
 - Content Provider
- Compile edilmiş olarak paket içerisinde yer alır.

Android Paket İçeriği



Android Paket İçeriği



- Compile edilmiş resource'lar
 - R.java
 - string.xml
 - ids.xml
 - layouts.xml

Kaynak Kod Dönüşümü - Decompile



Decompile

Dex → JAR → JAVA



**dex2jar
enjarify**

Java Decompiler

- JD-GUI
- JAD
- JADX
- Procyon
- ...



Kaynak Kod Dönüşümü - Decompile

- Dex2jar aracı ile class dosyasına dönüştürülmüş olan Android uygulaması, JD-GUI aracı ile kaynak koduna (decompile) geri çevirilebilir.
- Decompile işleminin yetersiz olduğu durumlarda incelenecek uygulamayı Disassembling işleminden geçirmek gerekebilir.

Android Tersine Mühendislik

Dex2Jar

Adından da anlaşılacağı üzere dex dosyalarını jar dosyalarına çevirmektedir.

Resimde görüldüğü üzere apk dosyamızı jar haline getirdik.

Android Tersine Mühendislik

Dex2Jar

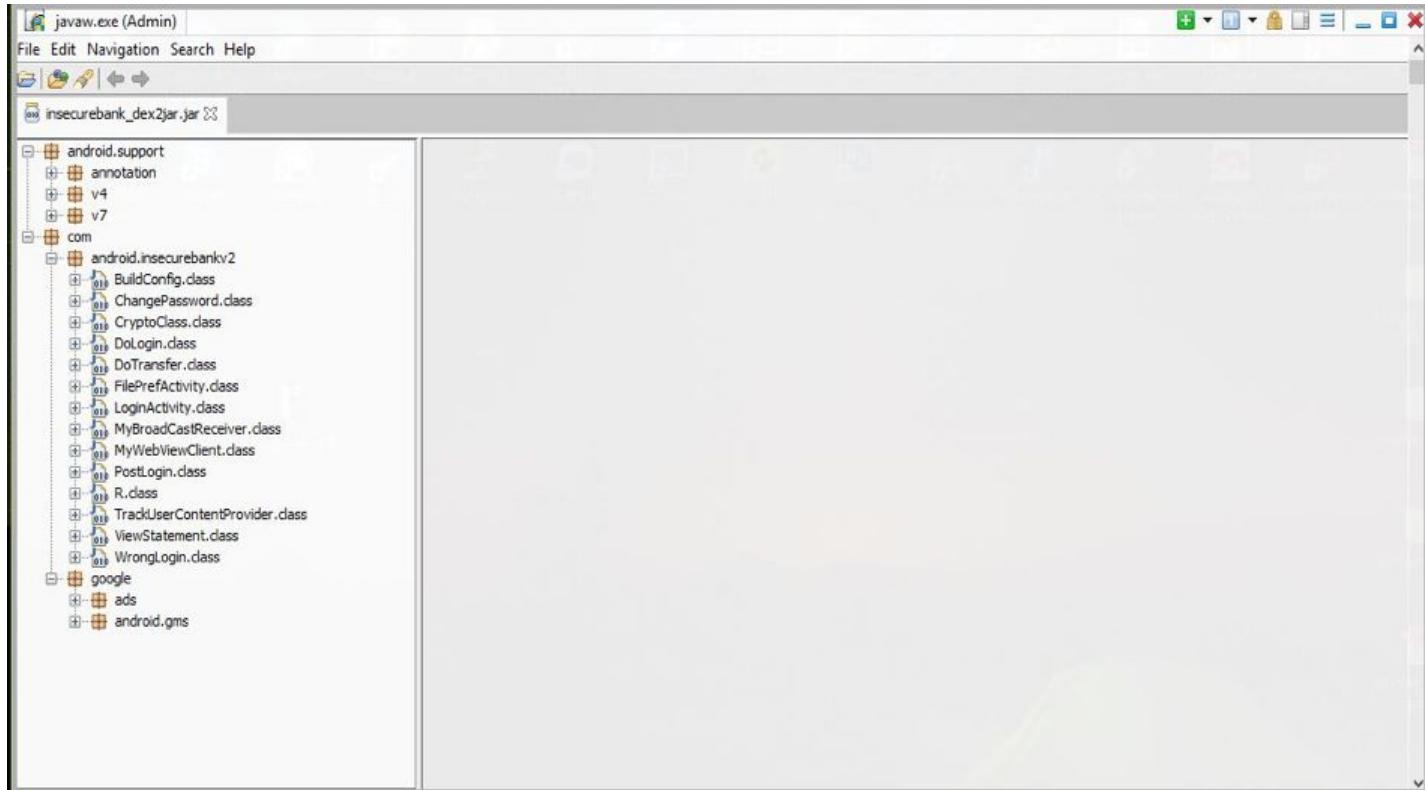
```
C:\Mobil Pentest\dex2jar-0.0.9.15>dex2jar.bat insecurebank.apk
this cmd is deprecated, use the d2j-dex2jar if possible
dex2jar version: translator-0.0.9.15
dex2jar insecurebank.apk -> insecurebank_dex2jar.jar
Done.
```

```
C:\Mobil Pentest\dex2jar-0.0.9.15>
```

Android Tersine Mühendislik

JD-GUI

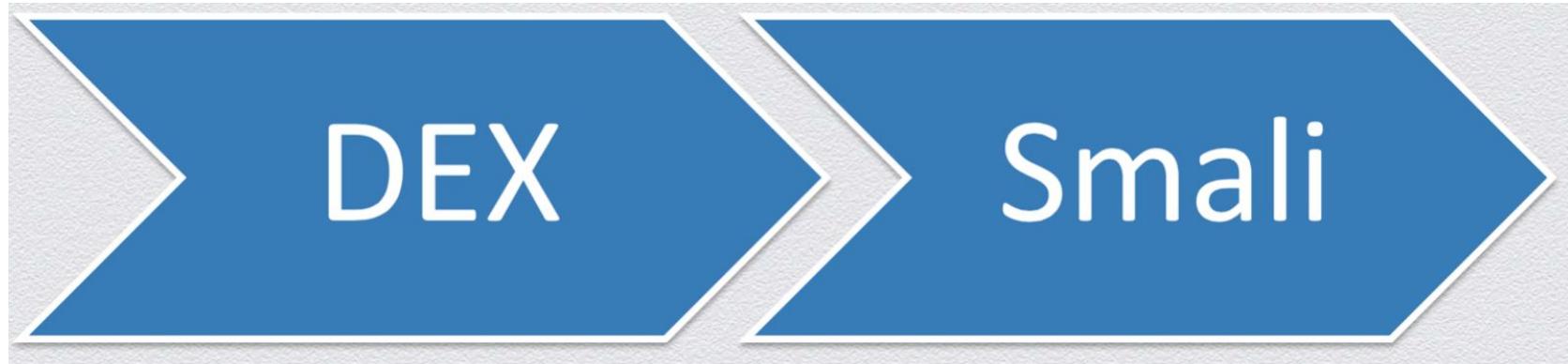
JAR haline
getirdiğimiz
dosyamızı
görüntülemek
için
kullanacağımız.



Kaynak Kod Dönüşümü - Decompile

- Decompile edilmiş JAR kodu tekrar compile edilerek çalıştırılabilir hale getirilemez.
- Decompile edilen kod yaklaşık koddur. %100 geri dönüşüm gerçekleştirilemez.
- Dex2jar çıkışlarından elde edilen JAR kodu çalıştırılamaz.
- Dalvik Bytecode, JAR koduna dönüştürülerek kolay okunabilir ve anlaşılabilir hale gelir.

Disassembling



Disassemble Dex → .smali



Dex Disassembler

- Baksmali
- Dedexer
- apktool

Mobil Sızma Testi Araçları

APKTool

APKTool apk dosyalarını decompile ederek smali kodlarına dönüştürür.

Kullanımı oldukça basit aşağıdaki resimde görüldüğü üzere b parametresi ile decompile etmekte.

Android Tersine Mühendislik

APKTool

```
komut İstemi
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. Tüm hakları saklıdır.

C:\Users\Ahmet GUREL>cd C:\

C:\>cd "Mobil Pentest\APKTool"

C:\Mobil Pentest\APKTool>apktool b insecurebank.apk
Exception in thread "main" brut.androlib.AndrolibException: brut.directory.PathNotExist: apktool.yml
    at brut.androlib.Androlib.readMetaFile(Androlib.java:143)
    at brut.androlib.Androlib.build(Androlib.java:160)
    at brut.androlib.Androlib.build(Androlib.java:155)
    at brut.apktool.Main.cmdBuild(Main.java:182)
    at brut.apktool.Main.main(Main.java:67)
Caused by: brut.directory.PathNotExist: apktool.yml
    at brut.directory.AbstractDirectory.getFileInput(AbstractDirectory.java:103)
    at brut.androlib.Androlib.readMetaFile(Androlib.java:139)
    ... 4 more

C:\Mobil Pentest\APKTool>
```

Disassembling

Disassemble DEX -> .smali

DEX dosyası okunabilir Dalvik Bytecode'a dönüştürülüyor.

.smali uzantılı Dalvik bytecode modifiye edilebilir.

Modifiye edilen Dalvik bytecode tekrar imzalanır,
paketlenir ve cihazda çalıştırılabilir.

Baksmali aracı ile disassemble işlemi yapılabilir.

Alınabilecek Önlemler / Anti Reversing

Kod Karmaşıklaştırma - Obfuscation



Progaurd



Dexgaurd



Java
obfuscators

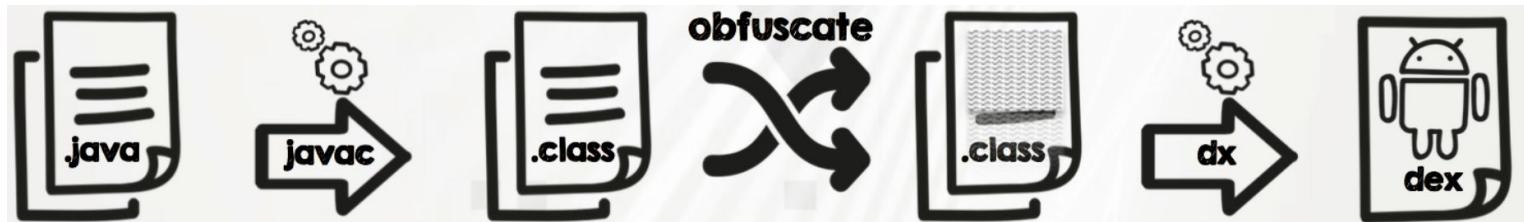
Alınabilecek Önlemler / Anti Reversing

Kod Karmaşıklaştırma - Obfuscation

- Kullanılmayan sınıflar, metodlar temizlenir.
- Bytecode optimize edilir.
- Kullanılmayan instructionlar temizlenir.
- Geriye kalan sınıflar, metodlar, alanlar ve değişkenler anlamsız kısa isimlerle adlandırılır.
- **ProGuard, DexGuard** ile obfuscation önlemi alınabilir.
- **ALLATORI** ile analistleri çıldırtacak obfuscation önlemleri alınabilir.

Alınabilecek Önlemler / Anti Reversing

Kod Karmaşıklaştırma - Obfuscation



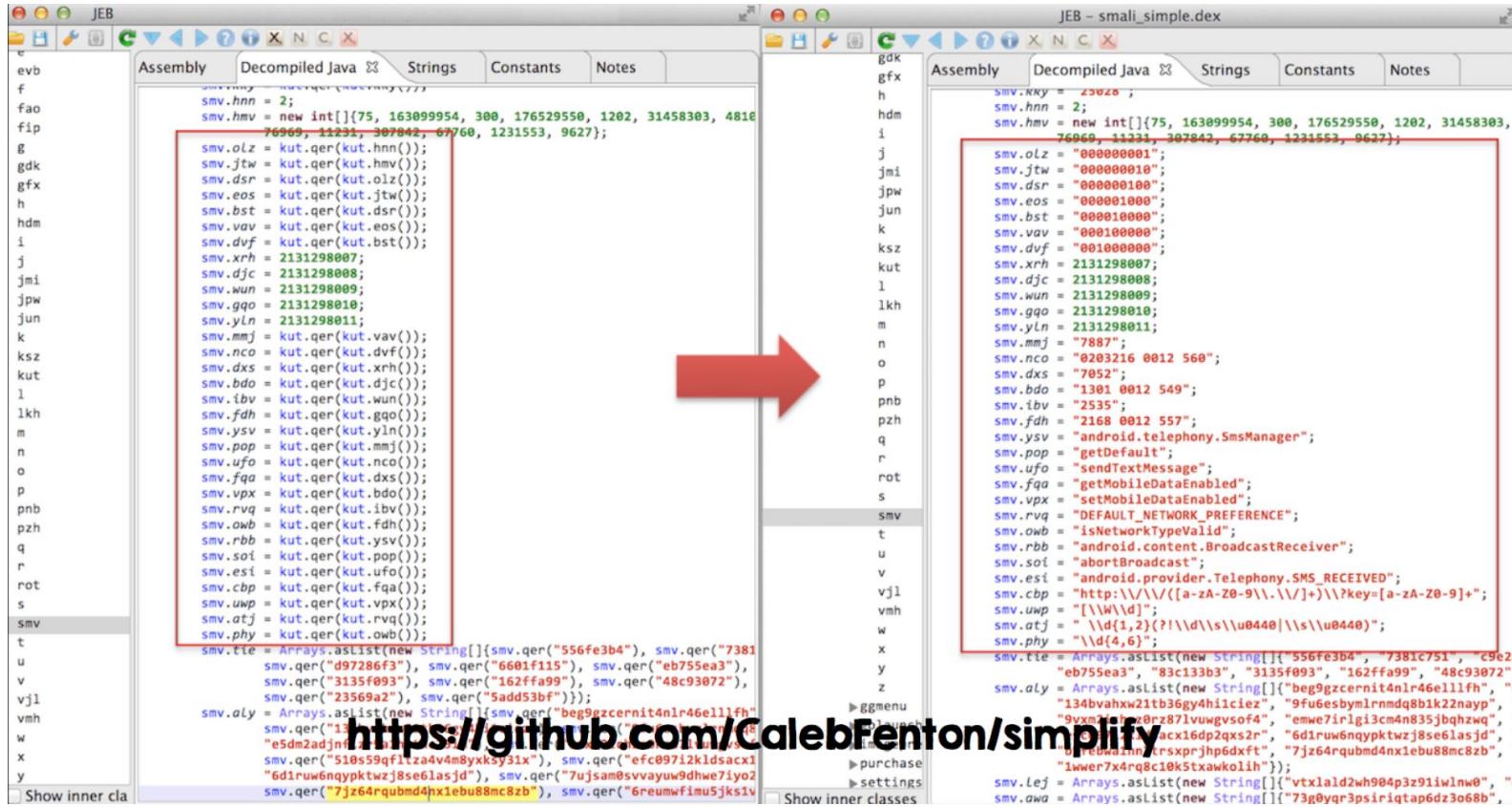
Alınabilecek Önlemler / Anti Reversing

Kod Karmaşıklaştırma - Obfuscation

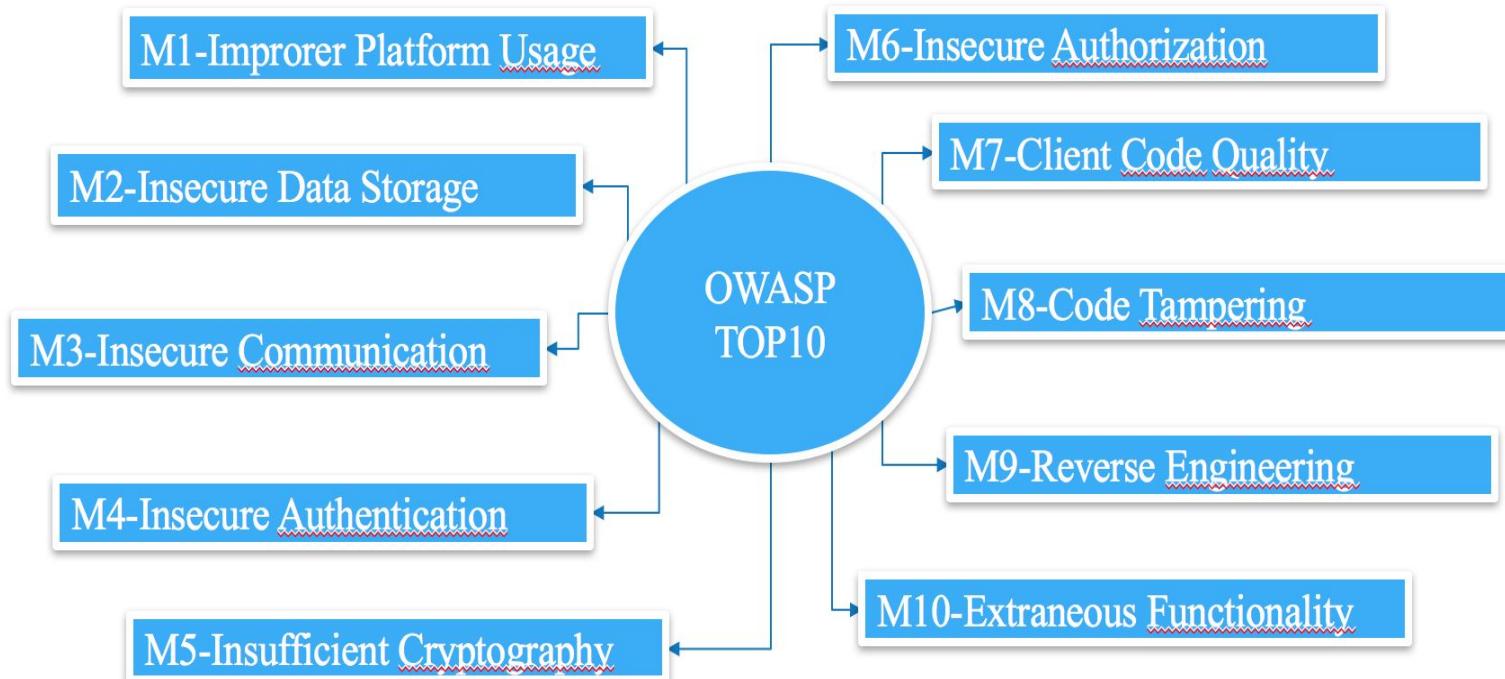
```
public class MyVehicleClass{  
  
    private Motor      myMotor;  
    private Tekerlek   myTekerler;  
    private int        vitesSayisi;  
  
    public int suratHesapla(int sure){  
  
        ...  
  
        return sonSurat;  
    }  
}
```

```
public class A{  
  
    private B      a;  
    private C      b;  
    private int    c;  
  
    public int a(int a){  
  
        ...  
  
        return c;  
    }  
}
```

Deobfuscation - Simplify



OWASP MOBILE TOP 10



https://www.owasp.org/index.php/Mobile_Top_10_2016-Top_10

Android Sızma Testi Ortam Kurulumu

Genymotion : <https://www.genymotion.com/fun-zone/>

Xposed Installer: <https://repo.xposed.info/module/de.robv.android.xposed.installer>

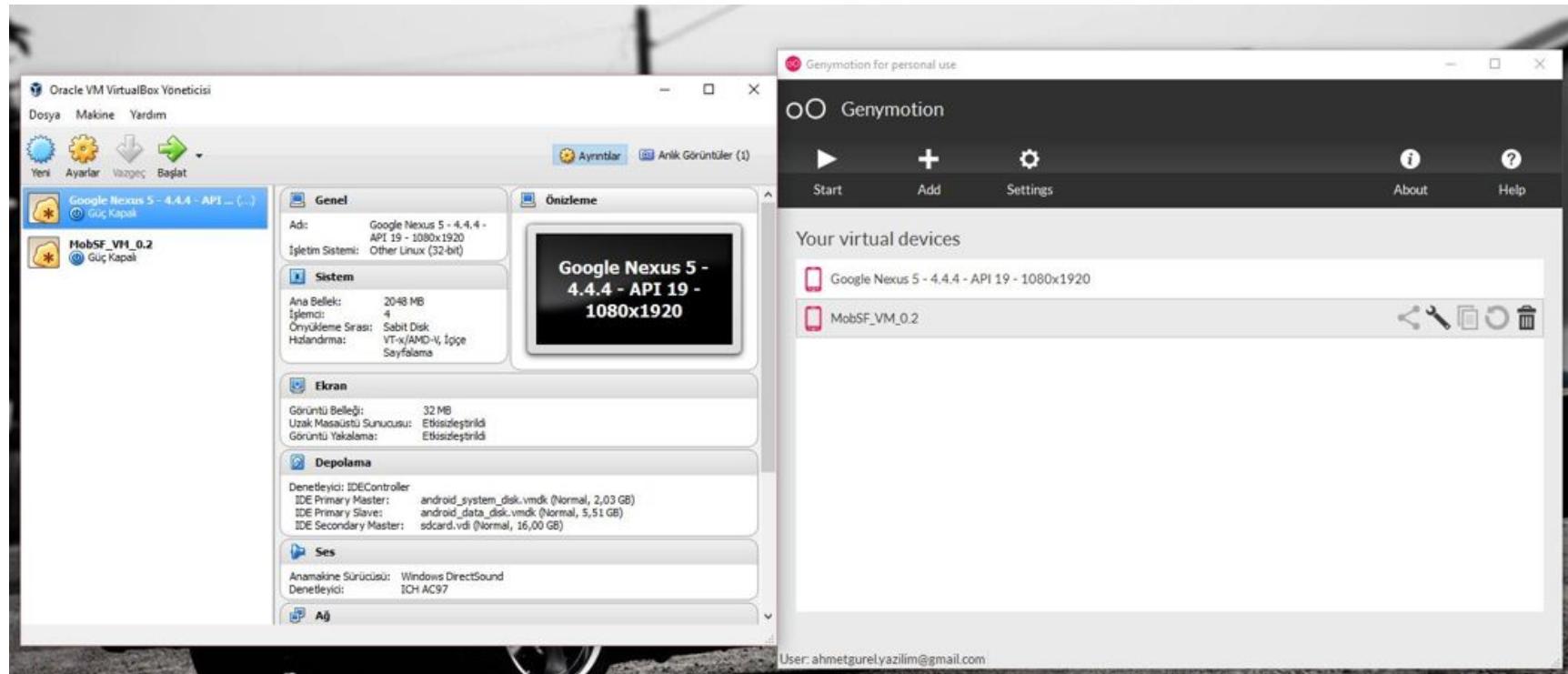
RootCloak : <https://repo.xposed.info/module/com.devadvance.rootcloak2>

JustTrustMe: <https://github.com/Fuzion24/JustTrustMe/releases>

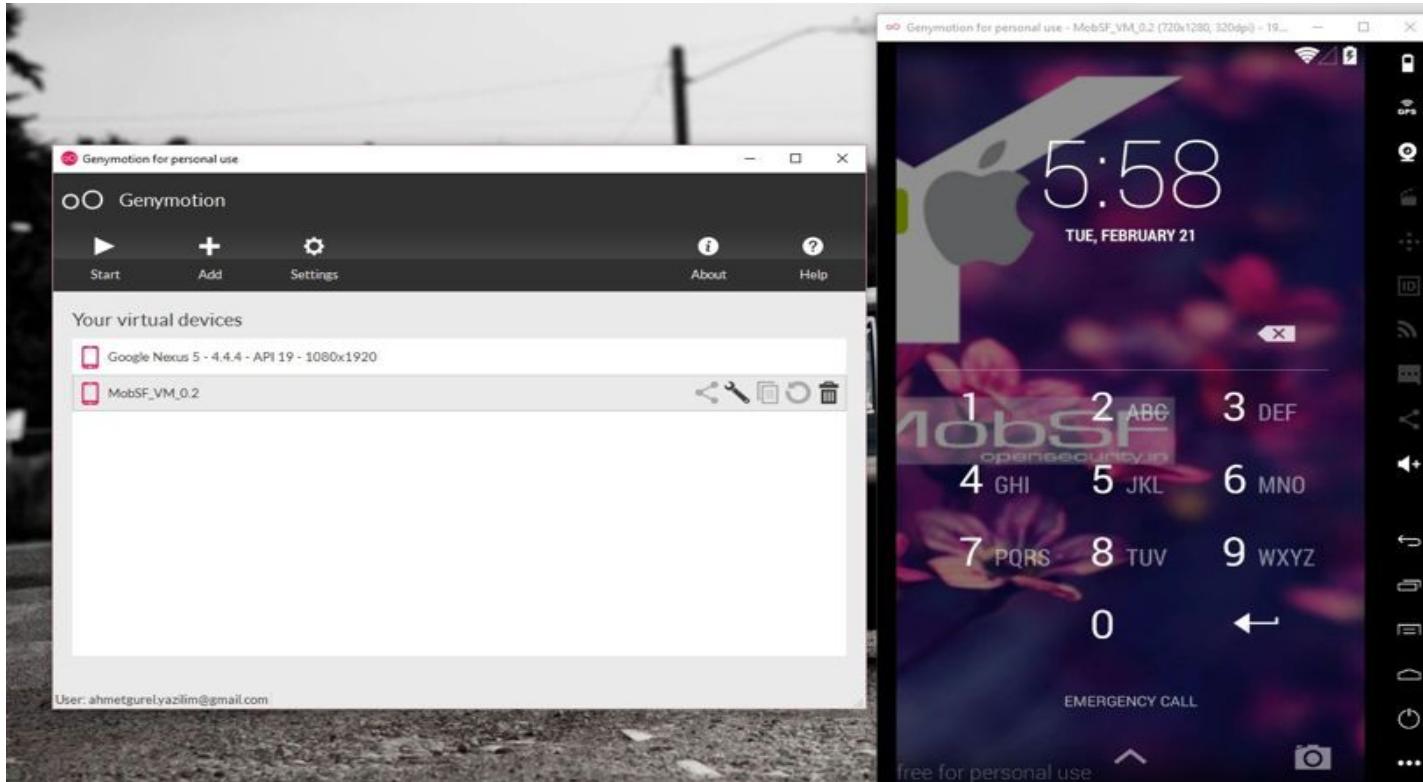
SSLUnpinning - Certificate Pinning Bypass : <https://repo.xposed.info/module/mobi.acpm.sslunpinning>

Inspeckage : <https://github.com/ac-pm/Inspeckage/releases/tag/v2.4>

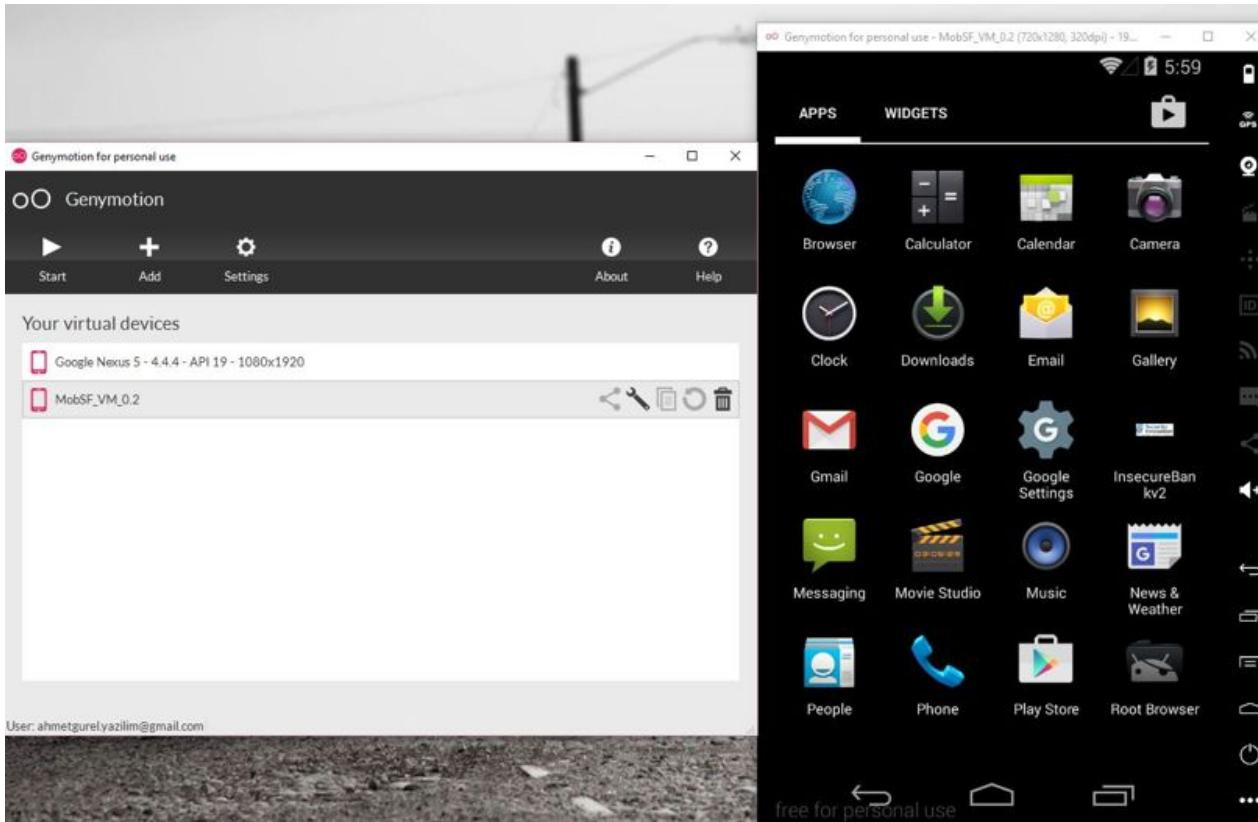
Ortam Kurulumu



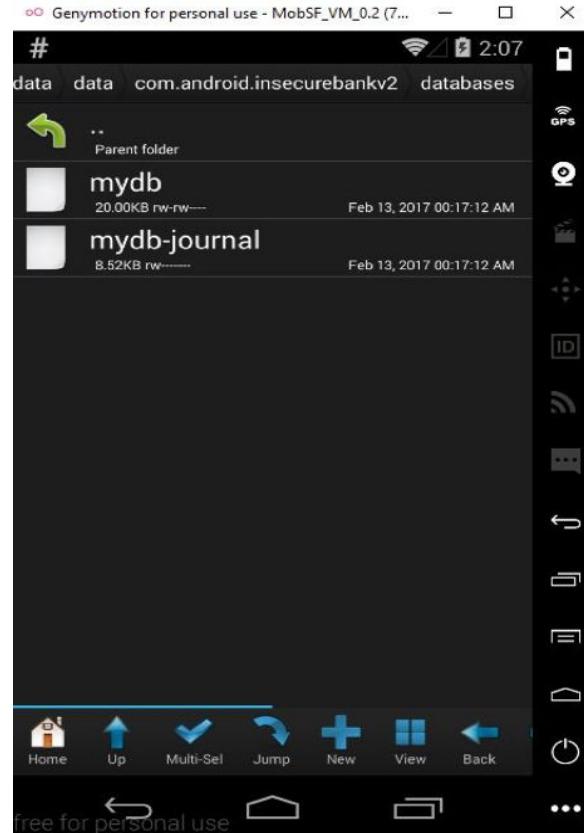
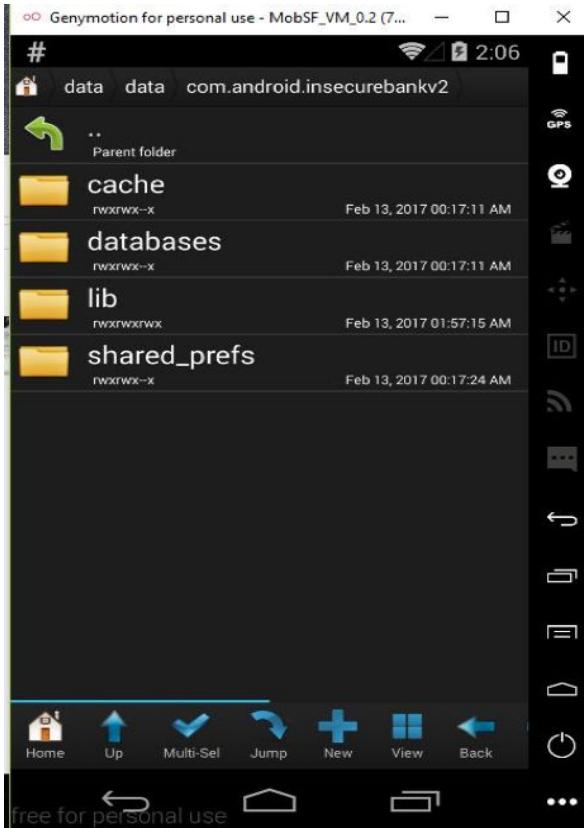
Ortam Kurulumu



Ortam Kurulumu



Android Uygulama Dosyaları



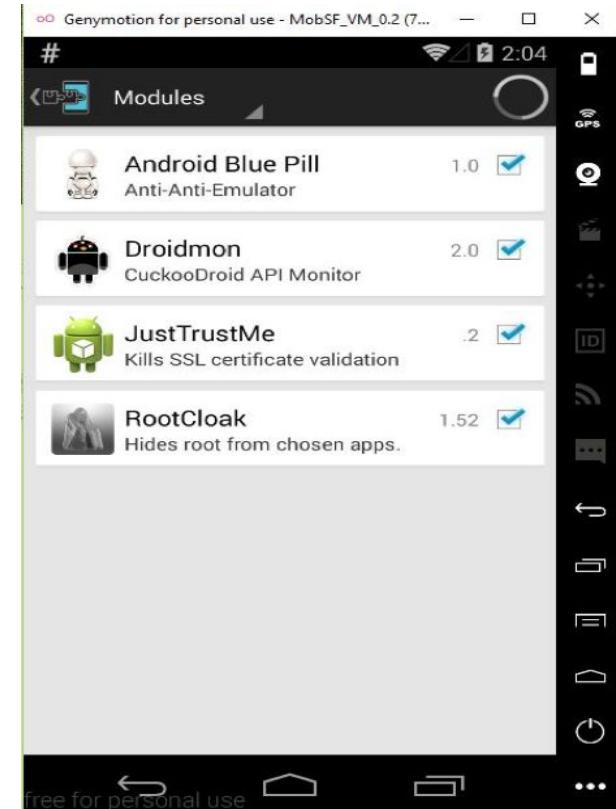
Xposed Modülleri

Xposed Modülleri Android cihazda uygulamaları özelleştirmek değiştirmek için kullanılır.

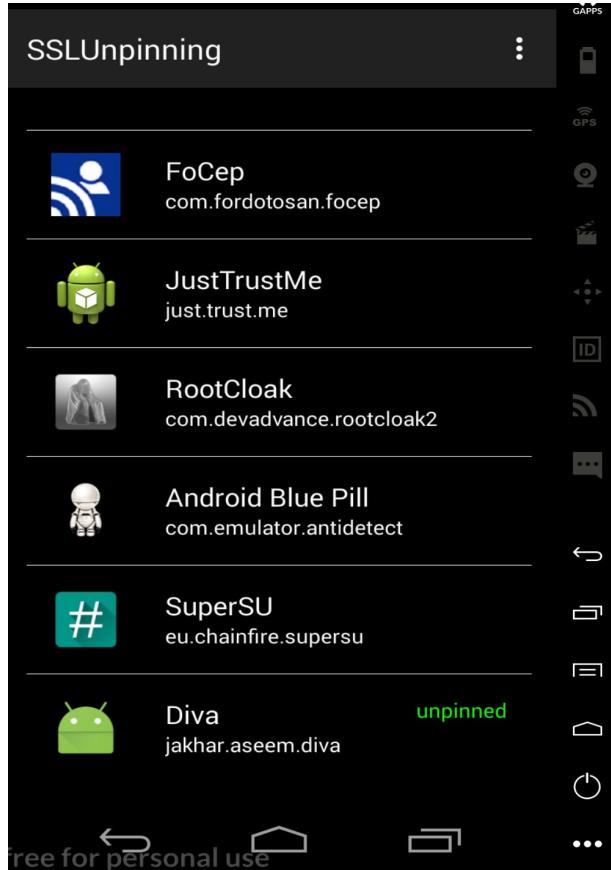
Uygulama geliştirilirken yazılan kontrollerin, izinlerin değiştirilmesine imkan verebiliyor.

Mesela yukarıda gördüğümüz RootCloak modülü bir uygulama cihaz root lumu diye kontrol edip, çalışmayıorsa bu kontrolü engellemeye/atlatmaya yaramaktadır ve güvenlik testleri için önemli bir yer tutmaktadır.

Bunun gibi birçok modül bulunmaktadır.

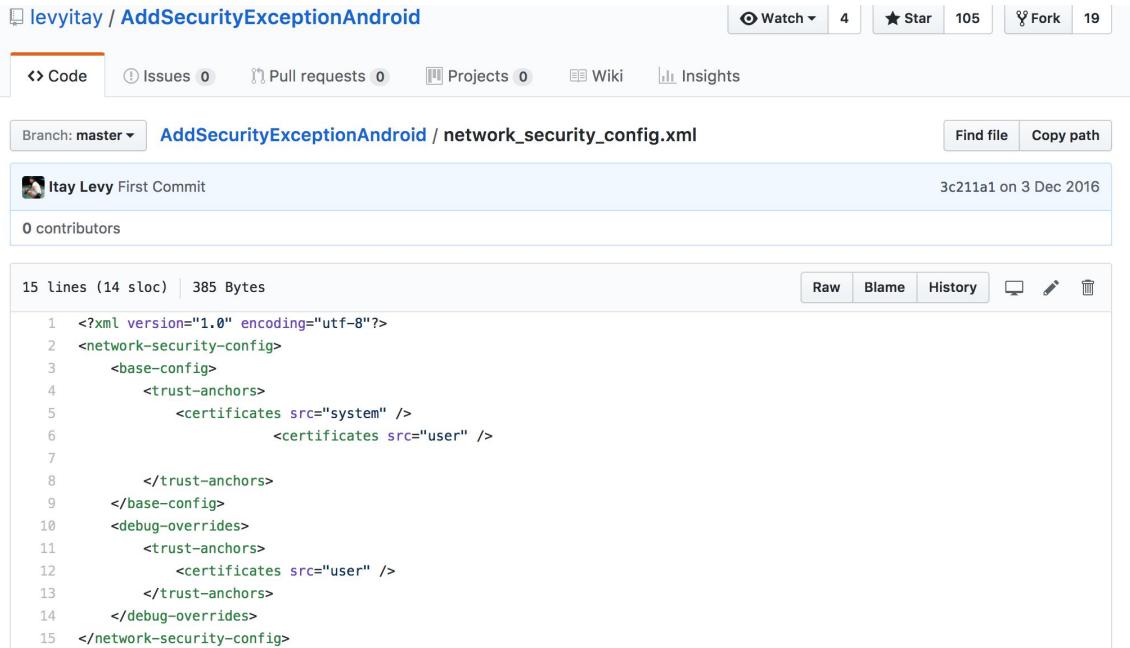


SSL Pinning Bypass



SSL Pinning Bypass

Android 7.0'da, Google, kullanıcıların Sertifika Yetkililerine (CA) güvenme biçiminde değişiklikler getirdi. Bu değişiklikler, üçüncü şahısların uygulamadan gelen ağ isteklerini dinlemelerini engeller:



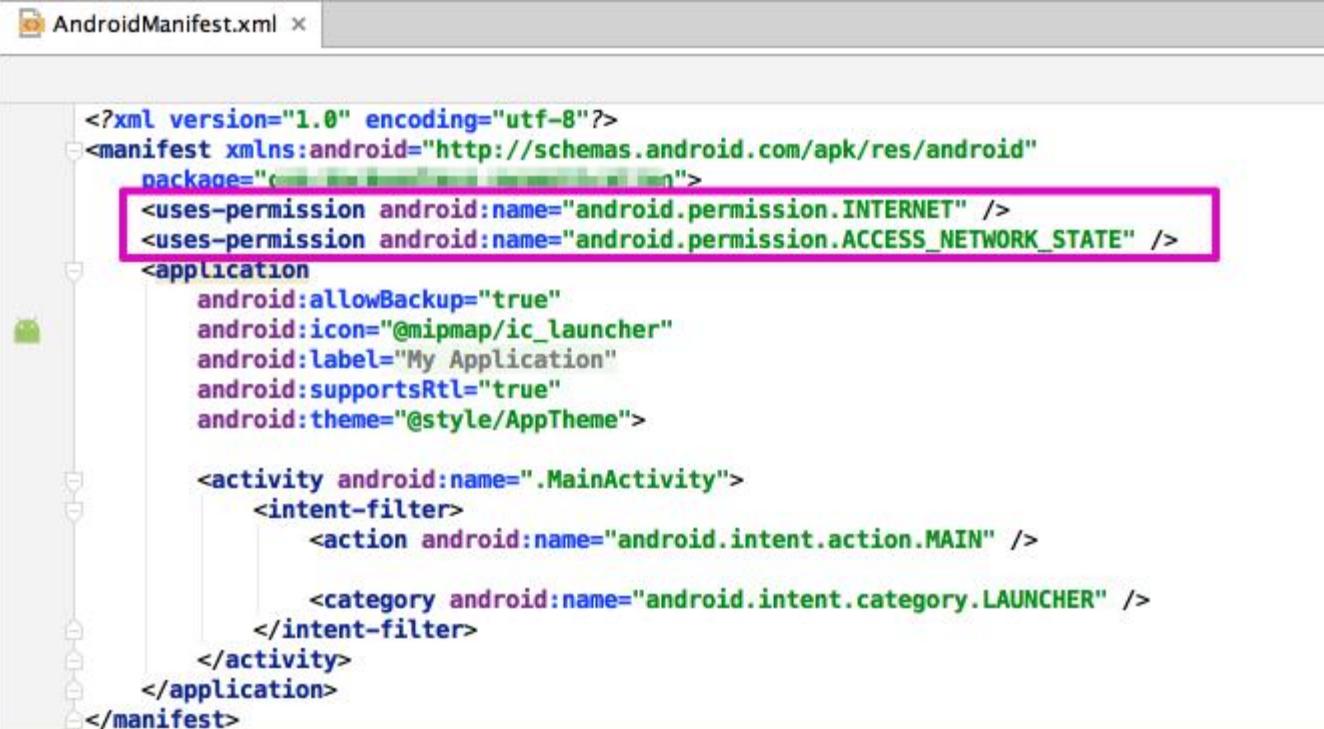
The screenshot shows a GitHub repository page for 'levityay / AddSecurityExceptionAndroid'. The repository has 4 issues, 105 forks, and 19 stars. The 'Code' tab is selected, showing the 'network_security_config.xml' file. The file was committed by Itay Levy on 3 Dec 2016. The code defines a network security configuration with trust anchors for system and user certificates.

```
<?xml version="1.0" encoding="utf-8"?>
<network-security-config>
    <base-config>
        <trust-anchors>
            <certificates src="system" />
            <certificates src="user" />
        </trust-anchors>
    </base-config>
    <debug-overrides>
        <trust-anchors>
            <certificates src="user" />
        </trust-anchors>
    </debug-overrides>
</network-security-config>
```

Android Uygulama İzinleri

Value	Meaning
<code>"normal"</code>	The default value. A lower-risk permission that gives requesting applications access to isolated application-level features, with minimal risk to other applications, the system, or the user. The system automatically grants this type of permission to a requesting application at installation, without asking for the user's explicit approval (though the user always has the option to review these permissions before installing).
<code>"dangerous"</code>	A higher-risk permission that would give a requesting application access to private user data or control over the device that can negatively impact the user. Because this type of permission introduces potential risk, the system may not automatically grant it to the requesting application. For example, any dangerous permissions requested by an application may be displayed to the user and require confirmation before proceeding, or some other approach may be taken to avoid the user automatically allowing the use of such facilities.
<code>"signature"</code>	A permission that the system grants only if the requesting application is signed with the same certificate as the application that declared the permission. If the certificates match, the system automatically grants the permission without notifying the user or asking for the user's explicit approval.
<code>"signatureOrSystem"</code>	A permission that the system grants only to applications that are in the Android system image or that are signed with the same certificate as the application that declared the permission. Please avoid using this option, as the <code>signature</code> protection level should be sufficient for most needs and works regardless of exactly where applications are installed. The <code>"signatureOrSystem"</code> permission is used for certain special situations where multiple vendors have applications built into a system image and need to share specific features explicitly because they are being built together.

Android Uygulama İzinleri



The screenshot shows the AndroidManifest.xml file in an IDE. Two permission declarations are highlighted with a pink rectangle: `<uses-permission android:name="android.permission.INTERNET" />` and `<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />`. The XML code is as follows:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.myapplication">
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="My Application"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```

Mobil Sızma Testi Araçları

ADB (Android Debug Bridge)

"Android Debug Bridge (Android Ayıklama Köprüsü - adb), bir emulator kopyasının veya Android'li bir cihazının durumunu yönetmeye izin veren çok yönlü bir araç.

Mobil Sızma Testi Araçları

ADB (Android Debug Bridge)

- **ADB push** : Cihaza dosya gönderir. **adb push uygulama.apk /sdcard/uygulama.apk**
- **ADB pull** : Cihazdan dosya alır. **adb pull /system/app/uygulama.apk**
- **ADB install** : Cihaza uygulama yükler. **adb install uygulama.apk**
- **ADB shell** : Cihazın komut satırına (terminal) bağlanır. **adb shell**

Mobil Sızma Testi Araçları

ADB (Android Debug Bridge)

```
C:\$ adb.exe devices
List of devices attached
192.168.169.101:5555    device

C:\$ adb.exe install canyoupwnme.apk
3254 KB/s (945721 bytes in 2.837s)
WARNING: linker: libdvm.so has text relocations. This is wasting memory and is a security risk. Please fix.
WARNING: linker: app_process has text relocations. This is wasting memory and is a security risk. Please fix.
      pkg: /data/local/tmp/canyoupwnme.apk
Success

C:\$ adb.exe pull /data/data/com.android.insecurebankv2/databases/mydb
723 KB/s (20480 bytes in 0.027s)

C:\$ adb.exe shell
root@mobsec:/ # pwd
pwd
/
root@mobsec:/ # cd data
cd data
root@mobsec:/data # ls
ls
anr
app
app-asec
app-lib
app-private
backup
```

Mobil Sızma Testi Araçları

Andro Guard

AndroGuard python ile geliştirilen statik kod analizi yapan bir araçtır.

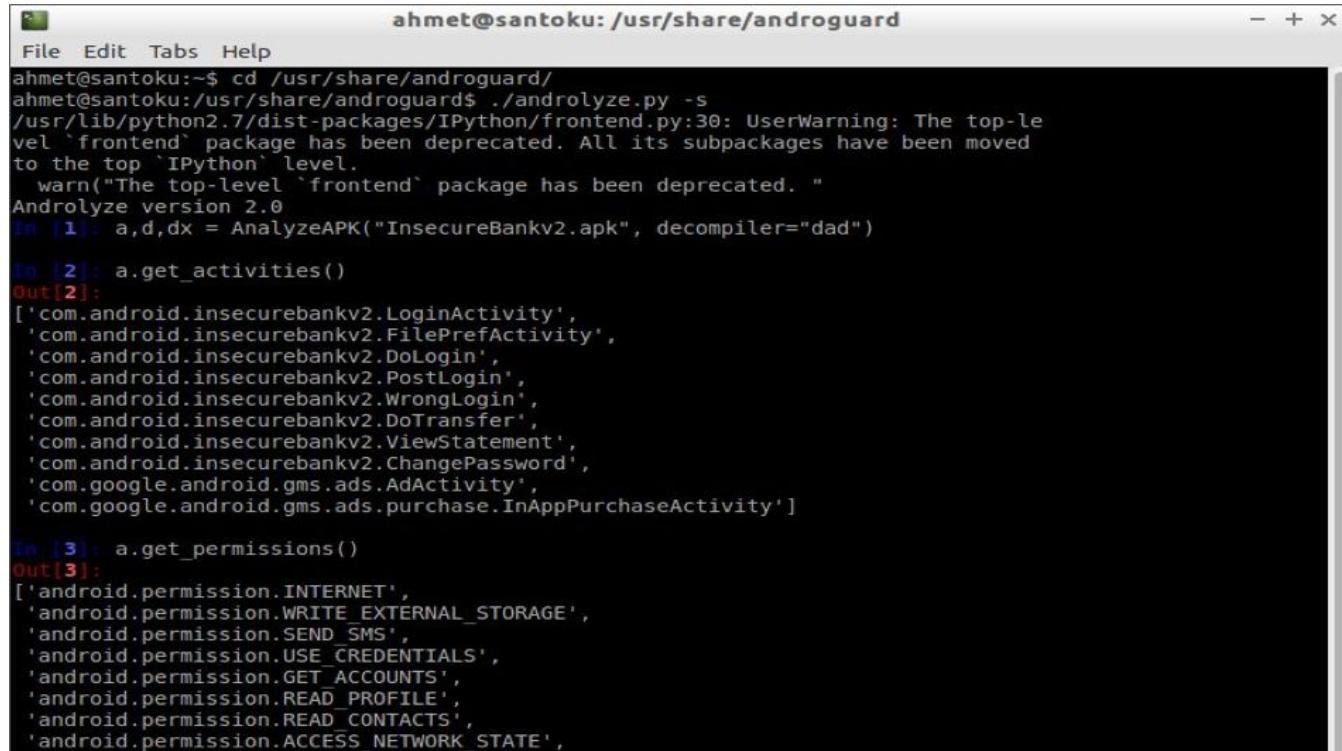
San Toku'nun içinde kurulu olarak gelmektedir.

AndroGuard Santoku üzerinde bu şekilde çalıştırılmaktadır. İlk olarak kurulu olduğu dizine gittik ve apk dosyamızı da oraya taşıdık. `./androlyze.py -s` ile çalıştık. İlk satırımıza

`a,d,dx= AnalyzeAPK("Insecurebankv2.apk", decompiler="dad")` komutunu yazarak apk dosyamızı göstererek decompile ediyoruz. Daha sonra programın parametreleri ile birçok analiz edebilmektedir. Resimde uygulamanın activitylerini ve izinlerini getirdik.

Mobil Sızma Testi Araçları

Andro Guard



The screenshot shows a terminal window titled "ahmet@santoku: /usr/share/androguard". The terminal is running Python code to analyze the APK "InsecureBankv2.apk" using the "androlyze.py" script. The output shows the following:

```
ahmet@santoku:~$ cd /usr/share/androguard/
ahmet@santoku:/usr/share/androguard$ ./androlyze.py -s
/usr/lib/python2.7/dist-packages/IPython/frontend.py:30: UserWarning: The top-level `frontend` package has been deprecated. All its subpackages have been moved to the top `IPython` level.
  warn("The top-level `frontend` package has been deprecated. "
Androlyze version 2.0
In [1]: a,d,dx = AnalyzeAPK("InsecureBankv2.apk", decompiler="dad")

In [2]: a.get_activities()
Out[2]:
['com.android.insecurebankv2.LoginActivity',
 'com.android.insecurebankv2.FilePrefActivity',
 'com.android.insecurebankv2.DoLogin',
 'com.android.insecurebankv2.PostLogin',
 'com.android.insecurebankv2.WrongLogin',
 'com.android.insecurebankv2.DoTransfer',
 'com.android.insecurebankv2.ViewStatement',
 'com.android.insecurebankv2.ChangePassword',
 'com.google.android.gms.ads.AdActivity',
 'com.google.android.gms.ads.purchase.InAppPurchaseActivity']

In [3]: a.get_permissions()
Out[3]:
['android.permission.INTERNET',
 'android.permission.WRITE_EXTERNAL_STORAGE',
 'android.permission.SEND_SMS',
 'android.permission.USE_CREDENTIALS',
 'android.permission.GET_ACCOUNTS',
 'android.permission.READ_PROFILE',
 'android.permission.READ_CONTACTS',
 'android.permission.ACCESS_NETWORK_STATE',
```

Mobil Sızma Testi Araçları

Burp Suite

Burp gelişmiş bir proxy yazılımıdır.

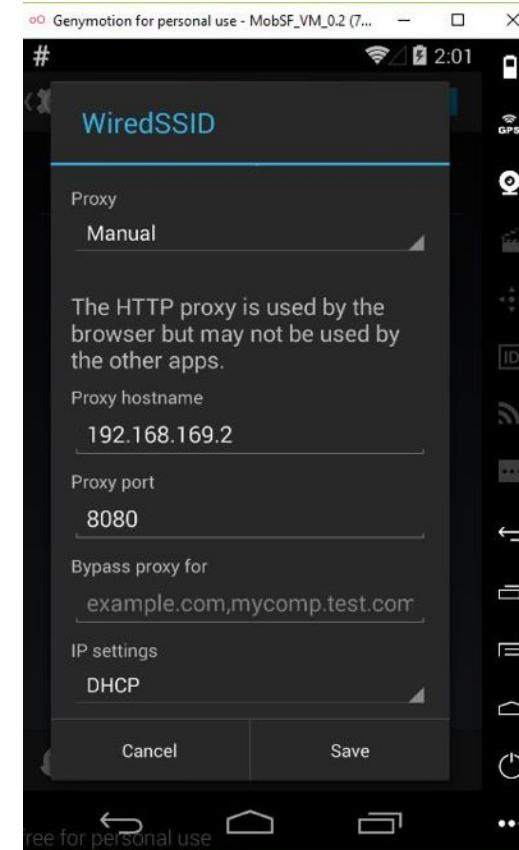
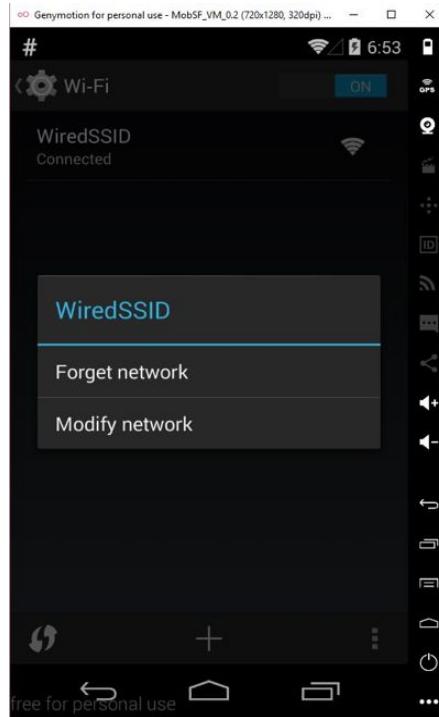
Bunun dışında birçok teste yardımcı olmakta ve imkan tanıtmaktadır.

Web Testlerinin olmazsa olmazı Burp Mobil testlerimizde de o kadar önemli.

Şimdi Burp Suite Emulatorümüzden bağlanmayı bakalım beraber.

Mobil Sızma Testi Araçları

Burp Suite



Mobil Sızma Testi Araçları

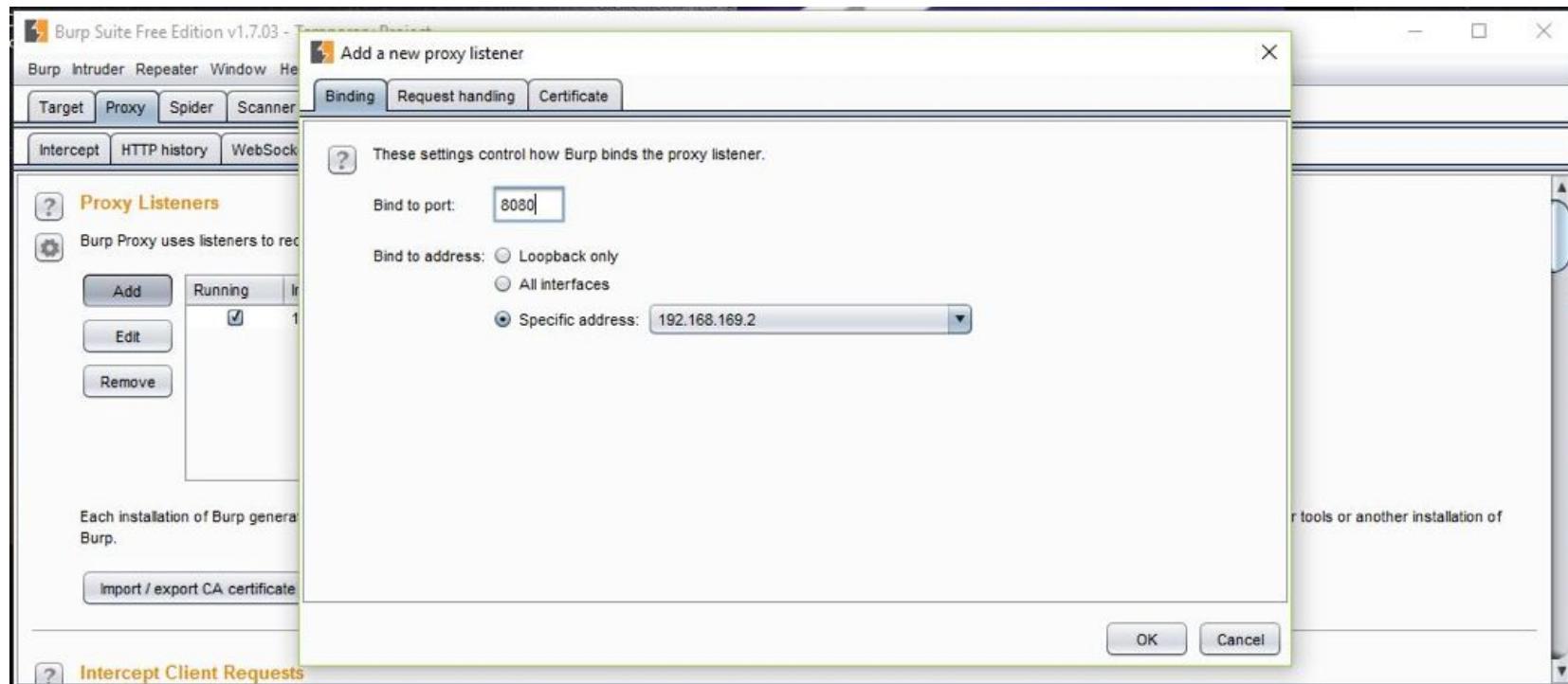
Burp Suite

Ayarlara (Settings) e girerek daha sonra Wi-Fi ye tıklayarak WiredSSID nin üzerine basılı tutarak Modify network diyerek Proxy belirliyoruz.

Burada IP adresi test yaptığınız makinenin IPsidir kendi ana makineniz ya da yukarıda bahsettiğim makineleri indirdiyseniz onun IP adresidir.

Mobil Sızma Testi Araçları

Burp Suite



Mobil Sızma Testi Araçları

Burp Suite



Mobil Sızma Testi Araçları

Burp Suite

Request

Raw Params Headers Hex

```
POST [REDACTED]
Host: t[REDACTED]
ActivationId: [REDACTED]
Cookie: [REDACTED]
Accept: */*
Connection: close
Channel: Mobile
AuthKey: [REDACTED]
Accept-Language: tr-TR;q=1.0, en-TR;q=0.9, en;q=0.8
Accept-Encoding: gzip, deflate
ApplicationVersion: 5.0.45
Content-Type: application/json
DeviceId: [REDACTED]
Language: tr
SessionToken: [REDACTED]
User-Agent: [REDACTED]
DeviceModel: iOS|Apple|iPhone8,4|10.3.3
IpAddress: 223
Content-Length: 223

{"PersonalNote":{"HasSms":false,"Status":"Undefined","EmailAddress": "[REDACTED]", "EmailDomain": "[REDACTED]","PeriodType":"Once","HasEmail":true,"HasPush":true,"Date":"2018-01-22T00:10:21","Note":"<svg onload=prompt(2)>\n\n","Id": -1}}
```

?

<

+

>

Type a search term

0 matches

Response

Raw Headers Hex

```
HTTP/1.1 200 OK
Cache-Control: no-cache
Pragma: no-cache
Content-Type: application/json; charset=utf-8
Expires: -1
X-Powered-By: ASP.NET
Date: Mon, 22 Jan 2018 07:20:54 GMT
Connection: close
Content-Length: 137
Set-Cookie: [REDACTED]

{"ResponseBody":{},"Header":{"ResponseStatus":true,"ResponseCode":"0","ResponseMessage":"Başarılı","ResponseDetailMessage":"Success"}}
```

?

<

+

>

Type a search term

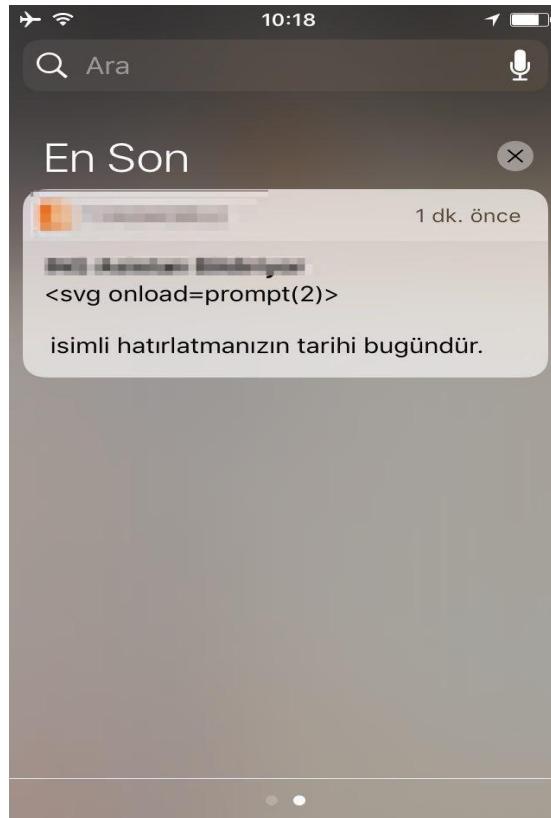
0 matches

Done

498 bytes | 11,930 millis

Mobil Sızma Testi Araçları

Burp Suite



Mobil Sızma Testi Araçları

Sqlite Veritabanı incelemeye Sqlite Browser ve Sqlite3 Kullanımı

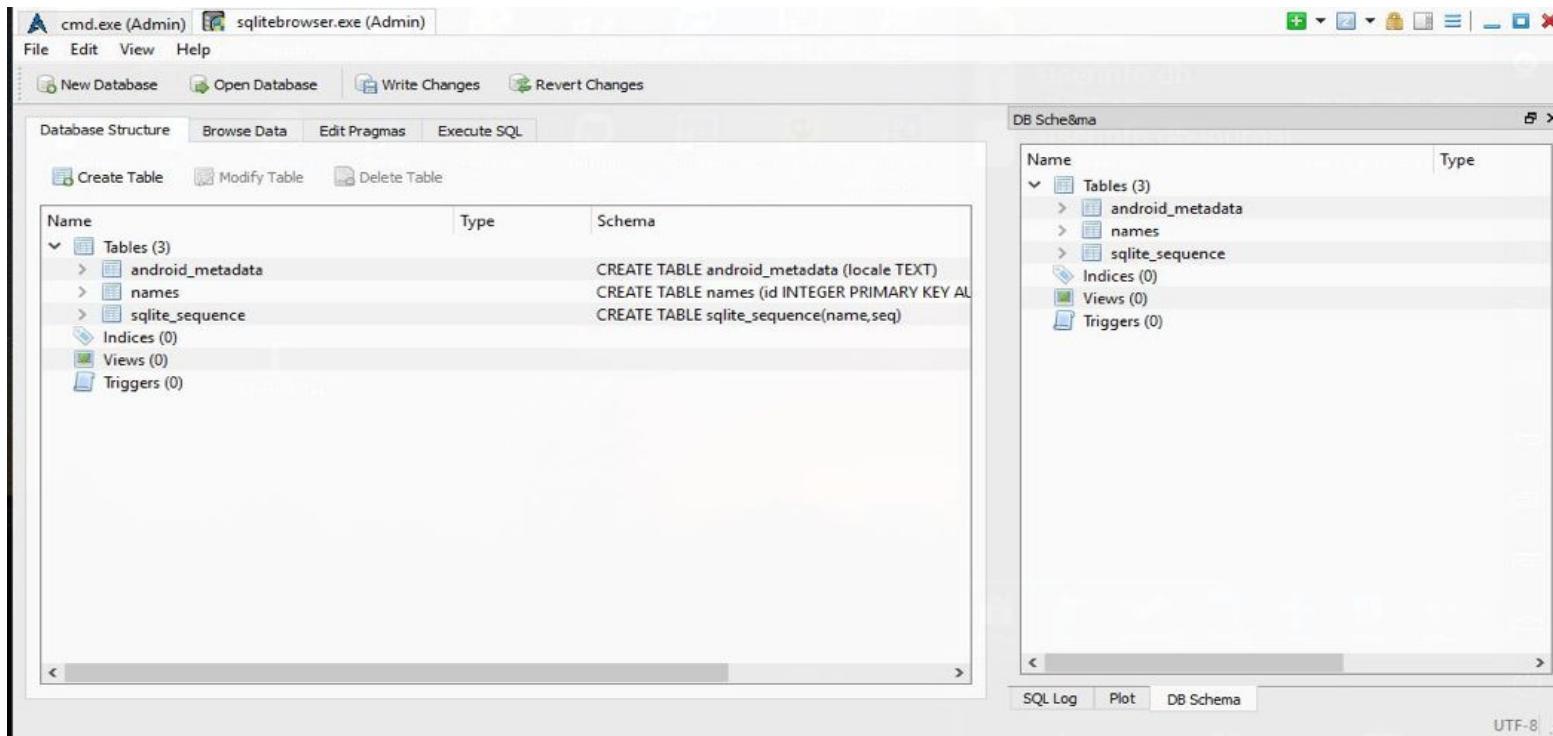
Uygulamayı cihazımıza aktardıktan sonra eğer dosyalar cihazda tutuluyorsa veritabanı dosyalarını ADB ile kendi bilgisayarımıza indirebiliriz.

Bu database dosyalarının içeriğini Sqlite Browser ile görüntüleyebiliriz.

Bunun dışında da Sqlite3 ile veritabanını seçerek sorgular yazıp bununla da görüntüleyebilmekteyiz.

Mobil Sızma Testi Araçları

Sqlite Veritabanı incelemeye Sqlite Browser ve Sqlite3 Kullanımı



Mobil Sızma Testi Araçları

Sqlite Veritabanı incelemeye Sqlite Browser ve Sqlite3 Kullanımı

```
C:\$ sqlite3.exe my.db
SQLite version 3.8.4.3 2014-04-03 16:53:12
Enter ".help" for usage hints.
sqlite> .tables
android_metadata  names
sqlite> select * from names;
sqlite> |
```

Mobil Sızma Testi Araçları

AndroBugs Framework

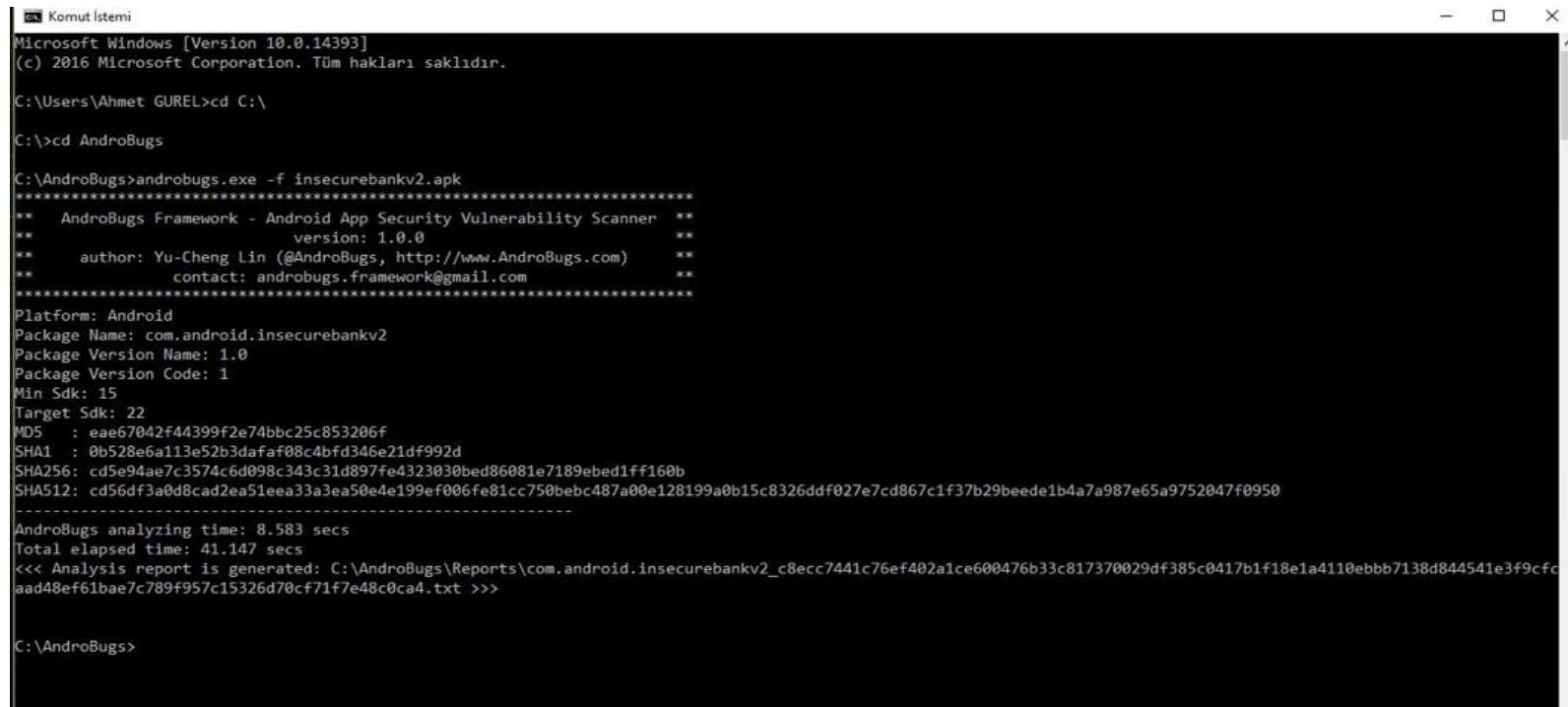
AndroBugs Framework, Android uygulamalarda güvenlik testi gerçekleştiren frameworklerden bir tanesidir.

Kullanımı oldukça basittir. Konsol üzerinden biz **androbugs -f apk_dosyasi** şeklinde kullanarak frameworkumuzu çalıştırıldık.

Bunun sonucunda kendi klasörünün altında Reports klasörünün altında detaylı rapor oluşturmaktadır.

Mobil Sızma Testi Araçları

AndroBugs Framework



```
Komut İstemci
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. Tüm hakları saklıdır.

C:\Users\Ahmet GUREL>cd C:\

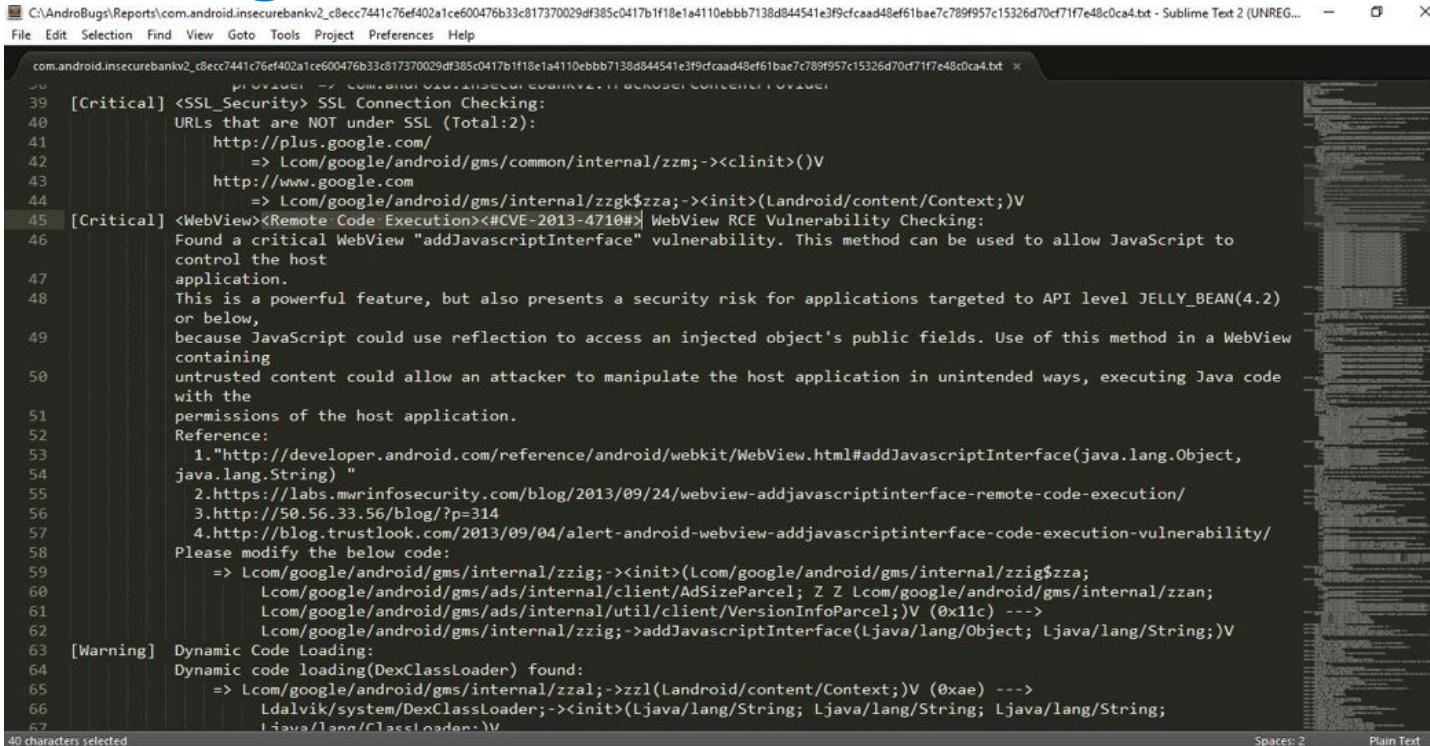
C:\>cd AndroBugs

C:\AndroBugs>androbugs.exe -f insecurebankv2.apk
*****
**  AndroBugs Framework - Android App Security Vulnerability Scanner  **
**          version: 1.0.0                                         **
**      author: Yu-Cheng Lin (@AndroBugs, http://www.AndroBugs.com)   **
**      contact: androbugs.framework@gmail.com                      **
*****
Platform: Android
Package Name: com.android.insecurebankv2
Package Version Name: 1.0
Package Version Code: 1
Min Sdk: 15
Target Sdk: 22
MD5 : eae67042f44399f2e74bbc25c853206f
SHA1 : 0b528e6a113e52b3dafaf08c4bfd346e21df992d
SHA256: cd5e94ae7c3574c6d098c343c31d897fe4323030bed86081e7189ebed1ff160b
SHA512: cd56df3a0d8cad2ea51eeaa33a3ea50e4e199ef006fe81cc750bebc487a00e128199a0b15c8326ddf027e7cd867c1f37b29beede1b4a7a987e65a9752047f0950
-----
AndroBugs analyzing time: 8.583 secs
Total elapsed time: 41.147 secs
<<< Analysis report is generated: C:\AndroBugs\Reports\com.android.insecurebankv2_c8ecc7441c76ef402a1ce600476b33c817370029df385c0417b1f18e1a4110ebbb7138d844541e3f9fc
aad48ef61bae7c789f957c15326d70cf71f7e48c0ca4.txt >>>

C:\AndroBugs>
```

Mobil Sızma Testi Araçları

AndroBugs Framework



The screenshot shows a Sublime Text 2 window displaying the results of an Android security audit. The file path is C:\AndroBugs\Reports\com.android.insecurebankv2_c8ecc7441c76ef402a1ce600476b33c817370029df385c0417b1f10e1a4110ebbb7138d844541e3f9fcfad48ef61bae7c789f957c15326d70cf71f7e48c0ca4.txt. The content of the file is as follows:

```
com.android.insecurebankv2_c8ecc7441c76ef402a1ce600476b33c817370029df385c0417b1f10e1a4110ebbb7138d844541e3f9fcfad48ef61bae7c789f957c15326d70cf71f7e48c0ca4.txt - Sublime Text 2 (UNREG... - □ ×
```

File Edit Selection Find View Goto Tools Project Preferences Help

```
39 [Critical] <SSL_Security> SSL Connection Checking:
40 URLs that are NOT under SSL (Total:2):
41     http://plus.google.com/
42         => Lcom/google/android/gms/common/internal/zzm;-><clinit>()V
43     http://www.google.com
44         => Lcom/google/android/gms/internal/zzgk$zza;-><init>(Landroid/content/Context;)V
45 [Critical] <WebView><Remote Code Execution><#CVE-2013-4710#| WebView RCE Vulnerability Checking:
46 Found a critical WebView "addJavascriptInterface" vulnerability. This method can be used to allow JavaScript to
control the host
application.
47 This is a powerful feature, but also presents a security risk for applications targeted to API level JELLY_BEAN(4.2)
or below,
48 because JavaScript could use reflection to access an injected object's public fields. Use of this method in a WebView
containing
49 untrusted content could allow an attacker to manipulate the host application in unintended ways, executing Java code
with the
50 permissions of the host application.
51 Reference:
52     1."http://developer.android.com/reference/android/webkit/WebView.html#addJavascriptInterface(java.lang.Object,
53 java.lang.String)"
54     2.https://labs.mwrinfosecurity.com/blog/2013/09/24/webview-addjavascriptinterface-remote-code-execution/
55     3.http://50.56.33.56/blog/?p=314
56     4.http://blog.trustlook.com/2013/09/04/alert-android-webview-addjavascriptinterface-code-execution-vulnerability/
57 Please modify the below code:
58     => Lcom/google/android/gms/internal/zzig;-><init>(Lcom/google/android/gms/internal/zzig$zza;
59         Lcom/google/android/gms/ads/internal/client/AdSizeParcel; Z Z Lcom/google/android/gms/internal/zzan;
60         Lcom/google/android/gms/ads/internal/util/client/VersionInfoParcel;)V (0x1c) --->
61         Lcom/google/android/gms/internal/zzig;->addJavascriptInterface(Ljava/lang/Object; Ljava/lang/String;)V
62 [Warning] Dynamic Code Loading:
63     Dynamic code loading(DexClassLoader) found:
64         => Lcom/google/android/gms/internal/zzal;->zzi(Landroid/content/Context;)V (0xae) --->
65             Ldalvik/system/DexClassLoader;-><init>(Ljava/lang/String; Ljava/lang/String; Ljava/lang/String;
66             Ljava/lang/ClassLoader;)V
67 40 characters selected
```

Mobil Sızma Testi Araçları

Mobile Security Framework (MobSF)

AndroBugs gibi MobSF de mobil uygulama analizi yapan bir frameworktur.

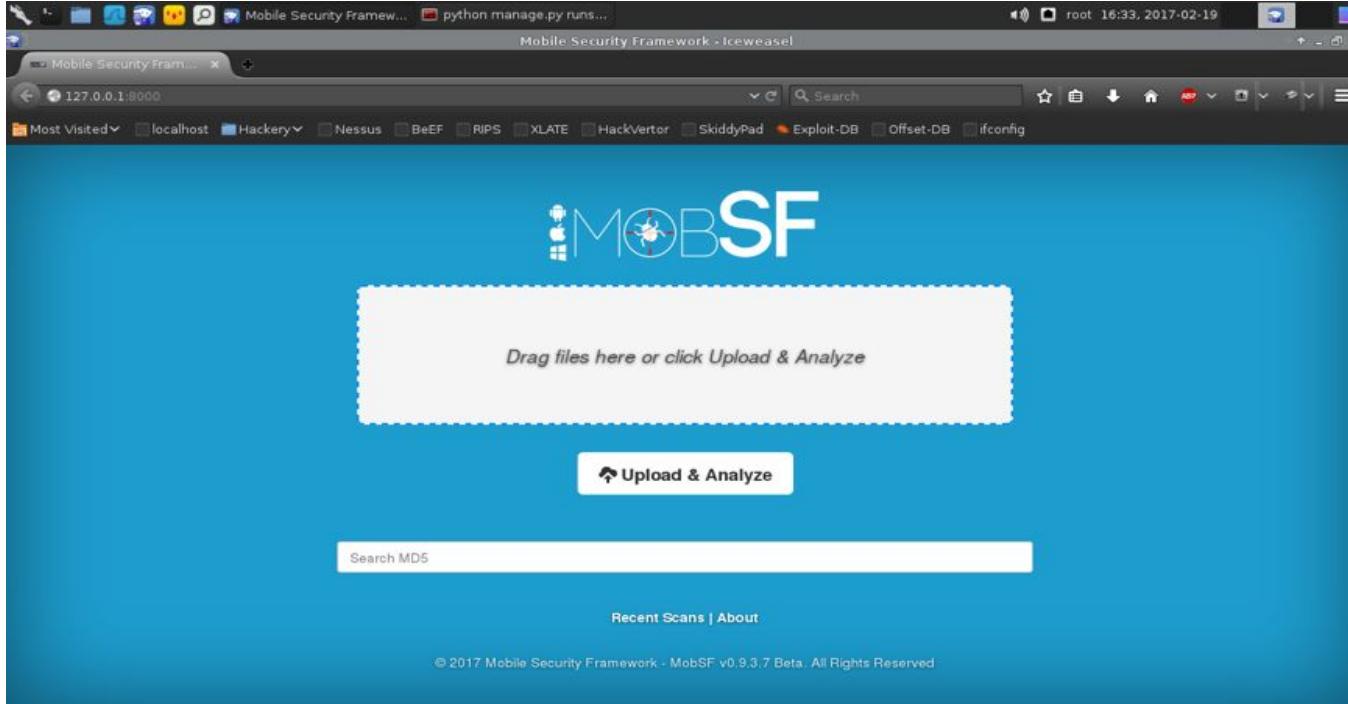
Şu an en kullanışlı ve sağlam araç denebilir. Oldukça popüler ve güzel bir araçtır.

MobSF i indirdikten sonra Windows, Linux ve OSX e kurabilirsiniz. Kurduktan sonra localhostunuzda tarayıcıda çalışmakta ve apk dosyasını seçerek direk çalışmakta. Oldukça basit bir kullanımı vardır.

Adres olarak **127.0.0.1:8000** adresinde çalışmaktadır.

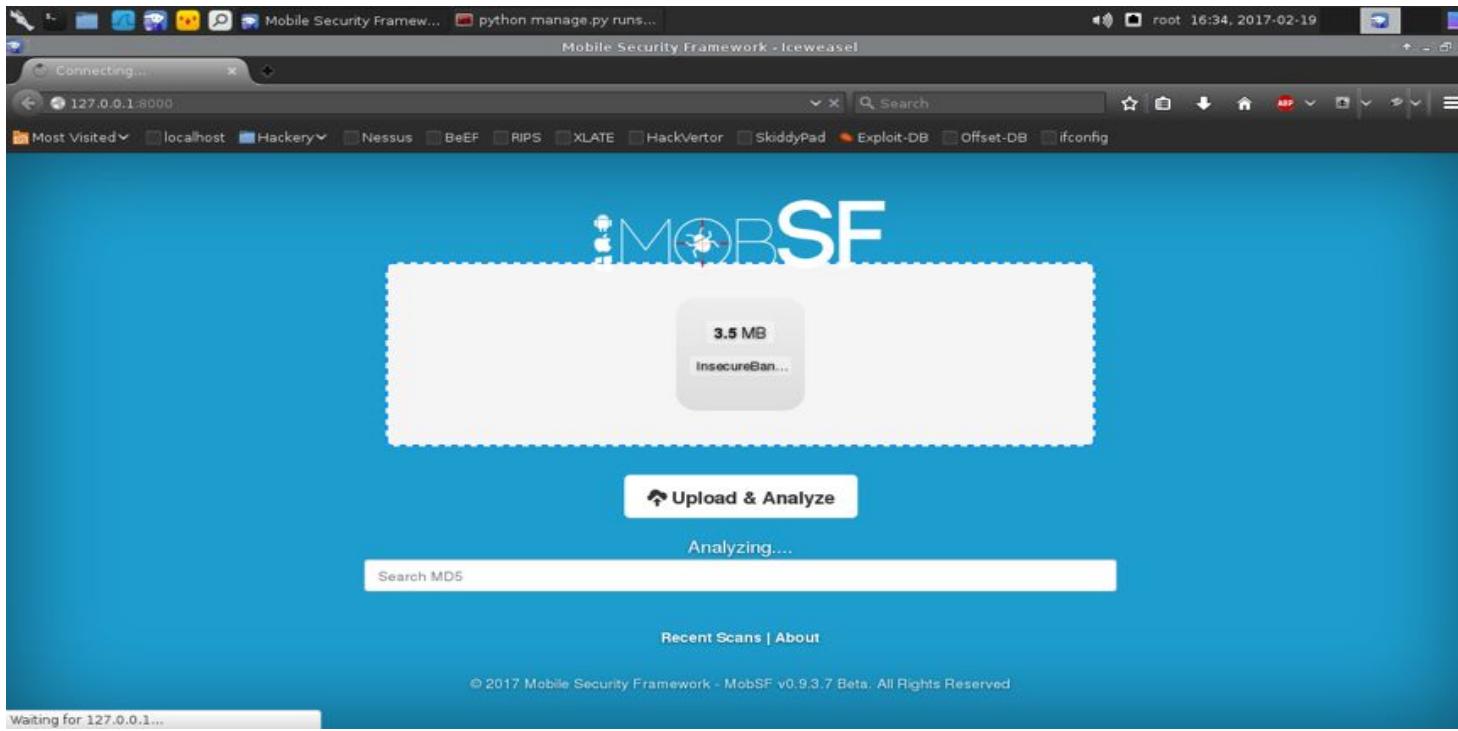
Mobil Sızma Testi Araçları

Mobile Security Framework (MobSF)



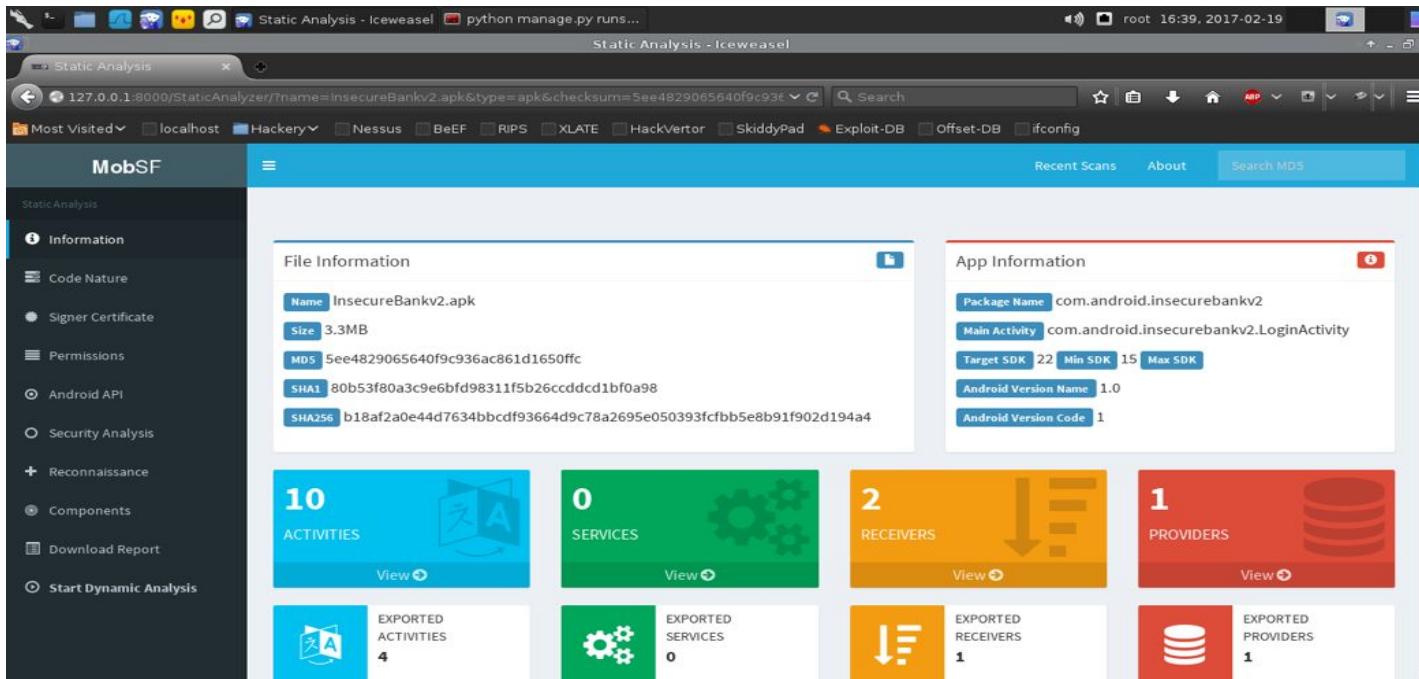
Mobil Sızma Testi Araçları

Mobile Security Framework (MobSF)



Mobil Sızma Testi Araçları

Mobile Security Framework (MobSF)



Mobil Sızma Testi Araçları

Mobile Security Framework (MobSF)

The screenshot shows a Linux desktop environment with a terminal window at the top and a web browser window below it. The terminal window has tabs for 'Static Analysis - Iceweasel' and 'python manage.py runs...'. The browser window displays the MobSF interface for static analysis of an APK file named 'InsecureBankv2.apk'. The main page shows a table of Android permissions:

PERMISSION	STATUS	INFO	DESCRIPTION
android.permission.SEND_SMS	dangerous	send SMS messages	Allows application to send SMS messages. Malicious applications may cost you money by sending messages without your confirmation.
android.permission.READ_CONTACTS	dangerous	read contact data	Allows an application to read all of the contact (address) data stored on your phone. Malicious applications can use this to send your data to other people.
android.permission.USE_CREDENTIALS	dangerous	use the authentication credentials of an account	Allows an application to request authentication tokens.
android.permission.ACCESS_NETWORK_STATE	normal	view network status	Allows an application to view the status of all networks.
android.permission.ACCESS_COARSE_LOCATION	dangerous	coarse (network-based) location	Access coarse location sources, such as the mobile network database, to determine an approximate phone location, where available. Malicious applications can use this to determine approximately where you are.
android.permission.INTERNET	dangerous	full Internet access	Allows an application to create network sockets.

Mobil Sızma Testi Araçları

Drozer

Drozerda mobil testlerde kullanılan dinamik analiz yapan bir frameworktur.

Uygulama çalışırken test etme imkanı verir.

Diğer AndroBugs ve MobSF de ise statik analizi yaptı fakat uygulama çalışmıyordu. Drozer'da uygulama çalışırken testlerimizi gerçekleştiriyoruz.

Drozer'ı kendi test pcnize kurduktan sonra aynı zamanda emulatordeki mobil cihaza da yükleyerek birbirleri ile haberleşmesini sağlıyoruz.

Mobil Sızma Testi Araçları

Drozer

Kendi test pc nize drozeri kurduktan sonra agent.apk yi emulatore atmayı unutmayın.

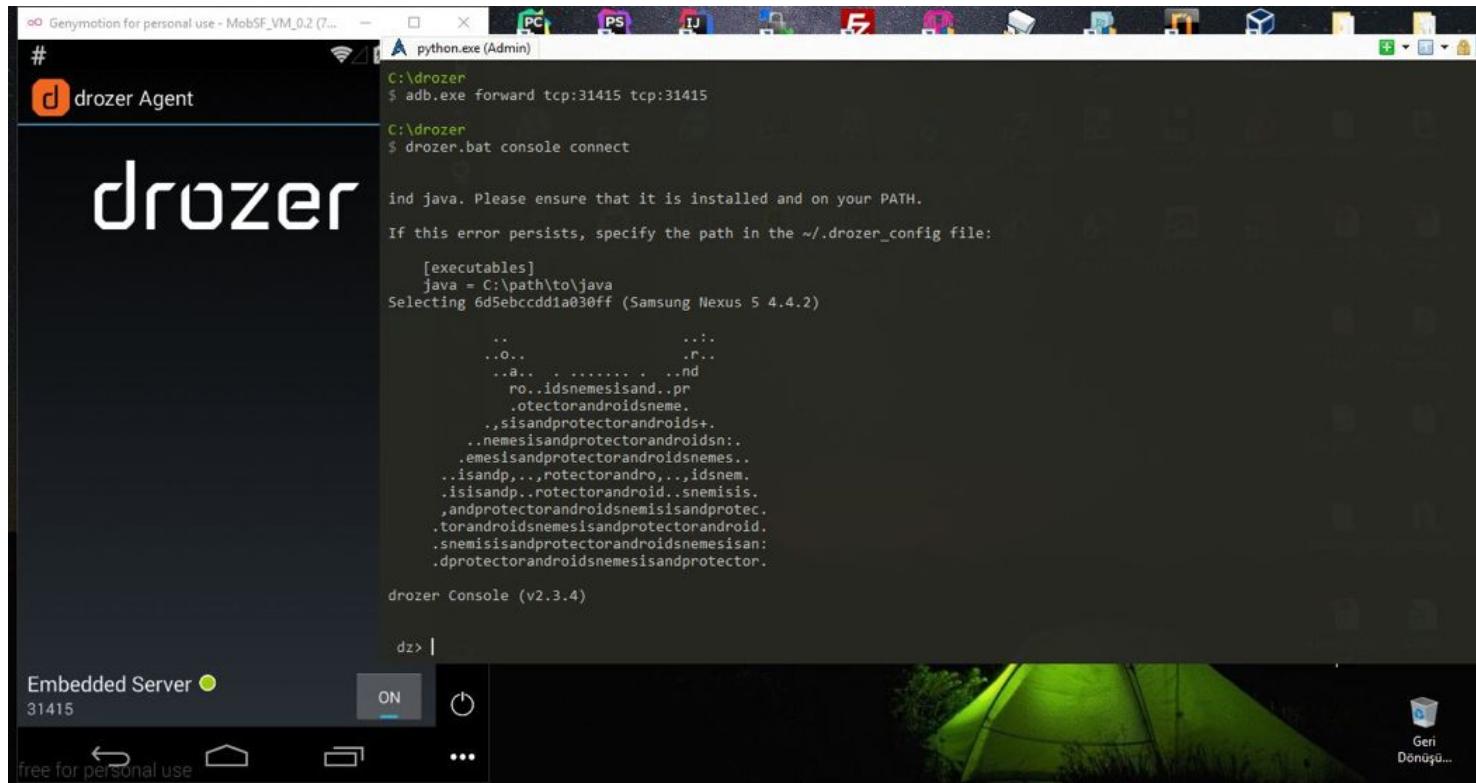
Bunu ister sürükle bırak ile istersenizde **adb. install agent.apk** komutu ile yapabilirsiniz.

Yükledikten sonra agent.apk yi emulatorde açarak off dan on'a alınız.

Daha sonra kurulum sayfasında gösterdiği gibi **adb forward tcp:31415 tcp:31415** komutunu verip **drozer console connect** diyerek drozerin komut satırına düşebilirsiniz.

Mobil Sızma Testi Araçları

Drozer



Mobil Sızma Testi Araçları

Drozer

run
app.package.list
-f insecurebank
komutu ile
kurulu paketler
arasında adı
insecurebank
olan paketi
arıyoruz.

```
...          ...
..o..      .r..
..a.. . . . . . .nd
ro..idsnemesisand..pr
.otectorandroidsneme.
.sisandprotectorandroids+.
..nemesisandprotectorandroidsn:.
.emesisandprotectorandroidsnemes..
..isandp...rotectorandro...idsnem.
.isisandp..rotectorandroid..snemisis.
.andprotectorandroidsnemisisandprotec.
.torandroidsnemesisandprotectorandroid.
.snemisisandprotectorandroidsnemesisan:
.dprotectorandroidsnemesisandprotector.

drozer Console (v2.3.4)
dz>
dz> run app.package.list -f insecurebank
com.android.insecurebankv2 (InsecureBankv2)
dz> run app.package.info -a com.android.insecurebankv2
Package: com.android.insecurebankv2
    Application Label: InsecureBankv2
    Process Name: com.android.insecurebankv2
    Version: 1.0
    Data Directory: /data/data/com.android.insecurebankv2
    APK Path: /data/app/com.android.insecurebankv2-1.apk
    UID: 10054
    GID: [3003, 1028, 1015]
    Shared Libraries: null
    Shared User ID: null
    Uses Permissions:
        - android.permission.INTERNET
        - android.permission.WRITE_EXTERNAL_STORAGE
```

Mobil Sızma Testi Araçları

Drozer

```
dz> run app.package.list -f insecurebank
com.android.insecurebankv2 (InsecureBankv2)
dz> run app.package.info -a com.android.insecurebankv2
Package: com.android.insecurebankv2
Application Label: InsecureBankv2
Process Name: com.android.insecurebankv2
Version: 1.0
Data Directory: /data/data/com.android.insecurebankv2
APK Path: /data/app/com.android.insecurebankv2-1.apk
UID: 10054
GID: [3003, 1028, 1015]
Shared Libraries: null
Shared User ID: null
Uses Permissions:
- android.permission.INTERNET
- android.permission.WRITE_EXTERNAL_STORAGE
- android.permission.SEND_SMS
- android.permission.USE_CREDENTIALS
- android.permission.GET_ACCOUNTS
- android.permission.READ_PROFILE
- android.permission.READ_CONTACTS
- android.permission.READ_PHONE_STATE
- android.permission.READ_CALL_LOG
- android.permission.ACCESS_NETWORK_STATE
- android.permission.ACCESS_COARSE_LOCATION
- android.permission.READ_EXTERNAL_STORAGE
Defines Permissions:
- None

dz> |
```

Mobil Sızma Testi Araçları

Drozer

```
C:\drozer
$ drozer.bat console devices
Could not find java. Please ensure that it is installed and on your PATH.

If this error persists, specify the path in the ~/.drozer_config file:

[executables]
java = C:\path\to\java
List of Bound Devices

Device ID          Manufacturer        Model           Software
6d5ebccdd1a030ff  Samsung            Nexus 5         4.4.2

C:\drozer
$ |
```

Mobil Sızma Testi Araçları

Drozer

```
C:\drozer
$ drozer.bat exploit list
Could not find java. Please ensure that it is installed and on your PATH.

If this error persists, specify the path in the ~/.drozer_config file:

[executables]
java = C:\path\to\java
exploit.remote.browser.addjavascriptinterface    WebView addJavascriptInterface Remote Code Execution (CVE-2012-6636)

exploit.remote.browser.knoxsmdm                  Abuse the New enrolment/UniversalMDMApplication application in Samsung Knox suite to
                                                    install rogue drozer agent

exploit.remote.browser.normalize                 Webkit Node Normalize (CVE-2010-1759)

exploit.remote.browser.useafterfree              Webkit Use After Free Exploit (Black Hat 2010)

exploit.remote.dos.remotewipe_browserdelivery   Invoke a USSD code that performs a remote wipe on Samsung Galaxy SIII (Ekoparty 2012)

exploit.remote.fileformat.polarisviewerbof_browserdelivery
                                                    Deliver Polaris Viewer 4 exploit files over browser (Mobile Pwn2Own 2012)

exploit.remote.fileformat.polarisviewerbof_generate
                                                    Generate Polaris Viewer 4 exploit DOCX (Mobile Pwn2Own 2012)

exploit.remote.socialengineering.unknownsources   Deliver the Rogue drozer Agent over browser and hold thumbs the user will install it

exploit.usb.socialengineering.usbdebugging       Install a Rogue drozer Agent on a connected device that has USB debugging enabled
```

Mobil Sızma Testi Örnekleri

Activity Bypass / InsecureBankv2 Login

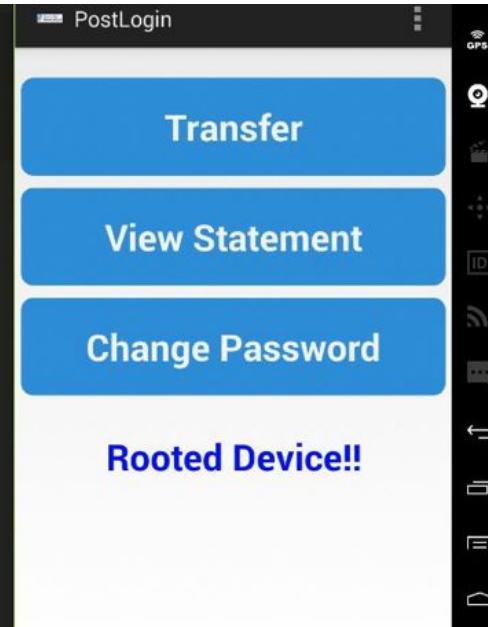
```
C:\drozer
$ drozer.bat console connect
Could not find java. Please ensure that it is installed and on your PATH.

If this error persists, specify the path in the ~/.drozer_config file:

[executables]
java = C:\path\to\java
Selecting 6d5ebccdd1a030ff (Samsung Nexus 5 4.4.2)

...          ...
...          .r..
...          ..nd
...          ro..idsnemesisand..pr
...          .otectorandroidsneme
...          ,sisinandprotectorandroids+.
...          ..nemesisandprotectorandroidsnn:
...          .emesisandprotectorandroidsnemes..
...          isandp,...rotectorandro...idsnem
...          .isisandp..rotectorandroid..snemisis.
...          andprotectorandroidsnemesisandprotec
...          .torandroidsnemesisandprotectorandroid.
...          snemisisandprotectorandroidsnemesisan:
...          dprotectorandroidsnemesisandprotector.

drozer Console (v2.3.4)
dz> run app.activity.start --component com.android.insecurebankv2 com.android.insecurebankv2.PostLogin
dz> 
```



run app.activity.start –component com.android.insecurebankv2 com.android.insecurebankv2.PostLogin

adb shell am start -n com.android.insecurebankv2/com.android.insecurebankv2.PostLogin

Mobil Sızma Testi Örnekleri

Root Detection Bypass

```
C:\drozer
$ adb.exe forward tcp:31415 tcp:31415

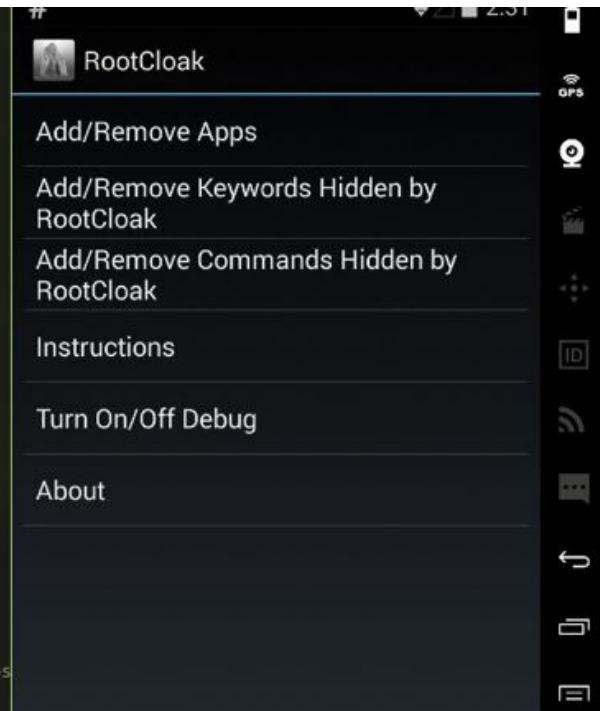
C:\drozer
$ drozer.bat console connect
Could not find java. Please ensure that it is installed and on your PATH.

If this error persists, specify the path in the ~/.drozer_config file:

[executables]
java = C:\path\to\java
Selecting 6d5ebccdd1a030ff (Samsung Nexus 5 4.4.2)

        ..          ...
        ..o..      .r..
        ..a..  .....  ..nd
        ro..idsnemesisand..pr
        .otectorandroidsneme.
        .,sisandprotectorandroids+.
        ..nemesisandprotectorandroids+.
        .emesisisandprotectorandroidsnemes..
        ..isandp...,rotectorandro,...,idsnem.
        .isisandp..rotectorandroid..snemisis.
        ,andprotectorandroidsnemisisandprotec.
        .torandroidsnemesisandprotectorandroid.
        .snemisisandprotectorandroidsnemesisan:
        .dprotectorandroidsnemesisandprotector.

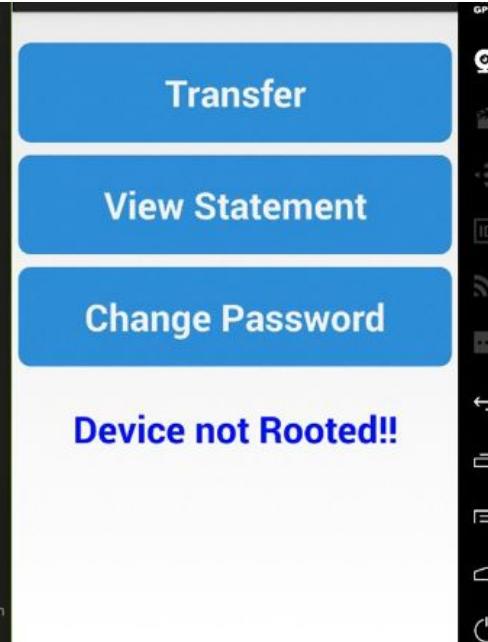
drozer Console (v2.3.4)
dz> run app.activity.start --component com.android.insecurebankv2 com.android.insecurebankv2.Pos
dz> █
```



Mobil Sızma Testi Örnekleri

Root Detection Bypass

```
Terminate batch job (Y/N)? y  
  
C:\drozer  
$ adb.exe forward tcp:31415 tcp:31415  
  
C:\drozer  
$ drozer.bat console connect  
Could not find java. Please ensure that it is installed and on your PATH.  
  
If this error persists, specify the path in the ~/.drozer_config file:  
  
[executables]  
java = C:\path\to\java  
Selecting 6d5ebccdd1a030ff (Samsung Nexus S 4.4.2)  
  
...  
.o.. . . . . . . . . . . . .  
ro.idsnesmisand.pr  
.otectorandrodnem.  
.sisandprotectorandroidst.  
.nemesisandprotectorandroidsn:.  
.emesisandprotectorandroidsomes..  
.isandp,..,rotectorandro,..,idsnem.  
.isisand..rotectorandroid..snemesis.  
.andprotectorandroidsnemisisandprotec.  
.torandroidsnemesisandprotectorandroid.  
.snemisisandprotectorandroidsnemesisan:  
.dprotectorandroidsnemesisandprotector,  
  
drozer Console (v2.3.4)  
dz> run app.activity.start --component com.android.insecurebankv2 com.android.insecurebankv2.PostLogin  
dz> █
```



Mobil Sızma Testi Örnekleri

DIVA (Damn Insecure and Vulnerable App)

The screenshot shows a mobile application interface titled "Diva". The main content area is titled "Welcome to DIVA!" and contains the following text:

DIVA (Damn insecure and vulnerable App) is an App intentionally designed to be insecure. The aim of the App is to teach developers/QA/security professionals, flaws that are generally present in the Apps due poor or insecure coding practices. If you are reading this you want to either learn App pentesting or secure coding and I sincerely hope that DIVA solves your purpose. So, sit back and enjoy the ride.

Below the welcome text is a vertical list of six items, each represented by a grey button-like box:

1. INSECURE LOGGING
2. HARDCODING ISSUES - PART 1
3. INSECURE DATA STORAGE - PART 1
4. INSECURE DATA STORAGE - PART 2
5. INSECURE DATA STORAGE - PART 3
6. INSECURE DATA STORAGE - PART 4

Mobil Sızma Testi Örnekleri

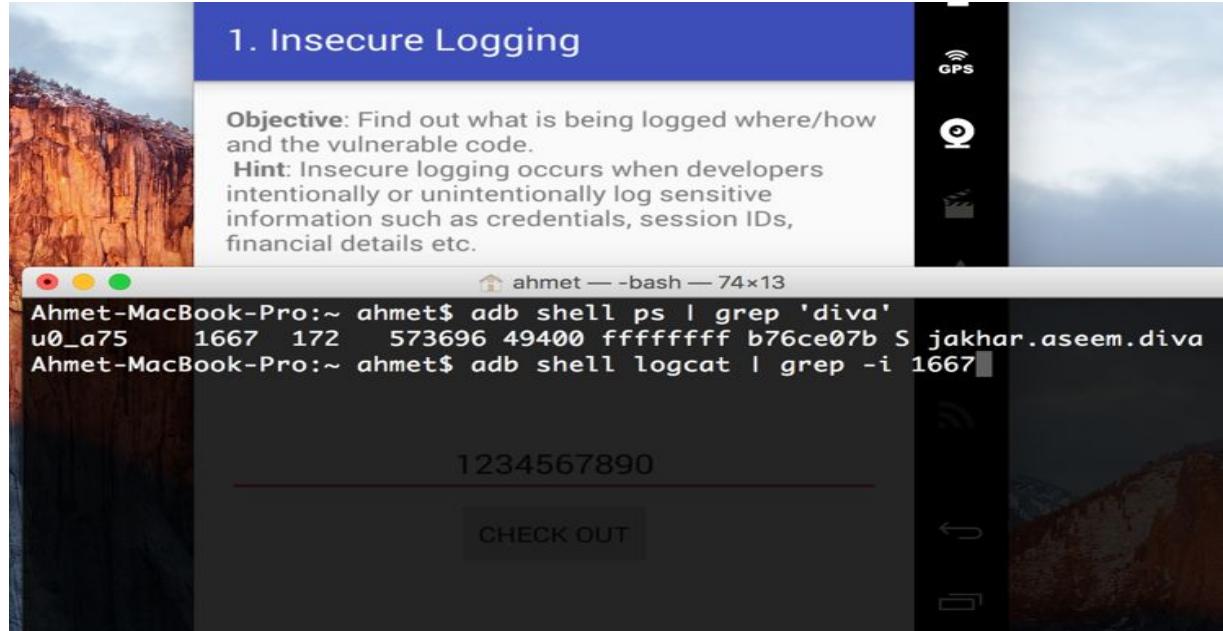
DIVA Insecure Logging

```
</> diva-beta.apk > AndroidManifest.xml

<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns="http://schemas.android.com/apk/res/android" android:versionCode="1" android:versionName="1.0" package="jakhar.aseem.diva" platformBuildVersionCode="23" platformBuildVersionName="6.0-2166767">
    <uses-sdk android:minSdkVersion="15" android:targetSdkVersion="23" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.INTERNET" />
    <application android:theme="@style/AppTheme" android:label="@string/app_name" android:icon="@mipmap/ic_launcher" android:debuggable="true" android:allowBackup="true" android:supportRtl="true">
        <activity android:theme="@style/AppTheme_NoActionBar" android:label="@string/app_name" android:name="jakhar.aseem.diva.MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:label="@string/d1" android:name="jakhar.aseem.diva.LogActivity" />
        <activity android:label="@String/d2" android:name="jakhar.aseem.diva.HardcodeActivity" />
        <activity android:label="@String/d3" android:name="jakhar.aseem.diva.InsecureDataStorage1Activity" />
        <activity android:label="@String/d4" android:name="jakhar.aseem.diva.InsecureDataStorage2Activity" />
        <activity android:label="@String/d5" android:name="jakhar.aseem.diva.InsecureDataStorage3Activity" />
        <activity android:label="@String/d6" android:name="jakhar.aseem.diva.InsecureDataStorage4Activity" />
        <activity android:label="@String/d7" android:name="jakhar.aseem.diva.SQLInjectionActivity" />
        <activity android:label="@String/d8" android:name="jakhar.aseem.diva.InputValidationURISchemeActivity" />
        <activity android:label="@String/d9" android:name="jakhar.aseem.diva.AccessControl1Activity" />
        <activity android:label="@String/opic_label" android:name="jakhar.aseem.diva.APIcredsActivity">
            <intent-filter>
                <action android:name="jakhar.aseem.diva.action.VIEW_CREDS" />
                <category android:name="android.intent.category.DEFAULT" />
            </intent-filter>
        </activity>
        <activity android:label="@String/d10" android:name="jakhar.aseem.diva.AccessControl2Activity" />
        <activity android:label="@String/opic2_label" android:name="jakhar.aseem.diva.APIcreds2Activity">
            <intent-filter>
                <action android:name="jakhar.aseem.diva.action.VIEW_CREDS2" />
                <category android:name="android.intent.category.DEFAULT" />
            </intent-filter>
        </activity>
        <provider android:name="jakhar.aseem.diva.NotesProvider" android:enabled="true" android:exported="true" android:authorities="jakhar.aseem.diva.provider.notesprovider" />
        <activity android:label="@String/d11" android:name="jakhar.aseem.diva.AccessControl3Activity" />
        <activity android:label="@String/d12" android:name="jakhar.aseem.diva.Hardcode2Activity" />
        <activity android:label="@String/pnotes" android:name="jakhar.aseem.diva.AccessControl3NotesActivity" />
        <activity android:label="@String/d13" android:name="jakhar.aseem.diva.InputValidation3Activity" />
    </application>
</manifest>
```

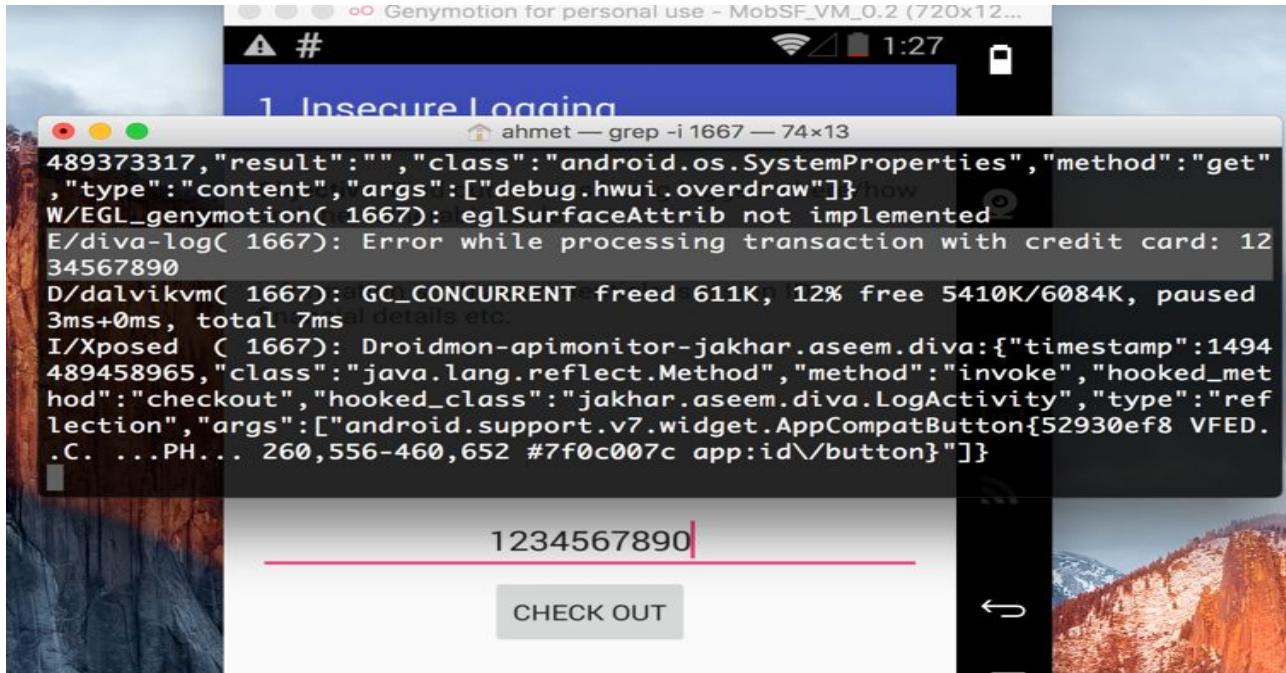
Mobil Sızma Testi Örnekleri

DIVA Insecure Logging



Mobil Sızma Testi Örnekleri

DIVA Insecure Logging



Mobil Sızma Testi Örnekleri

DIVA Hardcoding Issues - Part 1

```
</> diva-beta.apk > jakhar > aseem > diva > HardcodeActivity.java
```

```
package jakhar.aseem.diva;

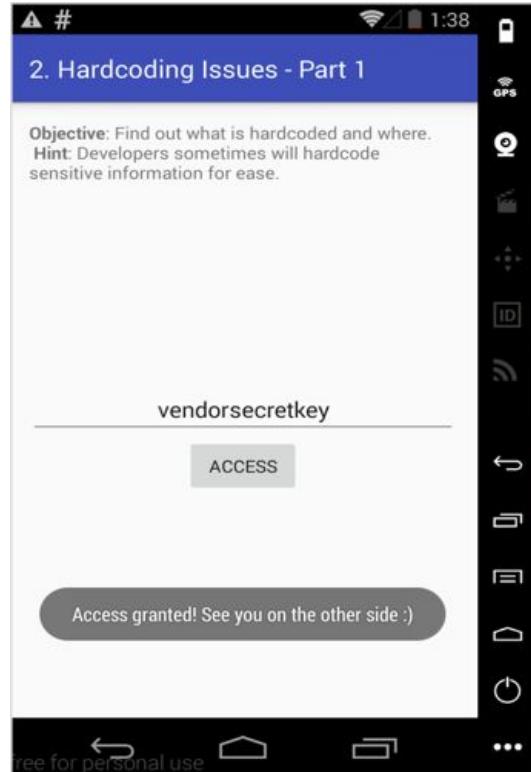
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;

public class HardcodeActivity extends AppCompatActivity {
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView((int) C0200R.layout.activity_hardcode);
    }

    public void access(View view) {
        if (((EditText) findViewById(C0200R.id.hcKey)).getText().toString().equals("vendorsecretkey")) {
            Toast.makeText(this, "Access granted! See you on the other side :)", 0).show();
        } else {
            Toast.makeText(this, "Access denied! See you in hell :D", 0).show();
        }
    }
}
```

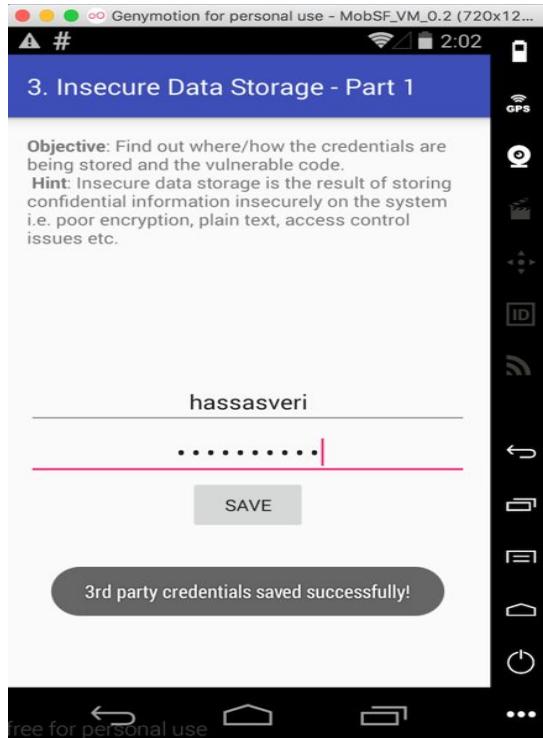
Mobil Sızma Testi Örnekleri

DIVA Hardcoding Issues - Part 1



Mobil Sızma Testi Örnekleri

DIVA Hardcoding Issues - Part 1



Mobil Sızma Testi Örnekleri

DIVA Hardcoding Issues - Part 1

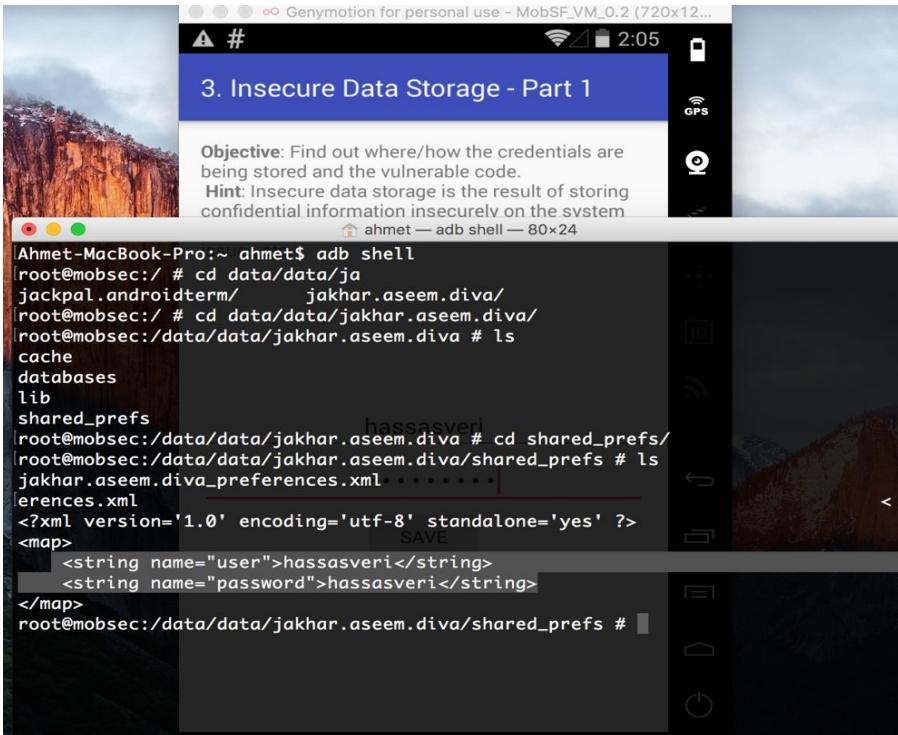
Yukarıdaki resimde gördüğünüz gibi hassasveri adında username ve password girdik.

Girdiğimiz bu verileri yazılımcı **/data/data/jakhar.aseem.diva/shared_prefs/jakhar.aseem.diva_preferences.xml** dosyasında tutmaktadır.

Adb ile cihazda shell alarak cihaz üzerinde bu dosyaya giderek görüntülediğimizde girdiğimiz username ve passworda ulaşmaktayız.

Mobil Sızma Testi Örnekleri

DIVA Insecure Data Storage - Part 1



Mobil Sızma Testi Örnekleri

DIVA Insecure Data Storage- Part 2

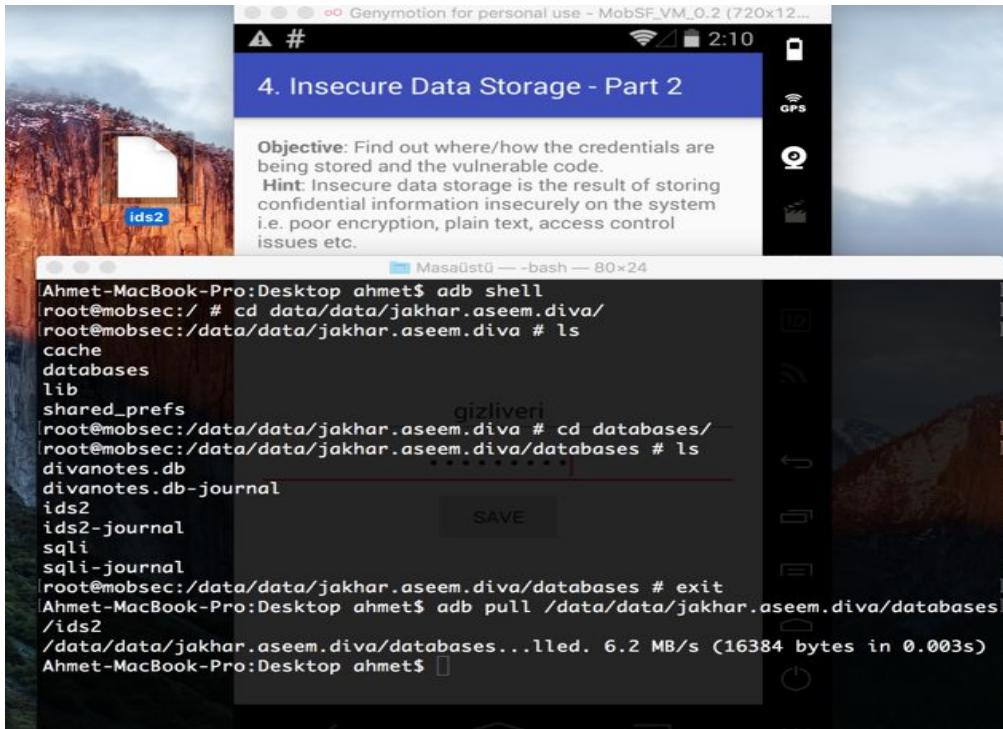
Bu bölümde güvensiz veri saklama yöntemlerinden kaynaklanan zayıfet bulunmaktadır.

Bu sefer girdiğimiz bilgiler veritabanına kaydolmaktadır.

Fakat kaydolduğu yer cihazın içinde **/data/data/jakhar.aseem.diva/databases** dizininin altındadır.

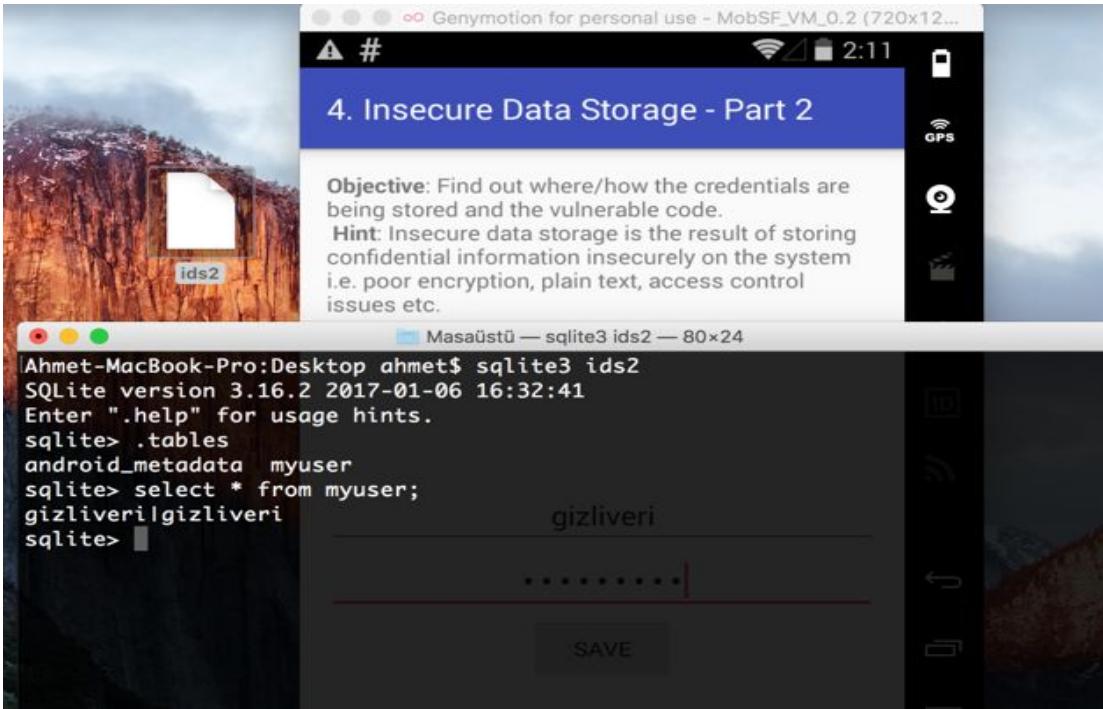
Mobil Sızma Testi Örnekleri

DIVA Insecure Data Storage- Part 2



Mobil Sızma Testi Örnekleri

DIVA Insecure Data Storage - Part 2



```
[Ahmet-MacBook-Pro:~ ahmet$ cd app_webview/  
[Ahmet-MacBook-Pro:app_webview ahmet$ ls -la  
total 120  
drwxr-xr-x 8 ahmet staff 272 16 Ago 15:59 .  
drwxr-xr-x+ 38 ahmet staff 1292 16 Ago 15:59 ..  
drwxr-xr-x 121 ahmet staff 4114 16 Ago 15:59 Cache  
-rw-r--r-- 1 ahmet staff 7168 16 Ago 15:59 Cookies  
-rw-r--r-- 1 ahmet staff 4640 16 Ago 15:59 Cookies-journal  
drwxr-xr-x 4 ahmet staff 136 16 Ago 15:59 Local Storage  
-rw-r--r-- 1 ahmet staff 38912 16 Ago 15:59 Web Data  
-rw-r--r-- 1 ahmet staff 512 16 Ago 15:59 Web Data-journal  
[Ahmet-MacBook-Pro:app_webview ahmet$ sqlite3 Web\ Data  
SQLite version 3.16.2 2017-01-06 16:32:41  
Enter ".help" for usage hints.  
sqlite> .tables  
autofill           autofill_profile_names   autofill_profiles_trash  
autofill_dates     autofill_profile_phones  credit_cards  
autofill_profile_emails  autofill_profiles    meta
```

Mobil Sızma Testi Örnekleri

DIVA Insecure Data Storage - Part 3

Bu kısımdada yine güvensiz veri saklama sorunlarından birine degenilmiş.

Uygulama bulunduğu klasöre gecici bir dosya oluşturarak hassas verileri buraya kaydetmekte.

Bu tip tespitler için her zaman uygulamanın klasörü,databases ve shared preferences dizinleri incelenmelidir.

Bunun dışında apk dosyası decompiler edilerek kaynak kodu incelenmeli ve mutla Android Manifest.xml dosyasındaki izinler ve diğer bilgiler analiz edilmelidir.

Mobil Sızma Testi Örnekleri

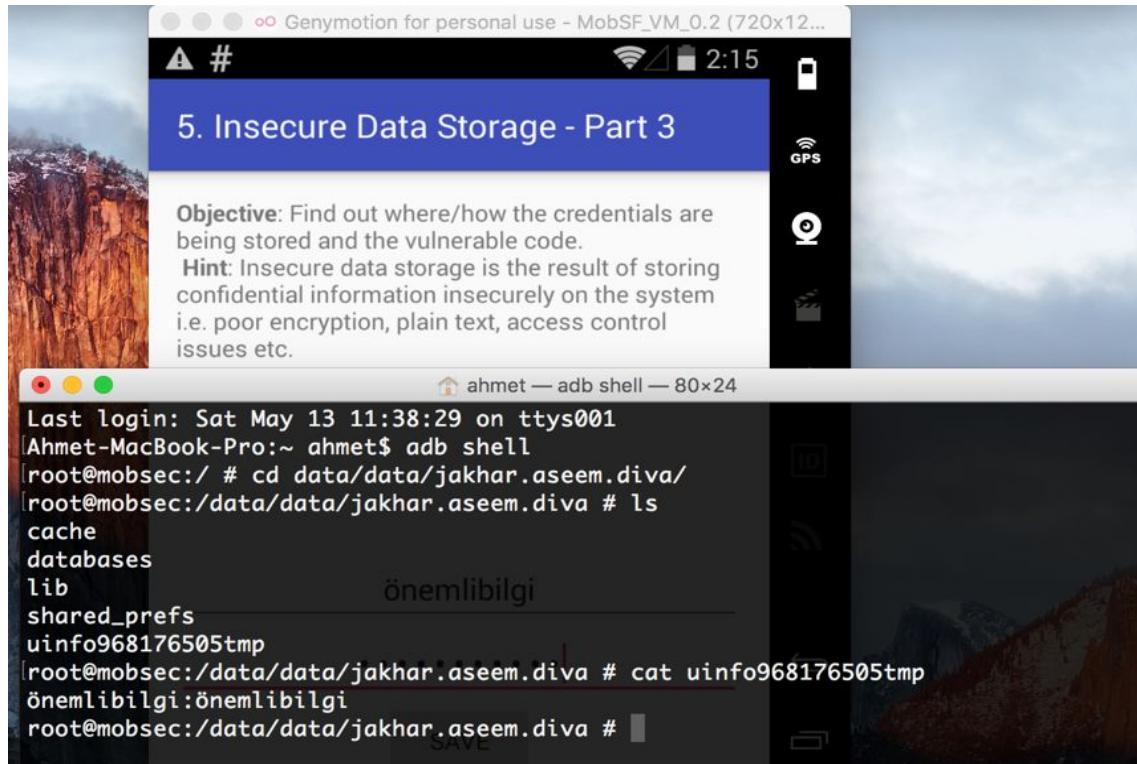
DIVA Insecure Data Storage- Part 3

```
public void saveCredentials(View paramView)
{
    EditText localEditText1 = (EditText)findViewById(2131493006);
    EditText localEditText2 = (EditText)findViewById(2131493007);
    File localFile1 = new File(getApplicationContext().dataDir);
    try
    {
        File localFile2 = File.createTempFile("userinfo", "tmp", localFile1);
        localFile2.setReadable(true);
        localFile2.setWritable(true);
        FileWriter localFileWriter = new FileWriter(localFile2);
        localFileWriter.write(localEditText1.getText().toString() + ":" + localEditText2.getText().toString() + "\n");
        localFileWriter.close();
        Toast.makeText(this, "3rd party credentials saved successfully!", 0).show();
        return;
    }
}
```

Yukarıdaki resimde gördüğünüz kodda userinfo adında geçici bir dosya oluşturmaktır ve alınan değerler buna yazılmaktadır.

Mobil Sızma Testi Örnekleri

DIVA Insecure Data Storage - Part 3



Mobil Sızma Testi Örnekleri

DIVA Insecure Data Storage - Part 4

Bu kısımdada yine girilen bilgiler cihaz içinde güvensiz şekilde tutulmaktadır.

Verilerimizi girip save diyoruz. Bu sefer sdcard'da bir dosya oluşturup ona kaydedilmektedir.

6. Insecure Data Storage - Part 4

Objective: Find out where/how the credentials are being stored and the vulnerable code.

Hint: Insecure data storage is the result of storing confidential information insecurely on the system i.e. poor encryption, plain text, access control issues etc.

The screenshot shows a mobile application interface. At the top, it says "6. Insecure Data Storage - Part 4". Below that is a section titled "Objective" with the text "Find out where/how the credentials are being stored and the vulnerable code." and a "Hint" section with the text "Insecure data storage is the result of storing confidential information insecurely on the system i.e. poor encryption, plain text, access control issues etc.". The main part of the screen shows a text input field containing "gizlibilgi" and some redacted text represented by dots. Below the input field is a grey "SAVE" button.

Mobil Sızma Testi Örnekleri

DIVA Insecure Data Storage - Part 4

```
</> diva-beta.apk > jakhar > aseem > diva > InsecureDataStorage4Activity.java

package jakhar.aseem.diva;

import android.os.Bundle;
import android.os.Environment;
import android.support.v7.app.AppCompatActivity;
import android.util.Log;
import android.view.View;
import android.widget.EditText;
import android.widget.Toast;
import java.io.File;
import java.io.FileWriter;

public class InsecureDataStorage4Activity extends AppCompatActivity {
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(C0200R.layout.activity_insecure_data_storage4);
    }

    public void saveCredentials(View view) {
        EditText usr = (EditText) findViewById(C0200R.id.ids4Usr);
        EditText pwd = (EditText) findViewById(C0200R.id.ids4Pwd);
        try {
            File uinfo = new File(Environment.getExternalStorageDirectory().getAbsolutePath() + "/.uinfo.txt");
            uinfo.setReadable(true);
            uinfo.setWritable(true);
            FileWriter fw = new FileWriter(uinfo);
            fw.write(usr.getText().toString() + ":" + pwd.getText().toString() + "\n");
            fw.close();
            Toast.makeText(this, "3rd party credentials saved successfully!", 0).show();
        } catch (Exception e) {
            Toast.makeText(this, "File error occurred", 0).show();
            Log.d("Div", "File error: " + e.getMessage());
        }
    }
}
```

Mobil Sızma Testi Örnekleri

DIVA Insecure Data Storage - Part 4

```
[root@mobsec:/mnt/sdcard # ls -la
drwxrwxrwx root      root          2015-10-05 22:27 .RootBrowser
-rw-rw-rwx root      root          22 2017-05-13 14:17 .uinfo.txt
drwxrwxrwx root      root          2016-03-07 23:25 150273
drwxrwxrwx root      root          2014-06-02 15:57 Alarms
drwxrwxrwx root      root          2014-06-10 18:24 Android
drwxrwxrwx root      root          2015-09-05 20:37 DCIM
drwxrwxrwx root      root          2017-04-13 20:58 Download
drwxrwxrwx root      root          2014-06-02 15:57 Movies
drwxrwxrwx root      root          2014-06-02 15:57 Music
drwxrwxrwx root      root          2014-06-02 15:57 Notifications
drwxrwxrwx root      root          2014-06-02 15:57 Pictures
drwxrwxrwx root      root          2014-06-02 15:57 Podcasts
drwxrwxrwx root      root          2014-06-06 16:16 Ringtones
drwxrwxrwx root      root          2017-03-31 18:32 XSSUnpinning
drwxrwxrwx root      root          2014-06-07 16:53 romtoolbox
[root@mobsec:/mnt/sdcard #
[root@mobsec:/mnt/sdcard #
[root@mobsec:/mnt/sdcard # cat .uinfo.tx
gizlibilgi:gizlibilgi
root@mobsec:/mnt/sdcard #
```

Mobil Sızma Testi Örnekleri

DIVA Input Validation Issues – Part 1

Bu kısımda input kontrol eksikliğinden kaynaklanan zafiyetlere değinilmiştir.

Buradaki input girilen değeri aramaktadır.

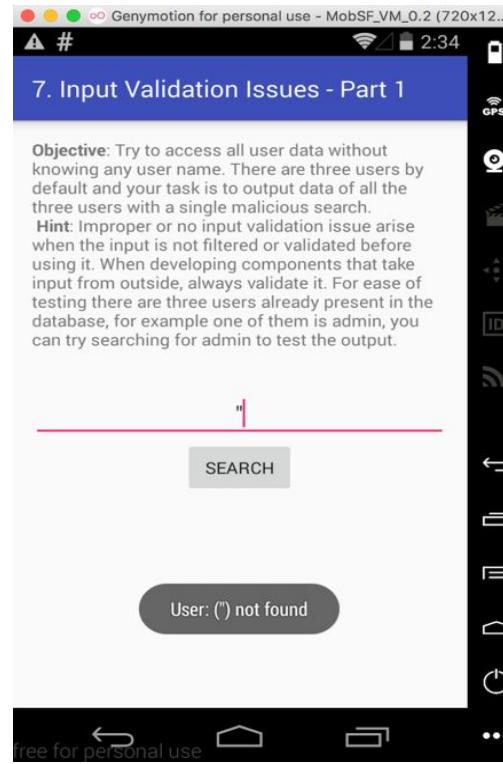
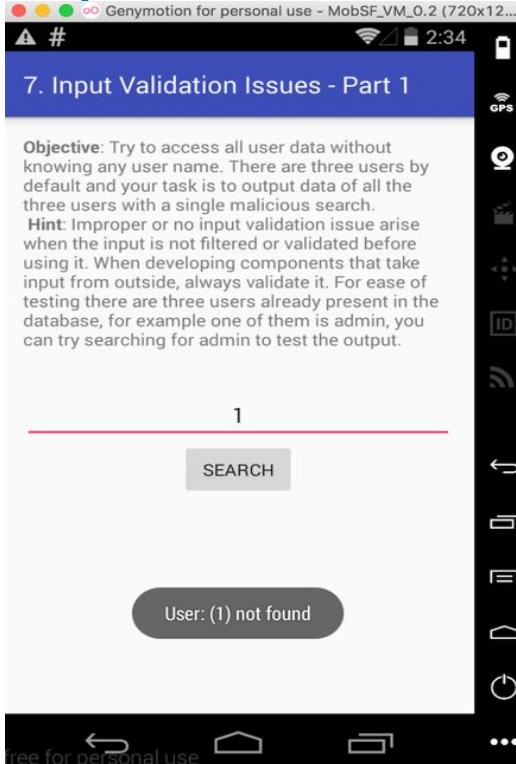
Varsa ekrana getirmekte yoksa bulunamadı olarak çıktı vermektedir.

Sql injection denemesi yapmak için “ tırnak kullanıyoruz fakat bize user bulunamadı olarak çıktı vermektedir.

‘ tırnak denediğimizde ise her hangi bir user bulunamadı çıktısı vermemekte boş döndürmektedir.

Mobil Sızma Testi Örnekleri

DIVA Input Validation Issues – Part 1



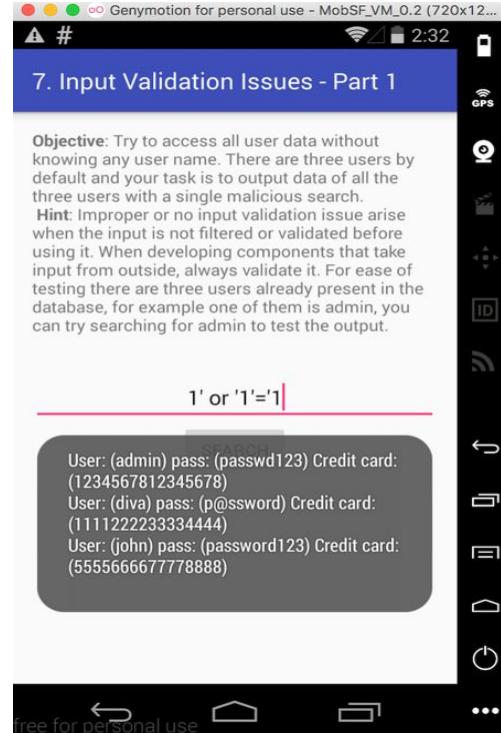
Mobil Sızma Testi Örnekleri

DIVA Input Validation Issues – Part 1

```
public void search(View paramView)
{
    EditText localEditText = (EditText) findViewById(2131493817);
    try
    {
        Cursor localCursor = this.mDB.rawQuery("SELECT * FROM sqluser WHERE user = '" + localEditText.getText().toString() + "'", null);
        StringBuilder localStringBuilder = new StringBuilder("");
        if ((localCursor != null) && (localCursor.getCount() > 0))
        {
            localCursor.moveToFirst();
            do
                localStringBuilder.append("User: (" + localCursor.getString(0) + ") pass: (" + localCursor.getString(1) + ") credit card: (" +
                while (localCursor.moveToNext());
        }
        while (true)
        {
            Toast.makeText(this, localStringBuilder.toString(), 0).show();
            return;
            localStringBuilder.append("User: (" + localEditText.getText().toString() + ") not found");
        }
    }
}
```

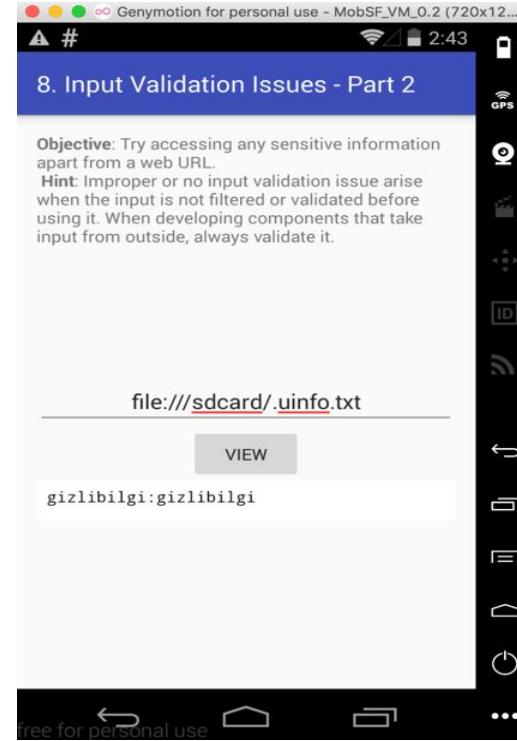
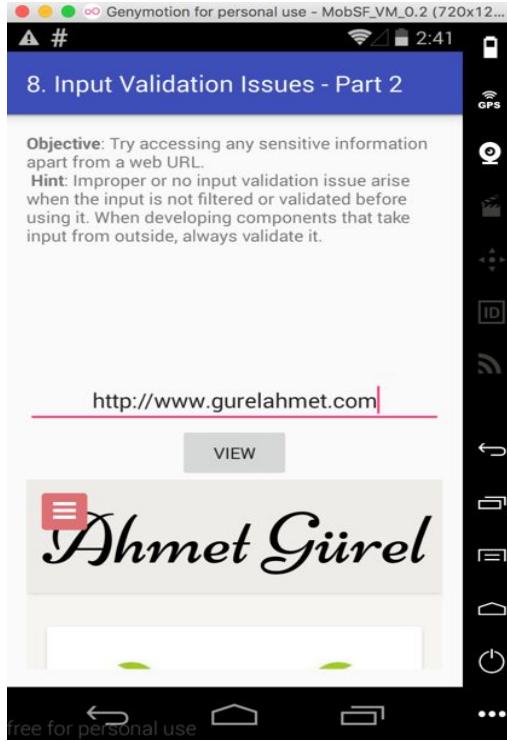
Mobil Sızma Testi Örnekleri

DIVA Input Validation Issues – Part 1



Mobil Sızma Testi Örnekleri

DIVA Input Validation Issues – Part 2



Mobil Sızma Testi Örnekleri

DIVA Acces Control Issues – Part 1

Erişim control sorunları başlıklı bu bölümde activitylerin AndroidManifest.xml dosyasında gerekli şekilde konfigurasyonu ve izinleri ayarlanamadığında activityler dışarıdan butonlara tıklanmadan açılabilmekteidir.

Drozer ve adb gibi araçlarla bu işlemler yapılabilmektedir.

Bu kısımda View Api Credentials a tıkladığımızda yeni bir activity açılarak api bilgilerini getiriyor.

Buraya tıklamadan dışarıdan komut ile activity başlatılabilmektedir.

Mobil Sızma Testi Örnekleri

DIVA Acces Control Issues – Part 1

9. Access Control Issues - Part 1

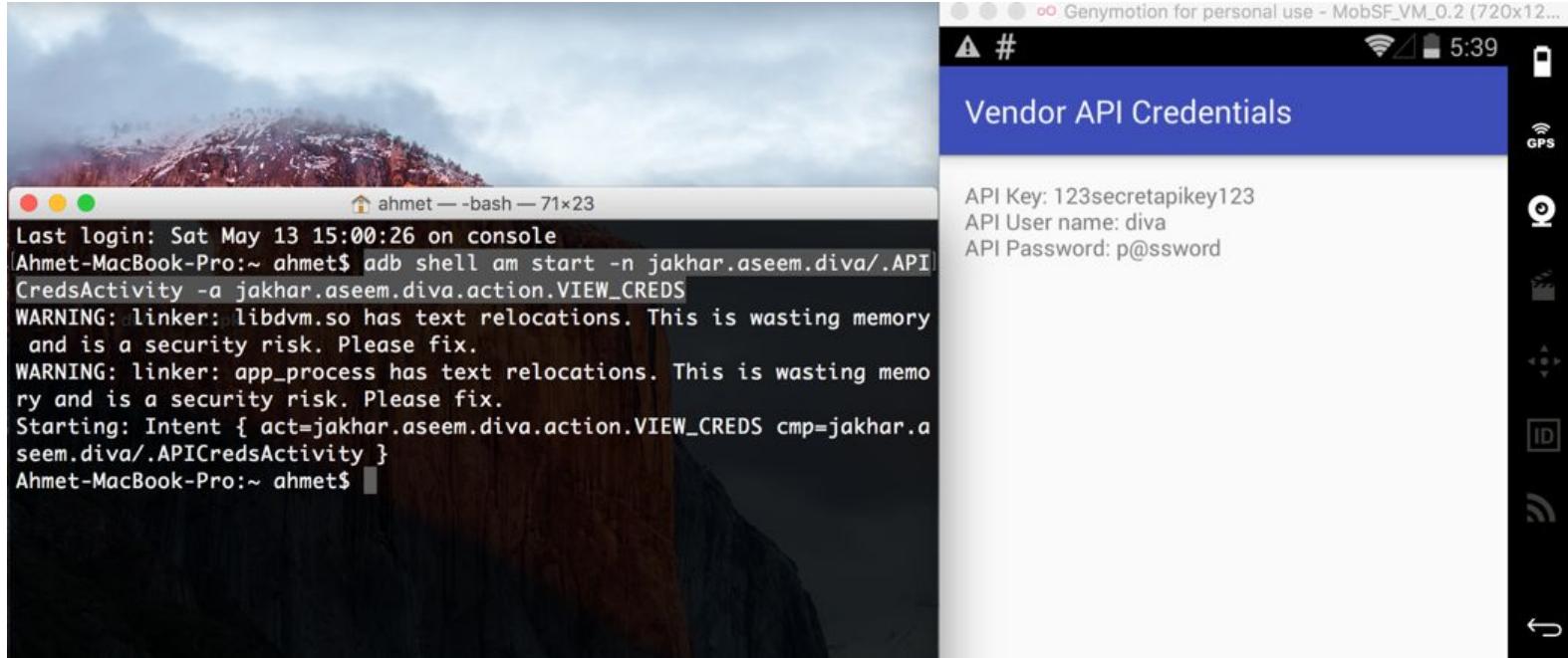
Objective: You are able to access the API credentials when you click the button. Now, try to access the API credentials from outside the app.

Hint: Components of an app can be accessed from other apps or users if they are not properly protected. Components such as activities, services, content providers are prone to this.

[VIEW API CREDENTIALS](#)

Mobil Sızma Testi Örnekleri

DIVA Acces Control Issues – Part 1



adb shell am start -n jakhar.aseem.diva/.API CredsActivity -a jakhar.aseem.diva.action.VIEW_CREDS

Mobil Sızma Testi Örnekleri

DIVA Acces Control Issues – Part 2

Erişim control sorunlarının ikinci kısmında ilk kısmına benzer sadece bu sefer uygulama açıldığında önmüze iki seçenek sunuyor Kayıt ol ve Zaten Kayıtlı Kullanıcıyım tarzında kayıt ol seçeneğine tıklandığı zaman PIN sormakta.

Fakat Zaten kayıtlı kullanıcımı sekmesinde ise direkt activity açılmakta. Bizim amacımız zaten kayıtlı kullanıcımı butonuna basmadan bu activity'i dışarıdan çalıştırırmak.

Mobil Sızma Testi Örnekleri

DIVA Acces Control Issues – Part 2

10. Access Control Issues - Part 2

Objective: You are able to access the Third Party app TVEETER API credentials after you have registered with Tveeter. The App requests you to register online and the vendor gives you a pin, which you can use to register with the app. Now, try to access the API credentials from outside the app without knowing the PIN. This is a business logic problem so you may need to see the code.

Hint: Components of an app can be accessed from other apps or users if they are not properly protected and some may also accept external inputs. Components such as activities, services, content providers are prone to this.

Register Now. Already Registered.

[VIEW TVEETER API CREDENTIALS](#)

Tveeter API Credentials

Register yourself at <http://payatu.com> to get your PIN and then login with that PIN!

Enter PIN received from Tveeter

[TVEETER API CREDENTIALS](#)

Mobil Sızma Testi Örnekleri

DIVA Acces Control Issues – Part 2

Tveeter API Credentials

TVEETER API Key: secrettveeterapikey

API User name: diva2

API Password: p@ssword2

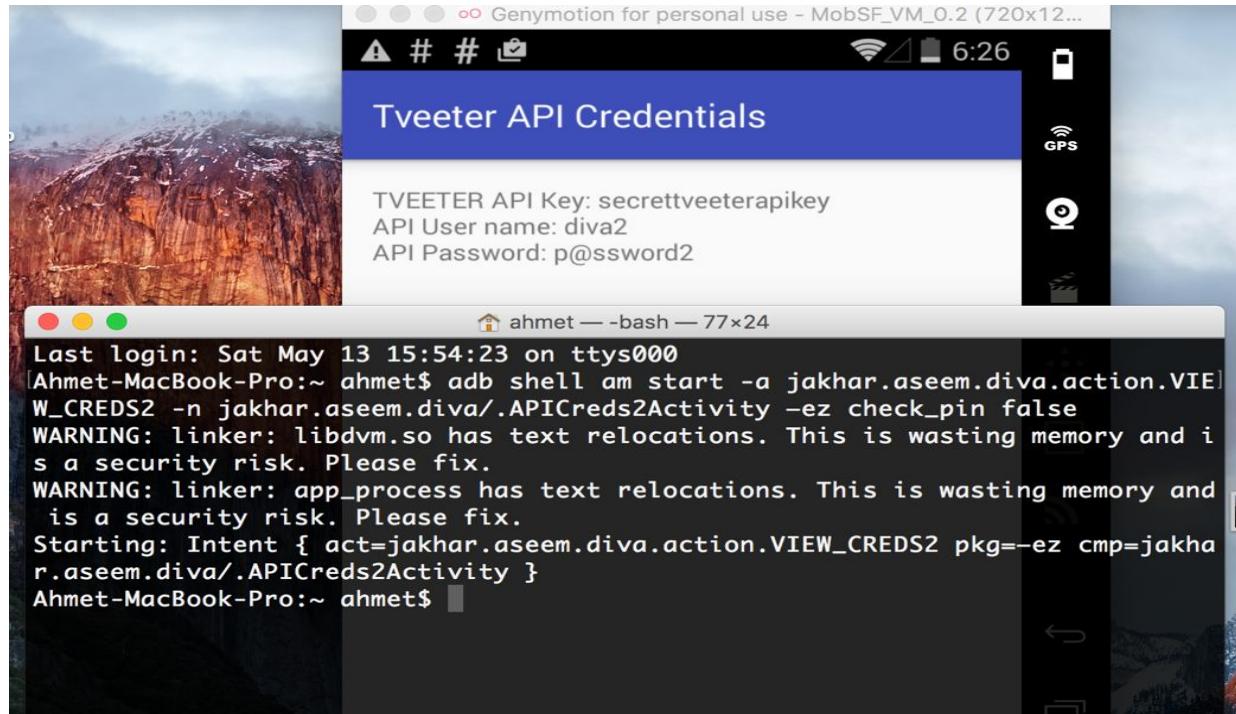
Mobil Sızma Testi Örnekleri

DIVA Acces Control Issues – Part 2

Kaynak kod incelendiği zaman check_pin true gibi bir kod satırı görülmekte ve biz bunu adb ile activity'i başlatırken –ez check_pin false parametresini ekleyerek activity'i dışarıdan tetikleyip hiç bir butona basmadan çalıştırılabilir.

Mobil Sızma Testi Örnekleri

DIVA Acces Control Issues – Part 2



Mobil Sızma Testi Örnekleri

DIVA Acces Control Issues – Part 3

Bu kısımda yine erişim control sorunlarına değinilmiştir.

Kullanıcıdan bir pin kodu girmesi istenmektedir.

Girilen PIN kodu sharedpreferences dizinin altında xml dosyasına yazılmaktadır. Buradan PIN koduna ulaşabilmektedir.

11. Access Control Issues - Part 3

Objective: This is a private notes application. You can create a PIN once and access your notes after entering the correct pin. Now, try to access the private notes from outside the app without knowing the PIN.

Hint: Components of an app can be accessed from other apps or users if they are not properly protected and some may also accept external inputs. Components such as activities, services, content providers are prone to this.

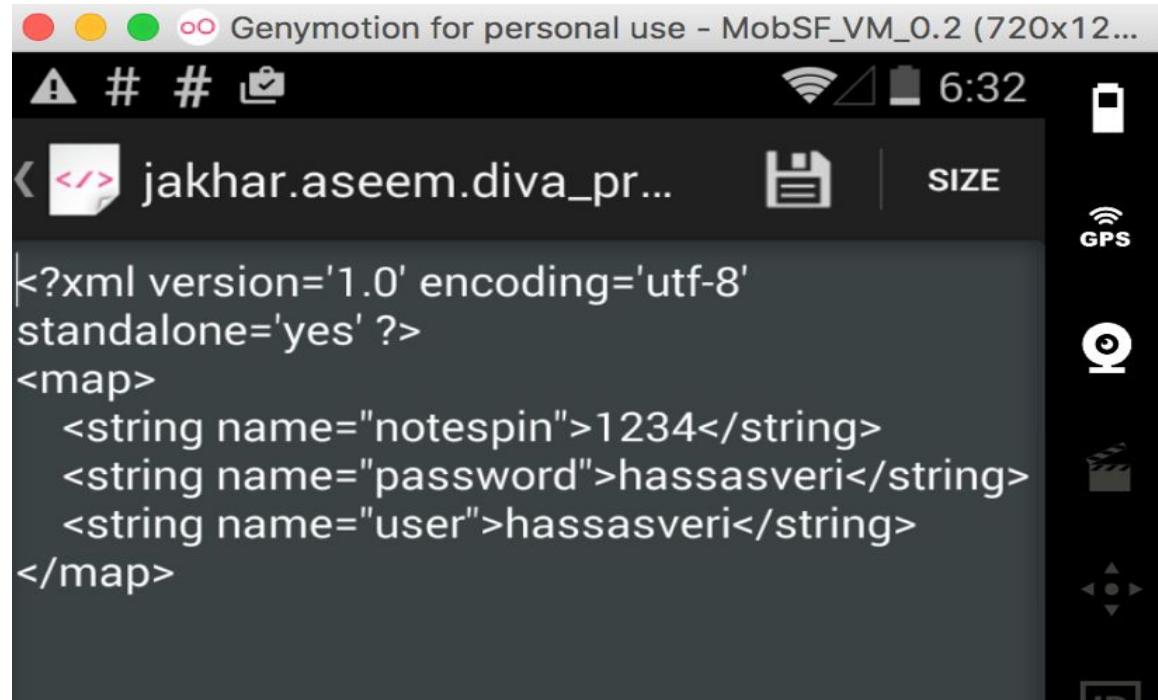
Enter 4 Digit PIN

CREATE/CHANGE PIN

GO TO PRIVATE NOTES

Mobil Sızma Testi Örnekleri

DIVA Acces Control Issues – Part 3



```
<?xml version='1.0' encoding='utf-8'  
standalone='yes' ?>  
1234</string>  
hassasveri</string>  
hassasveri</string>
```

Mobil Sızma Testi Örnekleri

DIVA Hardcoding Issues – Part 2

Bu aşama daha önce değiindiğimiz kaynak kodda bulunan parola ve doğrulama değerlerinden kaynaklanan güvenlik açıklıklarıdır.

Mobil Sızma Testi Örnekleri

DIVA Hardcoding Issues – Part 2

```
33 #include <jni.h>
34 #include <string.h>
35 #include "divajni.h"
36
37 #define VENDORKEY    "olsdfgad;lh"
38 #define CODE          ".dotdot"
39 #define CODESIZEMAX 20
40 /*
41  * Verify the key for access
42  *
43  * @param jkey    The key input by user
44  *
45  * @return 1 if jkey is valid, 0 otherwise. In other words
46  *        if the user key matches our key return 1, else return 0.
47 */
48 JNIEXPORT jint JNICALL Java_jakhar_aseem_diva_DivaJni_access(JNIEnv * env, jobject obj, jstring jkey) {
49
50     const char * key = (*env)->GetStringUTFChars(env, jkey, 0);
51
52     return ((strcmp(VENDORKEY, key, strlen(VENDORKEY)))?0:1);
53 }
54
```

Kaynak kodda bulunan bu key apk decompiler işlemlerinden sonra elde edilebilir durumdadır.

Mobil Sızma Testi Örnekleri

DIVA Hardcoding Issues – Part 2

The image is a composite of two screenshots. On the left, a terminal window titled 'player' is open on a Mac OS X desktop. The command 'strings libdivajni.so' is run, displaying various hardcoded strings. These include file paths like 'diva-beta.zip' and 'diva-beta', and Java method names such as 'Java_jakhar_aseem_diva_DivaJni_access' and 'Java_jakhar_aseem_diva_DivaJni_initiateLaunchSequence'. Other strings include 'strcpy', 'JNI_OnLoad', and various library names ('__cxa_finalize', '__cxa_atexit', etc.). On the right, a mobile application interface for 'Genymotion for personal use - MobSF_VM_0.2 (720x1280)' is shown. The title bar says '12. Hardcoding Issues - Part 2'. The main screen contains the text: 'Objective: Find out what is hardcoded and where.' and 'Hint: Developers sometimes will hardcode sensitive information for ease.' Below this is a text input field containing the string 'olsdfgad;lh'. A red underline highlights the portion 'olsdfgad;lh'. A grey button labeled 'ACCESS' is below the input field. At the bottom, a green rounded rectangle displays the message 'Access granted! See you on the other side :)'. The bottom of the screen shows standard Android navigation icons.

ahmet-MacBook-Pro:x86 ahmet\$ strings libdivajni.so

__cxa_finalize
__cxa_atexit
__stack_chk_fail
Java_jakhar_aseem_diva_DivaJni_access
Java_jakhar_aseem_diva_DivaJni_initiateLaunchSequence
strcpy
JNI_OnLoad
_edata
__bss_start
_end
libstdc++.so
libm.so
libc.so
libdl.so
libdivajni.so
d\$0[
olsdfgad;lh
.dotdot
;*2\$"
GCC: (GNU) 4.8
gold 1.11
.shstrtab
.dynsym

12. Hardcoding Issues - Part 2

Objective: Find out what is hardcoded and where.
Hint: Developers sometimes will hardcode sensitive information for ease.

olsdfgad;lh

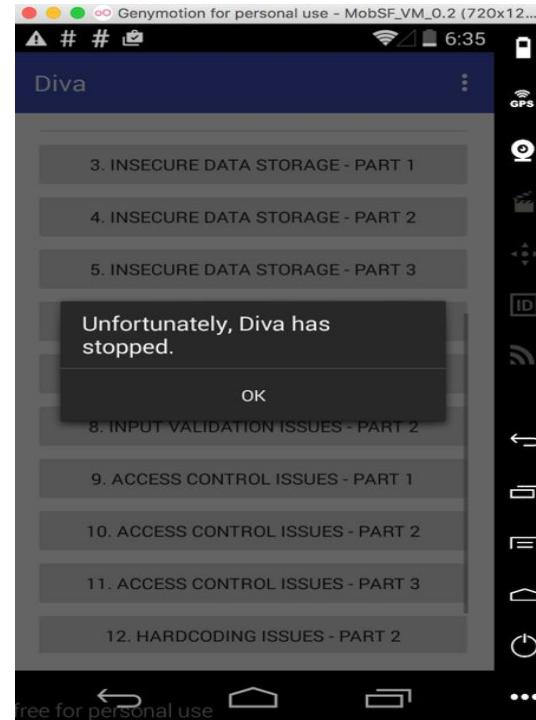
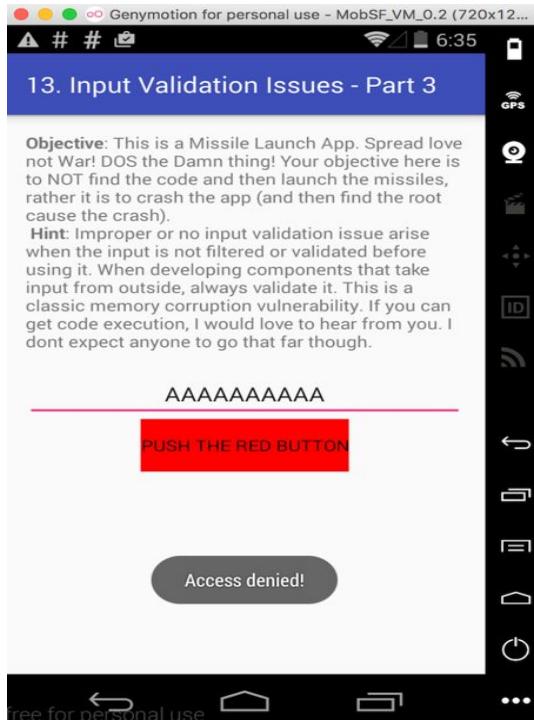
ACCESS

Access granted! See you on the other side :)

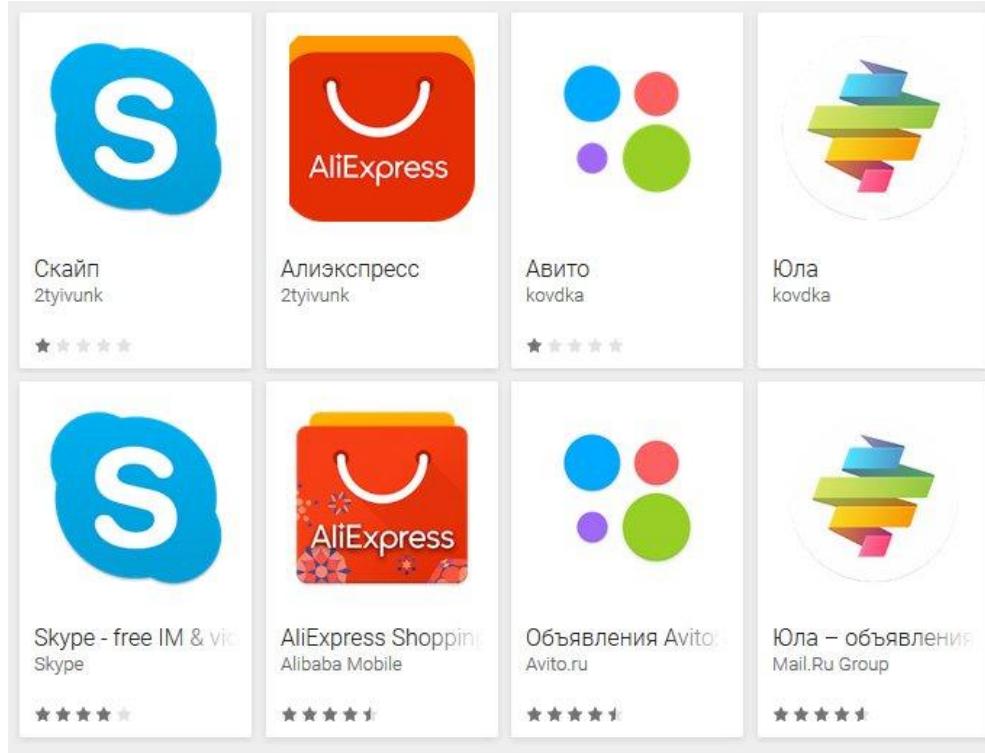
free for personal use

Mobil Sızma Testi Örnekleri

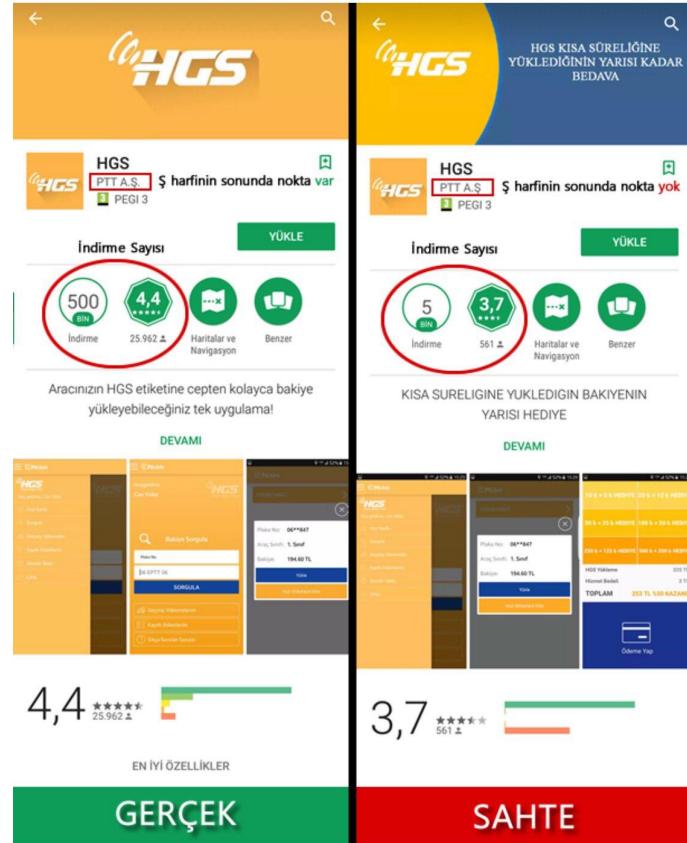
DIVA Input Validation Issues – Part 3



Google Play Uygulamaları Gerçek Olmayabilir !



Google Play Uygulamaları Gerçek Olmayabilir !



Check List

1. Network Trafiği
2. SSL Pinning Kontrolu / Bypass
3. Web Servis Kontrolleri
4. Yanlış Kimlik Doğrulaması
5. Oturum Yönetimi Kontrolü
6. IDOR
7. MongoDB/NoSQL Injection
8. Yönetim Panel Login Brute Force
9. Uygulama Komponent Analizi (Drozer)
10. File Verification
11. Uygulama Yetkilendirmeleri ve Kriptografik Kontroller
12. Lokal Veri Analizi
13. Veri Sızıntısı Android Backup
14. Tersine Mühendislik Hardcoded Credential Kontrolü
15. Patching/Modification
16. QRCode Backdoor
17. Subdomain Keşif
18. Email Keşif
19. Github vb. Siteler ile Hassas Veri Kontrolü

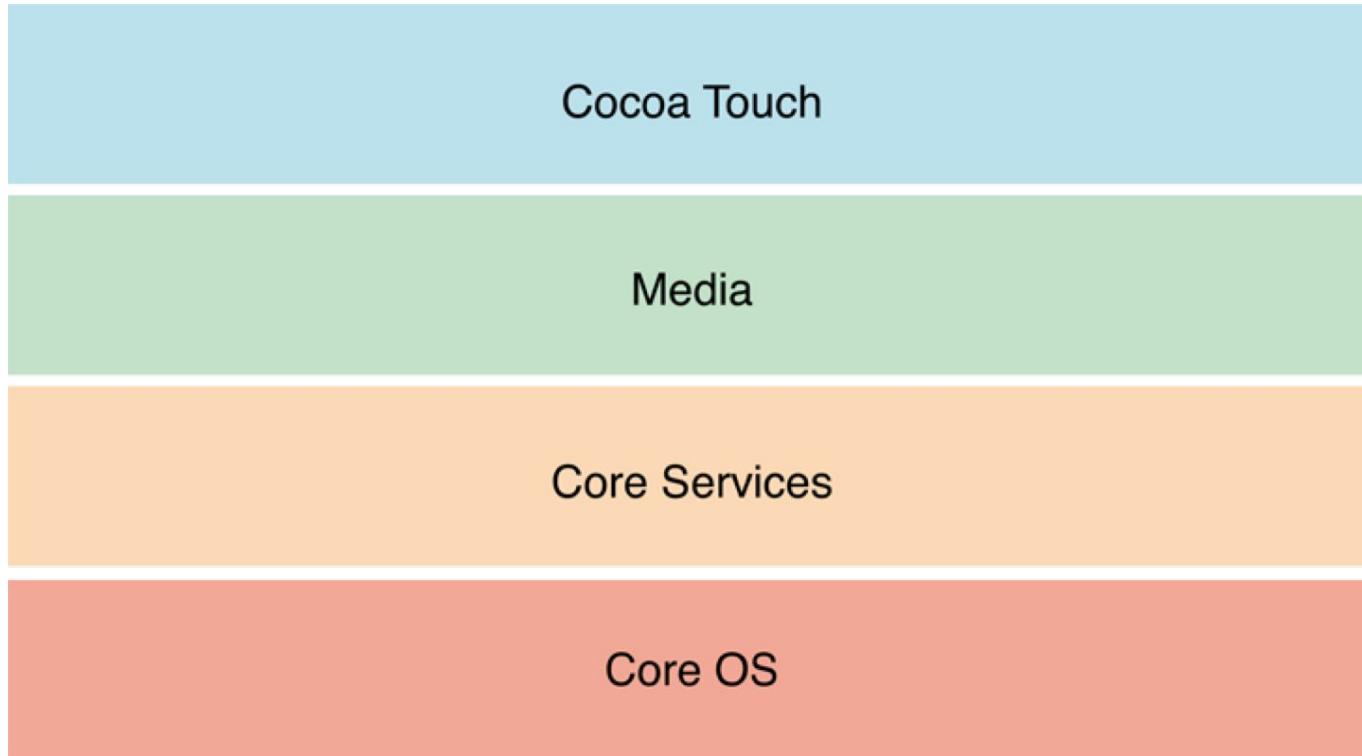
iOS Application Security

iOS (iPhone OS) Temelleri

iOS (iPhone OS) Apple'ın orijinal olarak iPhone için geliştirdiği ancak daha sonra iPod Touch ve iPad'de de kullanılan mobil işletim sistemidir.

Mac OS X'den türetilmiştir. Çekirdeği ve sistem kütüphaneleri ARM işlemciye göre modifiye edilmiştir. iOS içinde 4 katman bulundurmaktadır: Core OS tabakası, Core Servisleri tabakası, Medya tabakası ve Cocoa Touch tabakası.

iOS (iPhone OS) Temelleri



iOS (iPhone OS) Temelleri

1. Core OS Katmanı

Donanıma en yakın katman olup güç yönetimi, dosya sistemi ve diğer donanıma yakın programlama gereken birimlerin olduğu katmandır.

2. Core Services katmanı

Networking, dosya erişimleri, SQLite gibi uygulamaların Core OS kullanılarak hazırlanmış servislerin bulunduğu katmandır

3. Media katmanı

OpenAL, OpenGL ES gibi görüntü ve ses üzerine hazırlanmış kütüphanelerin bulunduğu katmandır.

Uygulamalarda videolar nasıl çalıştırılır, nasıl kaydedilir, ses nasıl kaydedilir vs. gibi hepsi bu katmandadır.

4. Cocoa Touch Katmanı

Kullanıcıya en yakın katman olan Cocoa Touch katmanı ise kullanıcı ile iletişimini sağlayan görsel arabirimleri sağlayan sınıfların yer aldığı katmandır.

Dokunmatik ekran üzerinde yapılan parmak hareketleri algılayan bir yapıya sahiptir.

iOS (iPhone OS) Temelleri

iOS Geliştirme Ortamı

Apple iOS cihazlarda geliştirme yapılabilmesi için çıkardığı Objective-C ve Swift programlama dili ile geliştirilir. Geliştirme ortamı Xcode dur. Mac OS'te geliştirme zorunluluğu vardır çünkü Xcode sadece Mac OSX işletim sisteminde çalışmaktadır. Bunun haricinde Android dünyasında olduğu gibi hybrid uygulama geliştirme platformları ile de iOS uygulama geliştirilebilmektedir. JavaScript ve diğer programlama dilleri kullanıralarak Xamarin, React Native gibi frameworkler ile de geliştirme yapılabilir. Native uygulama için Objective-C ve Swift programlama dilleri kullanılmalıdır.

iOS Depolama

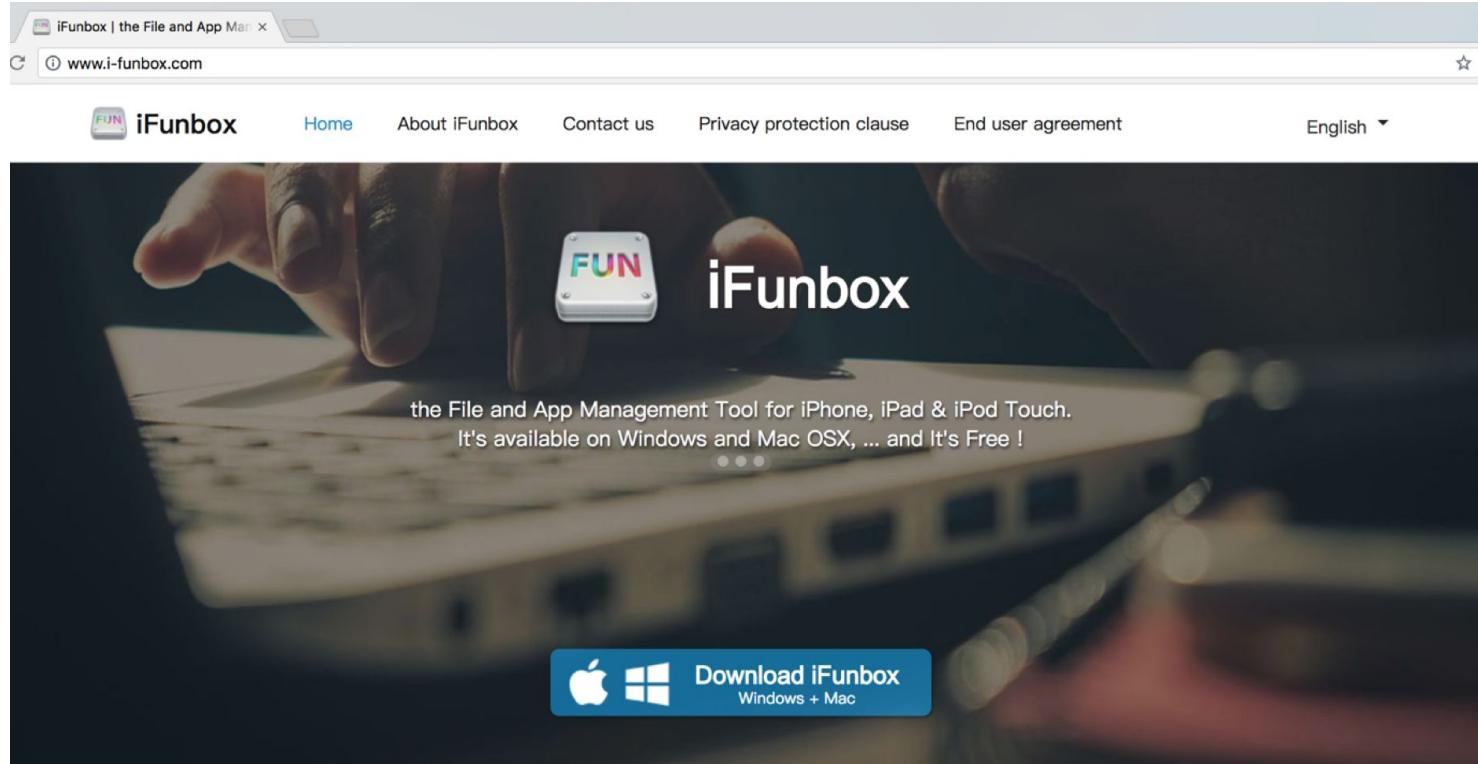
iOS cihazlar üzerinde 2 adet depolama ile kullanıcı dosyaları ile sistem dosyaları birbirinden ayrılmış durumdadır. Biri System Partition diğeri ise User Partition dır. Aynı Anroid sizma testinde bahsettiğimiz gibi cihaz orjinal işletim sisteminde kullanıcının erişemediği kısımlara cihazı rootlayarak root kullanıcı hakları ile erişmişistik. iOS sizma testlerinde yine aynı şekilde Jailbreak yükleyerek en yetkili kullanıcı olan root kullanıcısına geçiş yaparak cihaz üzerinde ki hassas bilgilerin ve uygulama dosyalarının tutulduğu alanlara geçiş yapacağız.

iOS (iPhone OS) Dosya Yönetimi ve IPA Uzantılı Dosya Yükleme

iOS işletim sistemi yüklü cihazınızı bilgisayarınıza bağladığınız iTunes uygulaması mevcutsa onun üzerinden açıp dosyalarınızı ve cihazınızı yönetebilirsiniz.

Aynı zamanda bazı kısıtlamalar olmadan işlem yapıp cihazınıza dosya ve uygulamalarınızı yönetmek istersenizde üçüncü parti uygulama olan iFunBox gibi yazılımlar mevcuttur.

iOS (iPhone OS) Dosya Yönetimi ve IPA Uzantılı Dosya Yükleme

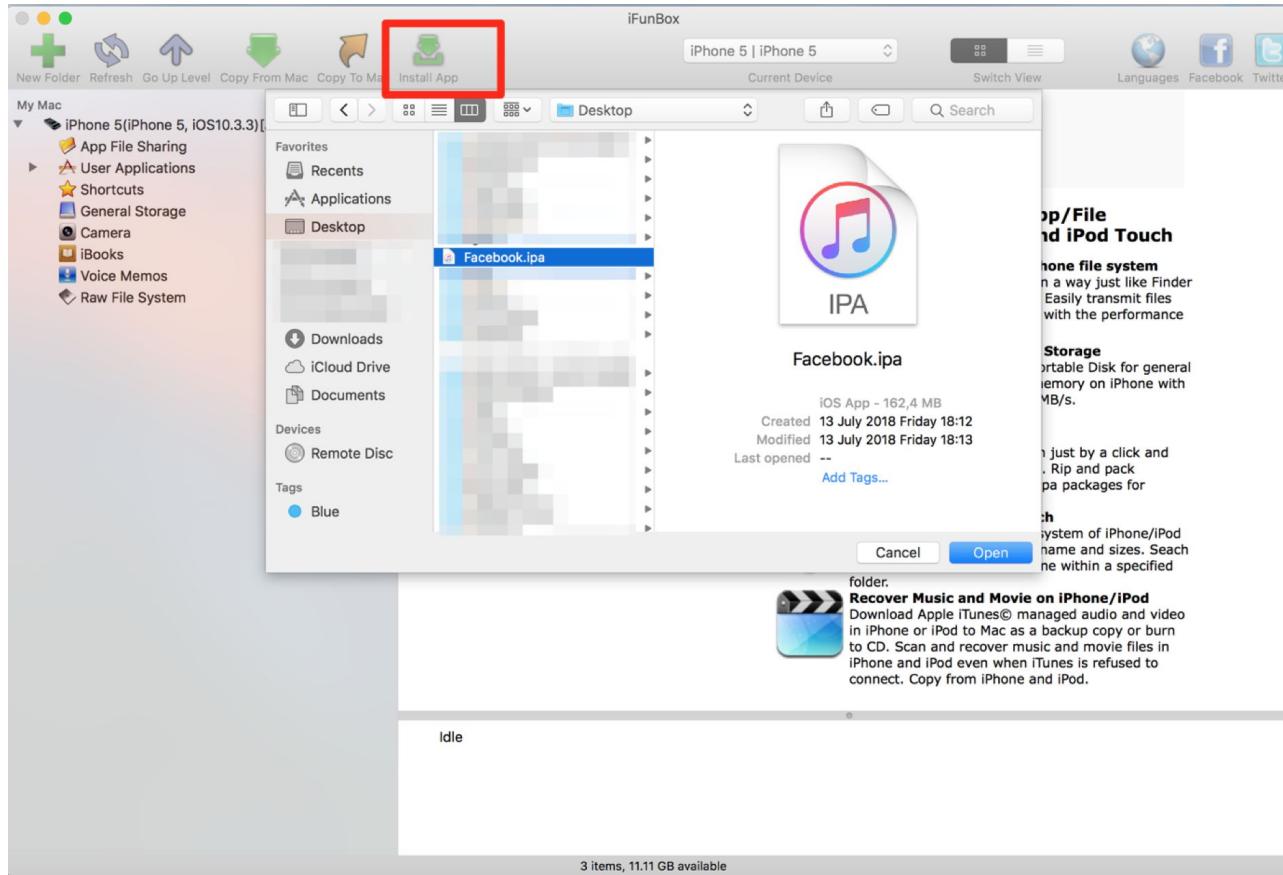


Şekilde görülen sayfa üzerinden işletim sisteminize uygun kurulum dosyasını indirerek iFunBox uygulamasını cihazınıza kurabilirsiniz.

iOS (iPhone OS) Dosya Yönetimi ve IPA Uzantılı Dosya Yükleme



iOS (iPhone OS) Dosya Yönetimi ve IPA Uzantılı Dosya Yükleme



iOS (iPhone OS) Dosya Yönetimi ve IPA Uzantılı Dosya Yükleme

Şu an bağladığımız cihaz iPhone 5 ve iOS sürümü 10.3.3 dür. Cihazın işletim sistemi jailbreak işlemi yapılmıştır Install App yazan kısımdan ipa (iOS App Store Package) uzantılı iOS işletim sistemi için uygulamaların kurulum dosyaları ile uygulama kurmak istediğimizde kurabiliyoruz.

Fakat Jailbreak yapılmamış cihazlarda güvenlik sebeplerinden dolayı bir çok uygulamanın kurulumuna izin vermemektedir. iTunes ya da Apple Store üzerinde markette bulunan uygulamalar kurulabilmektedir.

Android sizme testi kısmında android dünyasında ayarlardan yabancı kaynaklardan uygulama yüklemeyi kabul ettiğimizde APK dosyaları ile markette olmayan uygulamalarıda kurabilmekteydi. iOS işletim sistemine ipa dosyası ile uygulama kurabilmek için ya bir Geliştirici (Developer) hesabınız olmalı ya da Jailbreak yapılip root haklarında çalışan bir iOS işletim sistemine sahip olmanız gerekmektedir.

Android cihazlara rootladıkten sonra ADB ile ulaşıp cihaz dosyalarına erişmişlik iOS işletim sisteminde Jailbreak yapıldığında cihaza aynı ağ üzerinden ya da kablo üzerinden SSH ile erişilmektedir.

iOS (iPhone OS) Jailbreak İşlemi

iOS işletim sistemini özgürleştirip, daha fazla erişim sağlamak için yapılan işleme "Jailbreak" adı verilmiştir. Jailbreak işlemi işletim sistemine "root" erişimi sağlamak için kullanılan bir yöntemdir.

Bu işlem yapılırken işletim sisteminde yetki yükseltme zafiyetini kullanarak yapılmaktadır. Bunun için her sürümde Jailbreak yapılamamaktadır. Yapılan sürümlerde de Apple geçmiş versiyonlarının imzalarını kaldırdığı için biraz zorlayıcı bir süreç olabilmektedir.

Burada Jailbreak yapacağımız cihaz iPhone 5 olup iOS versiyonu 10.3.3 dür. İlk olarak Cydia Impactor aracını bilgisayarımıza indirip kurmamız gerekmektedir.

iOS (iPhone OS) Jailbreak İşlemi



Cydia Impactor

Cydia Impactor is a GUI tool for working with mobile devices. It has features already, but is still very much a work-in-progress. It is developed by saurik ([Twitter](#) and [website](#)).

You can use this tool to install IPA files on iOS and APK files on Android. It also can help you exploit the series of Android "Master Key" vulnerabilities.

Download whatever the latest version of Cydia Impactor is for [Mac OS X](#), [Windows](#), [Linux \(32-bit\)](#), or [Linux \(64-bit\)](#).
(These URLs will always redirect to the most recent versions, so feel free to directly link to the packages from howto guides.)

Note: Do *not* "Run as Administrator" Impactor; doing this makes drag/drop of files not work on Windows 10.

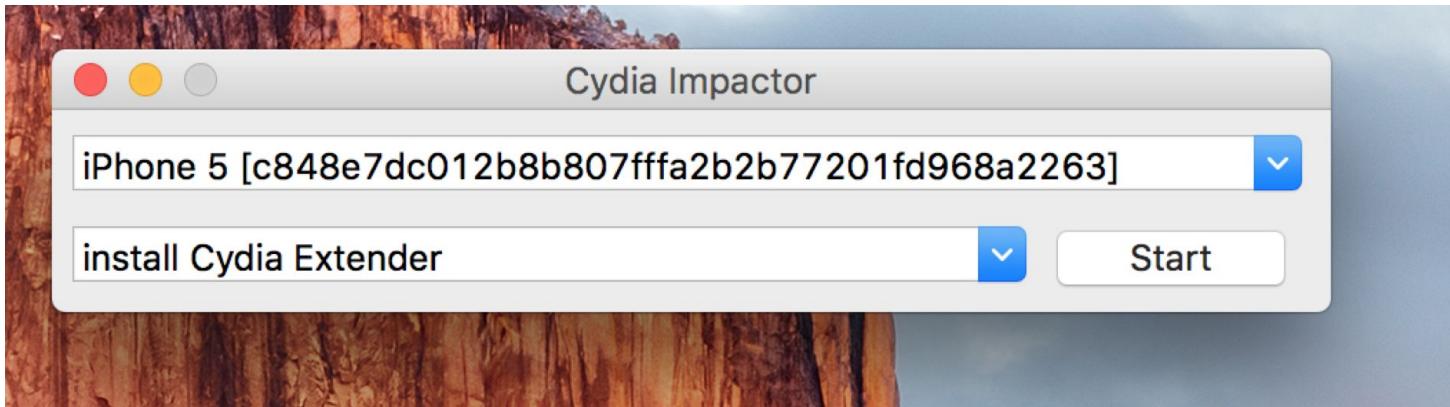
Also: Tons of people are downloading Cydia Impactor in an attempt to install some kind of Pokemon Go hack in the form of an IPA file... to their Android device... an IPA file is for devices running iOS only, not Android.

If you are on Windows, you may have to install a device driver to talk to your Android device over USB. If your device is not detected, use Impactor's USB Driver Scan feature to attempt to automatically construct and install a driver for your device. You do not need the Android SDK installed to use Impactor.

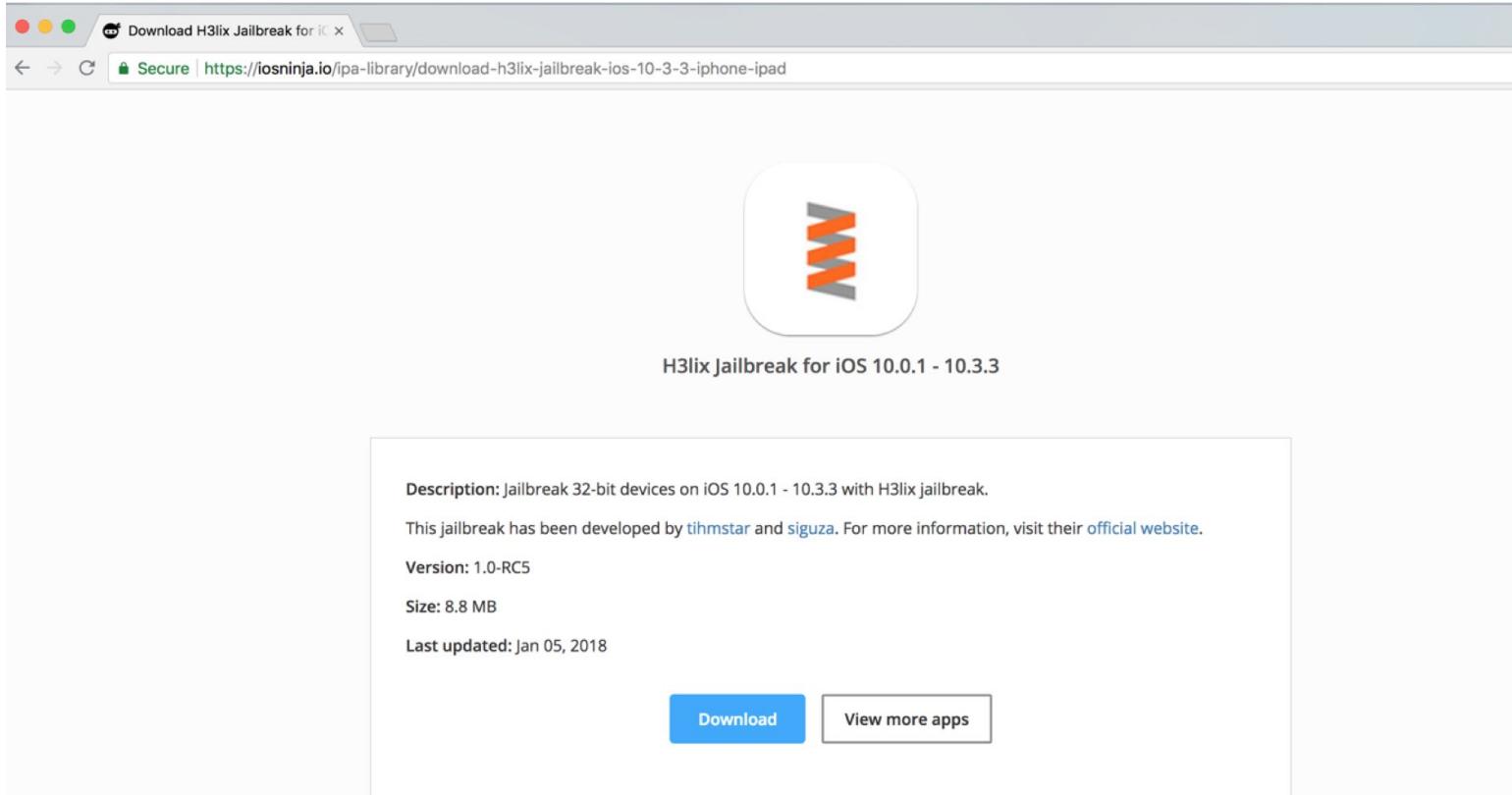
For iOS, if you are using either Windows or macOS, you definitely need to have iTunes installed for this tool to work (for different reasons). You do not need Xcode installed to use Impactor (even for features such as signing IPA files).

To download new versions, use "Check for Updates..." under the Impactor menu from inside of the application. Impactor will also occasionally prompt about new versions that come out.
(This feature is currently not available in the Linux versions.)

iOS (iPhone OS) Jailbreak İşlemi



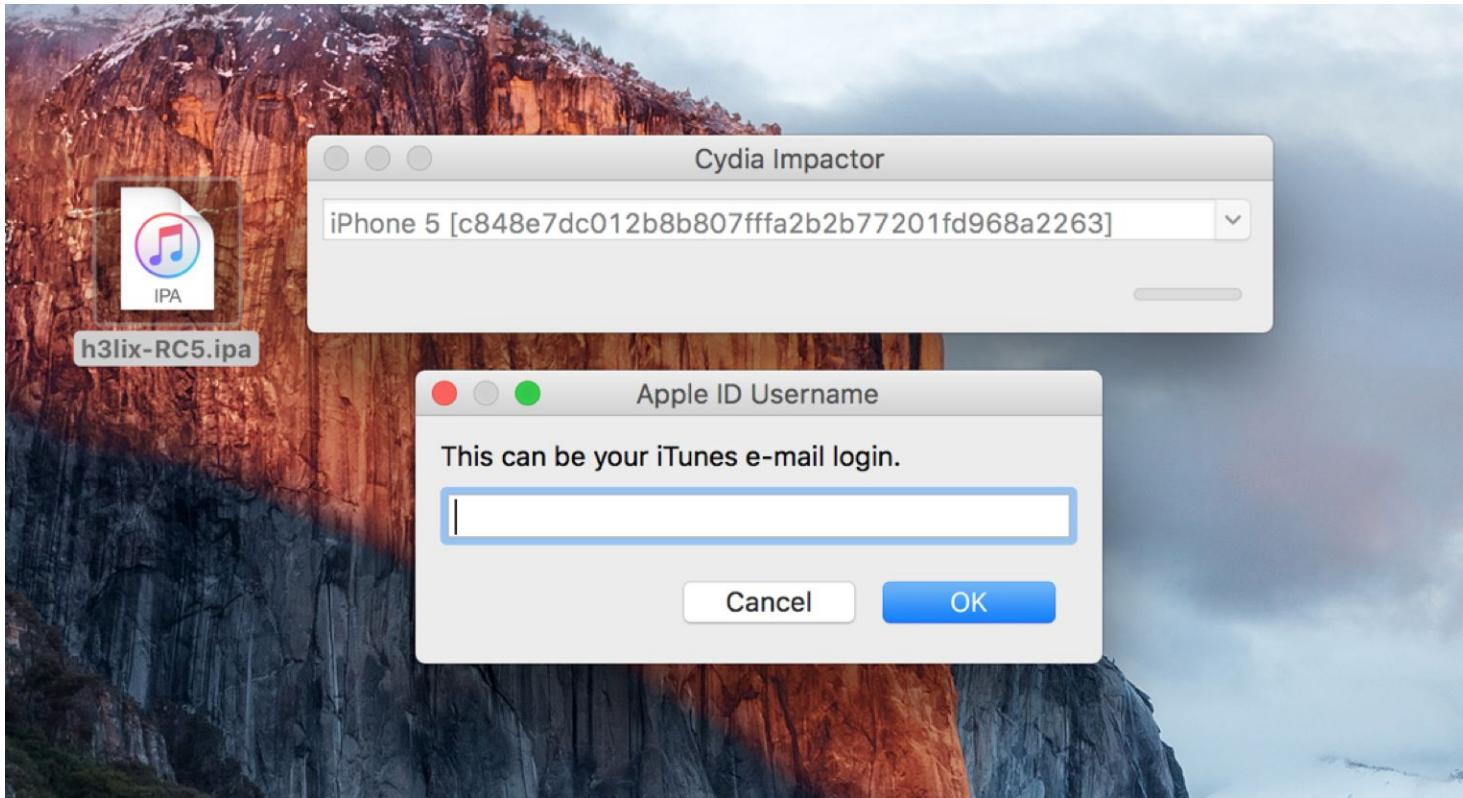
iOS (iPhone OS) Jailbreak İşlemi



The screenshot shows a web browser window with the title "Download H3lix Jailbreak for iOS". The URL in the address bar is "https://iosninja.io/ipa-library/download-h3lix-jailbreak-ios-10-3-3-iphone-ipad". The main content of the page features a large circular icon containing a stylized orange and grey zigzag pattern. Below the icon, the text "H3lix Jailbreak for iOS 10.0.1 - 10.3.3" is displayed. A detailed description box contains the following information:
Description: Jailbreak 32-bit devices on iOS 10.0.1 - 10.3.3 with H3lix jailbreak.
This jailbreak has been developed by [tihmstar](#) and [siguza](#). For more information, visit their [official website](#).
Version: 1.0-RC5
Size: 8.8 MB
Last updated: Jan 05, 2018
At the bottom of the description box are two buttons: a blue "Download" button and a white "View more apps" button.

iPhone 5 cihazımızın iOS versiyonu 10.3.3 olduğu için H3lix Jailbreak 10.0.1-10.3.3 sürümleri arasında jailbreak işlemi yapmaktadır. Şekilde görülen site üzerinden indirilebilir.

iOS (iPhone OS) Jailbreak İşlemi



iOS (iPhone OS) Jailbreak İşlemi

Şekillerde görüldüğü üzere indirdiğimiz h3lix ipa dosyasını Cydia Impactor üzerine sürükleyerek bırakıyoruz.

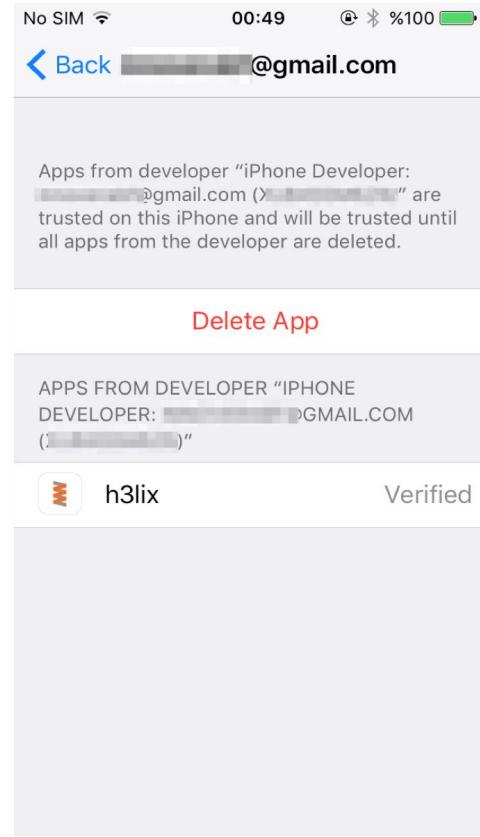
Yükleme işlemi için iTunes hesabı mail adresi ve parola istemekte.

Yoksa iTunes üzerinden ücretsiz hesap oluşturabilirsiniz.

iOS (iPhone OS) Jailbreak İşlemi



iOS (iPhone OS) Jailbreak İşlemi



iOS (iPhone OS) Jailbreak İşlemi

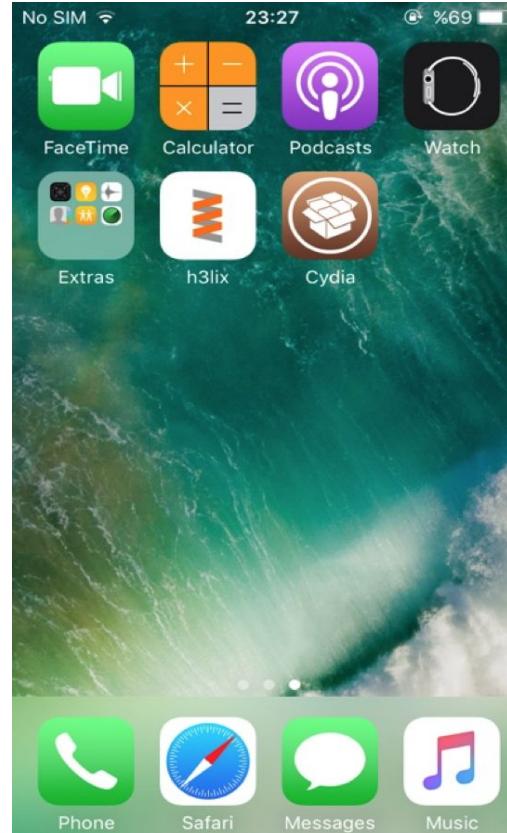
No SIM ⚡ 8:23 PM ⚡



jailbreak

iOS 10.x jailbreak by
tihmstar and siguza

Special thanks to
qwertyoruiop
jk9357

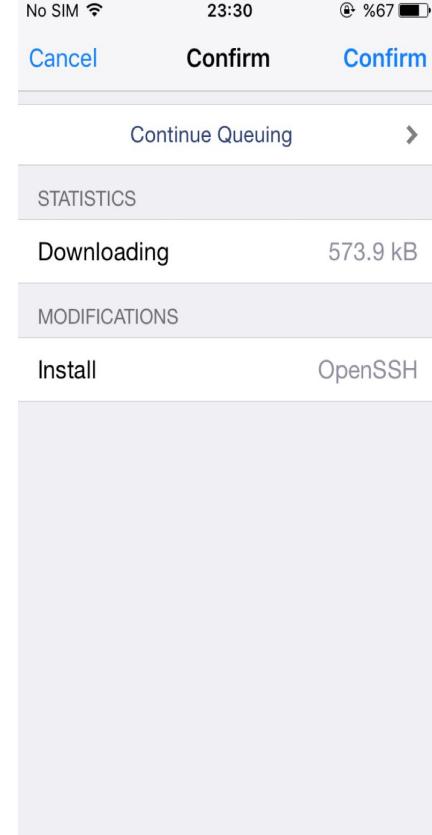
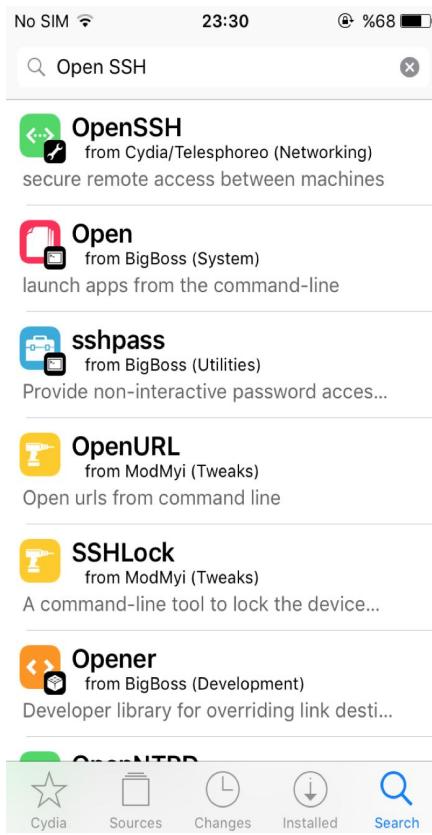
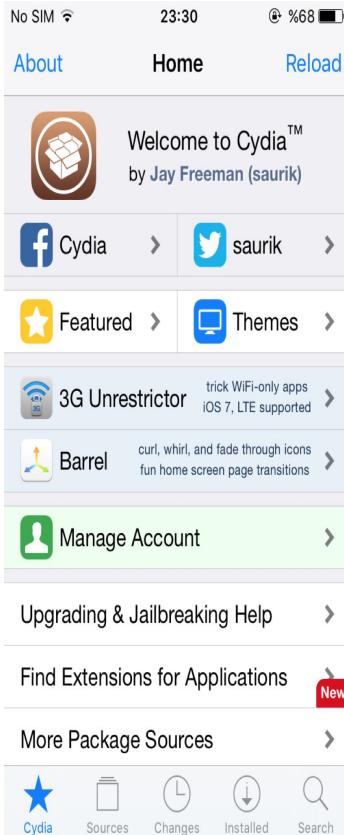


iOS (iPhone OS) SSH ile Bağlanma

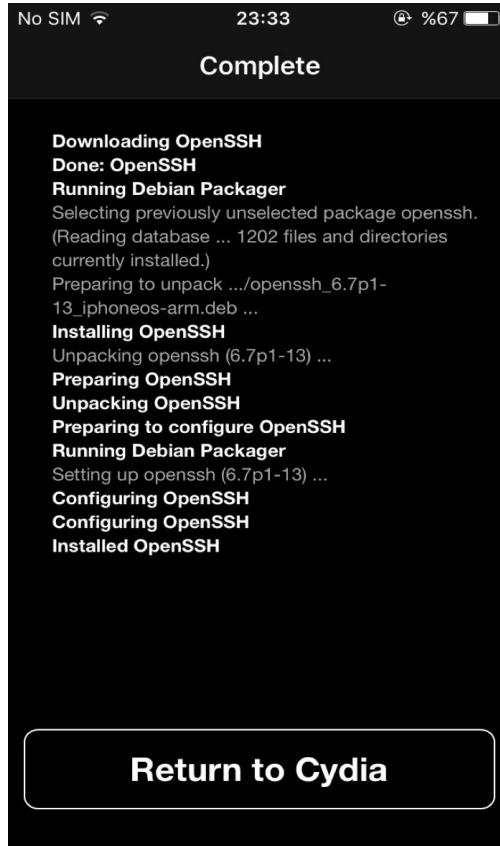
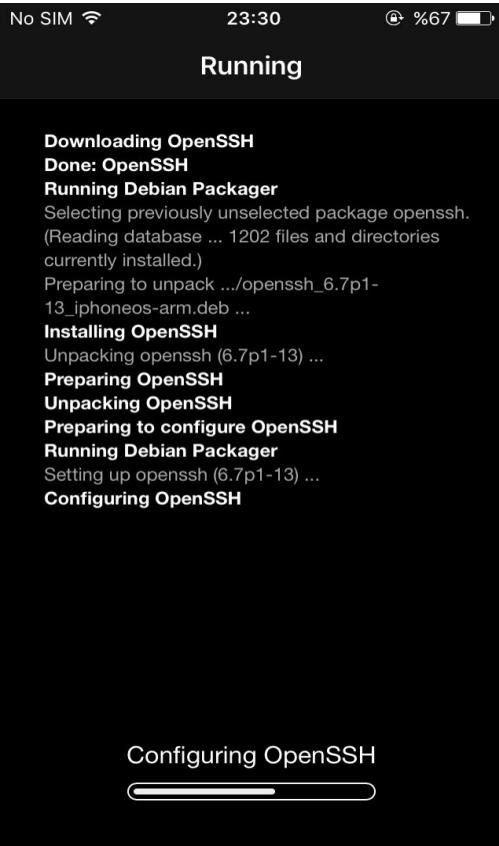
Aynı Ağ (Wi-Fi) Üzerinde Bulunan Cihaza SSH Bağlantısı Kurmak

iOS cihazımıza jailbreak yaparak root kullanıcı haklarında çalışmasını sağladık. Şimdi cihazımıza bağlanarak cihaz üzerinde komut çalıştırıp dosya ve dizinleri inceleyerek cihaz üzerinde ki testleri yapmamız gerekmektedir. Android dünyasında ADB ile bağlanırken bu iOS dünyasında SSH ile sağlanmaktadır. Jailbreak yapılan cihaza gelen Cydia uygulama marketi açılarak OpenSSH uygulaması kurulmalıdır.

iOS (iPhone OS) SSH ile Bağlanma



iOS (iPhone OS) SSH ile Bağlanma



iOS (iPhone OS) SSH ile Bağlanma

Jailbreak yapılmış iOS cihazımıza Cydia uygulama marketi üzerinden OpenSSH’ı başarılı bir şekilde kurduktan sonra hem bilgisayarımızı hemde iOS cihazımızı aynı ağ (Wi-Fi)’ye bağlıyoruz.

Daha sonra telefonumuzun IP adresine SSH bağlantısı yapacağız. Kurduğumuz OpenSSH uygulamasının varsayılan kullanıcı adı “**root**” parolası “**alpine**” dir.

iOS (iPhone OS) SSH ile Bağlanma

```
● ● ● ahmet — ssh root@192.168.1.31 — 120x30
+ ~ — ssh root@192.168.1.31

Last login: Tue Jul 10 17:42:28 on ttys001
[Ahmets-MacBook-Pro:~ ahmet$ ssh root@192.168.1.31]
The authenticity of host '192.168.1.31 (192.168.1.31)' can't be established.
RSA key fingerprint is SHA256:315oTqckg6BuENptIoLL0chIzyG+AaxP1/M+WudT1Q.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.31' (RSA) to the list of known hosts.
[root@192.168.1.31's password:
[iPhone-5:~ root# id
uid=0(root) gid=0(wheel) groups=0(wheel),1(daemon),2(kmem),3(sys),4(tty),5(operator),8(procview),9(procmod),20(staff),29(certusers),80(admin)
[iPhone-5:~ root# pwd
/var/root
[iPhone-5:~ root# ls
Library/ Media/
iPhone-5:~ root# ]
```

iOS (iPhone OS) SSH ile Bağlanma

Görüldüğü üzere **ssh root@IP_Adresi** olarak bağlantı sağlayarak parola kısmına “**alpine**” girdiğimizde cihazımıza başarılı bir şekilde bağlanıp root haklarıyla komut çalıştırabildiğimiz görülmektedir.

Aynı ağ üzerinden SSH bağlantısı yapıldığında bazı yavaşlıklar ve kopmalar olabilmektedir. Daha sağlıklı bir bağlantı için USB ile iOS cihazımızı bilgisayara bağlayarak SSH’ı USB kablo ile yapmak daha sağlıklı olacaktır.

iOS (iPhone OS) SSH ile Bağlanma

USB ile Bağlanılan Cihaza SSH Bağlantısı Kurmak

OpenSSH uygulaması cihaza kurduktan sonra USB ile iOS cihaza SSH bağlantısı yapabilmek için usbmuxd-1.0.8 aracını kullanacağız. 'usbmuxd', "USB multiplexing daemon" anlamına gelir. Bu daemon sorumlu USB üzerinden bir iPhone veya iPod touch'a çoklayıcı bağlantı kurmamıza yarar.

Kullanıcılara göre USB üzerinden müziğini, rehberinizi, fotoğraflarınızı vb. senkronize edebilirsiniz. Geliştiriciler ise cihazdaki herhangi bir dinleme localhost soketine bağlayabilme gibi özellikleri vardır. Usbmuxd aracını

<https://github.com/WildDylan/iOS-Reverse-Tools/tree/master/usbmuxd-1.0.8> adresinden indirebilirsiniz.

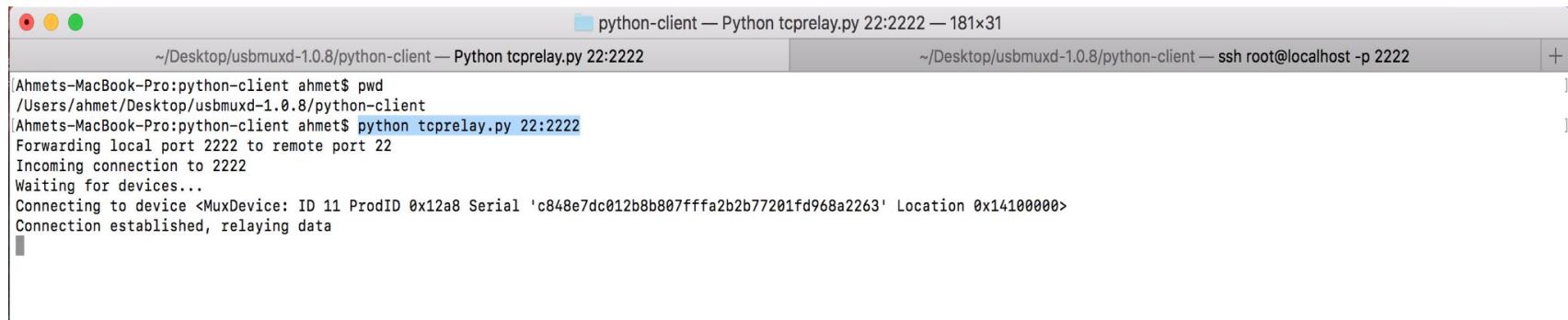
iOS (iPhone OS) SSH ile Bağlanma

The screenshot shows a GitHub repository page for 'iOS-Reverse-Tools/usbmuxd-1'. The repository has a single commit from WildDylan dated Dec 28, 2016, which committed all tools. The page lists various files and their commit details:

File	Commit Message	Time Ago
..	Commit all tools.	2 years ago
Modules	Commit all tools.	2 years ago
common	Commit all tools.	2 years ago
daemon	Commit all tools.	2 years ago
libusbmuxd	Commit all tools.	2 years ago
python-client	Commit all tools.	2 years ago
stuff	Commit all tools.	2 years ago
tools	Commit all tools.	2 years ago
udev	Commit all tools.	2 years ago
.gitattributes	Commit all tools.	2 years ago
.gitignore	Commit all tools.	2 years ago
AUTHORS	Commit all tools.	2 years ago
CMakeLists.txt	Commit all tools.	2 years ago
COPYING.GPLv2	Commit all tools.	2 years ago
COPYING.GPLv3	Commit all tools.	2 years ago
COPYING.LGPLv2.1	Commit all tools.	2 years ago
README	Commit all tools.	2 years ago
README.devel	Commit all tools.	2 years ago
libusbmuxd.pc.in	Commit all tools.	2 years ago

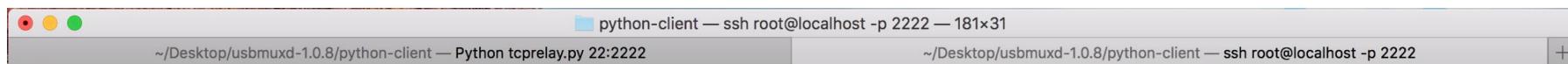
iOS (iPhone OS) SSH ile Bağlanma

Usbmuxd aracını indirdikten sonra python-client dizinine giderek burada ki tcprelay.py aracını çalıştırarak Görüldüğü üzere **python tcprelay.py 22:2222** ile port yönlendirmesi yapılmaktadır.



```
python-client — Python tcprelay.py 22:2222 — 181x31
~/Desktop/usbmuxd-1.0.8/python-client — Python tcprelay.py 22:2222
~/Desktop/usbmuxd-1.0.8/python-client — ssh root@localhost -p 2222 +
```

```
[Ahmet's-MacBook-Pro:python-client ahmet$ pwd
/Users/ahmet/Desktop/usbmuxd-1.0.8/python-client
[Ahmet's-MacBook-Pro:python-client ahmet$ python tcprelay.py 22:2222
Forwarding local port 2222 to remote port 22
Incoming connection to 2222
Waiting for devices...
Connecting to device <MuxDevice: ID 11 ProdID 0x12a8 Serial 'c848e7dc012b8b807ffffa2b2b77201fd968a2263' Location 0x14100000>
Connection established, relaying data]
```



```
python-client — ssh root@localhost -p 2222 — 181x31
~/Desktop/usbmuxd-1.0.8/python-client — Python tcprelay.py 22:2222
~/Desktop/usbmuxd-1.0.8/python-client — ssh root@localhost -p 2222 +
```

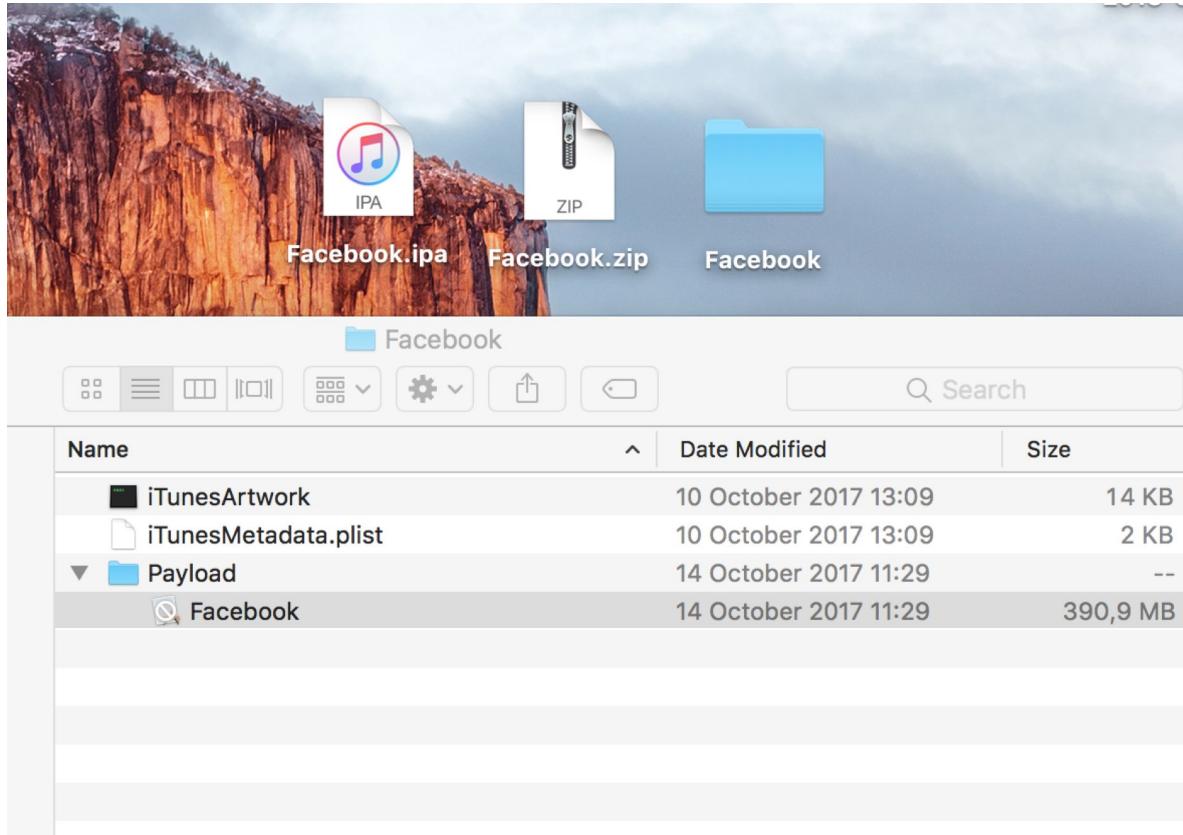
```
Last login: Tue Jul 10 23:50:53 on ttys000
[Ahmet's-MacBook-Pro:python-client ahmet$ ssh root@localhost -p 2222
The authenticity of host '[localhost]:2222 ([127.0.0.1]:2222)' can't be established.
RSA key fingerprint is SHA256:315oTqckg6BuENptIoL0chIzyG+AaxP1/M+WudT1Q.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[localhost]:2222' (RSA) to the list of known hosts.
root@localhost's password:
iPhone-5:~ root# id
uid=0(root) gid=0(wheel) groups=0(wheel),1(daemon),2(kmem),3(sys),4(tty),5(operator),8(procview),9(procmod),20(staff),29(certusers),80(admin)
iPhone-5:~ root#
```

iOS (iPhone OS) Dosya ve Dizin Yapısı

iOS dosya ve dizin yapısına geçmeden önce IPA (iOS App Store Package) dosya yapısına değinelim. IPA dosyaları üst bölmelerdede dejindiğimiz gibi iOS cihazlarda çalışan uygulamaların kurulum dosyalarıdır.

Aynı Android dünyasında ki APK dosyası gibidir. IPA dosya uzantısı aslında sıkıştırılmış bir arşiv dosyasıdır. Uzantı olarak .ipa olan dosya .zip olarak kaydedilip arşiv dosyaları ile açılabilir.

iOS (iPhone OS) Dosya ve Dizin Yapısı



iOS (iPhone OS) Dosya ve Dizin Yapısı

iTunesArtwork: Uygulama simgesini barındırır.

iTunesMetadata.plist: Uygulamaya ait bilgiler içerir.

Payload klasörü ise içerisinde Facebook.app dizini bulunur. Uygulamaya ait tüm dosyalar bu .app dizini içerisindeindedir. Bir IPA dosyasının içeriği bu şekilde incelenebilir.

IPA dosyaları market dışından iFunBox ile cihaz üzerine kurulumunu anlatmıştık.

Şimdi ise SSH ile bağlantı yapabildiğimiz iOS cihaza IPA dosyasının yüklenmesini yapacağız.

iOS (iPhone OS) Dosya ve Dizin Yapısı

```
[Ahmet's-MacBook-Pro:python-client ahmet$ scp -r /Users/ahmet/Desktop/Facebook/Payload/Facebook.app/ root@192.168.1.31:/Application
[root@192.168.1.31's password:
Localizable.json
AdsCountriesConfig.json
Localizable.strings
InfoPlist.strings
DateFormatConfig.json
NumberFormatConfig.json
AppIntentVocabulary.plist
CurrencyFormatConfig.json
CodeResources
ResourceRules
Default-fbapi~ipad.png
Localizable.json
AdsCountriesConfig.json
Localizable.strings
InfoPlist.strings
DateFormatConfig.json
NumberFormatConfig.json
Localizable.stringsdict
AppIntentVocabulary.plist
CurrencyFormatConfig.json
Localizable.json
AdsCountriesConfig.json
Localizable.strings
InfoPlist.strings
DateFormatConfig.json
NumberFormatConfig.json
AppIntentVocabulary.plist
CurrencyFormatConfig.json
Icon-Production20x20@2x-ipad.png
Localizable.json
AdsCountriesConfig.json
Localizable.strings
InfoPlist.strings
DateFormatConfig.json
NumberFormatConfig.json
AppIntentVocabulary.plist
CurrencyFormatConfig.json
Default-fbauth02x~iphone.png
Icon-Production29x29-ipad.png
Localizable.json
AdsCountriesConfig.json
Localizable.strings
InfoPlist.strings
DateFormatConfig.json
NumberFormatConfig.json
AppIntentVocabulary.plist
CurrencyFormatConfig.json
Default-fbauth~ipad.png
Localizable.json
AdsCountriesConfig.json
100% 233KB 1.7MB/s 00:00
100% 11KB 844.0KB/s 00:00
100% 64KB 2.3MB/s 00:00
100% 1997 381.9KB/s 00:00
100% 2340 435.8KB/s 00:00
100% 218 47.1KB/s 00:00
100% 329 15.2KB/s 00:00
100% 4077 419.3KB/s 00:00
100% 567KB 2.2MB/s 00:00
100% 6 0.8KB/s 00:00
100% 4152 495.5KB/s 00:00
100% 373KB 2.4MB/s 00:00
100% 14KB 1.4MB/s 00:00
100% 1045KB 2.2MB/s 00:00
100% 3415 224.1KB/s 00:00
100% 2449 321.6KB/s 00:00
100% 218 49.5KB/s 00:00
100% 1710 333.1KB/s 00:00
100% 371 82.3KB/s 00:00
100% 4747 661.6KB/s 00:00
100% 191KB 1.3MB/s 00:00
100% 12KB 1.1MB/s 00:00
100% 517KB 2.6MB/s 00:00
100% 1679 324.9KB/s 00:00
100% 3024 232.9KB/s 00:00
100% 218 49.0KB/s 00:00
100% 162 24.7KB/s 00:00
100% 3812 429.1KB/s 00:00
100% 1315 255.7KB/s 00:00
100% 258KB 2.8MB/s 00:00
100% 13KB 1.4MB/s 00:00
100% 725KB 2.9MB/s 00:00
100% 2212 274.2KB/s 00:00
100% 3001 418.4KB/s 00:00
100% 218 39.6KB/s 00:00
100% 227 49.5KB/s 00:00
100% 4190 550.2KB/s 00:00
100% 5854 693.3KB/s 00:00
100% 973 194.2KB/s 00:00
100% 80 8.5KB/s 00:00
100% 11KB 986.3KB/s 00:00
100% 64 13.2KB/s 00:00
100% 1690 318.2KB/s 00:00
100% 2446 296.1KB/s 00:00
100% 218 36.5KB/s 00:00
100% 290 70.6KB/s 00:00
100% 3824 517.9KB/s 00:00
100% 3882 669.9KB/s 00:00
100% 198KB 2.5MB/s 00:00
100% 12KB 1.3MB/s 00:00
```

iOS (iPhone OS) Dosya ve Dizin Yapısı

scp -r /Users/ahmet/Desktop/Facebook/Payload/Facebook.app/ root@192.168.1.31:/Application
komutu ile arşiv olarak kaydedilen IPA dosyasından elde edilen Payload dizini altında ki .app dosyası
scp ile ssh üzerinden cihazımızın Application dizinini kopyalanmaktadır.

iOS (iPhone OS) Dosya ve Dizin Yapısı

```
[Ahmet's-MacBook-Pro:python-client ahmet$ ssh root@192.168.1.31
[root@192.168.1.31's password:
[iPhone-5:~ root# cd /Application/Facebook.app/
[iPhone-5:/Application/Facebook.app root# ls
```

<code>APResources.bundle/</code>	<code>Default@2x~ipad.png*</code>	<code>Icon-Production40x40@2x~ipad.png*</code>	<code>el.lproj/</code>	<code>meetspinner.3f*</code>
<code>Assets.car*</code>	<code>Default@2x~iphone.png*</code>	<code>Icon-Production40x40@3x.png*</code>	<code>en-GB.lproj/</code>	<code>ms.lproj/</code>
<code>ComponentScriptBundle-packed.bcbundle*</code>	<code>Default-ipad.png*</code>	<code>Icon-Production40x40~ipad.png*</code>	<code>en.lproj/</code>	<code>nb.lproj/</code>
<code>ComponentScriptBundle.bcbundle*</code>	<code>FBAvatarEditorResources/</code>	<code>Icon-Production60x60@2x.png*</code>	<code>es-ES.lproj/</code>	<code>nl.lproj/</code>
<code>ComponentScriptBundle.js*</code>	<code>FBPrivacyModule.bundle/</code>	<code>Icon-Production60x60@3x.png*</code>	<code>es.lproj/</code>	<code>nuxes.plist*</code>
<code>Default-568h@2x~iphone.png*</code>	<code>FBVideoHomeKitResources/</code>	<code>Icon-Production76x76@2x~ipad.png*</code>	<code>fi.lproj/</code>	<code>pl.lproj/</code>
<code>Default-667h@2x.png*</code>	<code>FB_FBNP_min11p8db.caf*</code>	<code>Icon-Production76x76~ipad.png*</code>	<code>fr.lproj/</code>	<code>pt-PT.lproj/</code>
<code>Default-736h@3x.png*</code>	<code>Facebook</code>	<code>Icon-Production83.5x83.5@2x~ipad.png*</code>	<code>hi.lproj/</code>	<code>pt.lproj/</code>
<code>Default-812h@3x.png*</code>	<code>Facebook.crc*</code>	<code>Info.plist*</code>	<code>hr.lproj/</code>	<code>react_native_routes.json*</code>
<code>Default-Landscape@2x~ipad.png*</code>	<code>Frameworks/</code>	<code>MobileTopUpProducts.json*</code>	<code>hu.lproj/</code>	<code>ro.lproj/</code>
<code>Default-Landscape~ipad.png*</code>	<code>Ghay*</code>	<code>PkgInfo*</code>	<code>id.lproj/</code>	<code>ru.lproj/</code>
<code>Default-fbapi-568h@2x~iphone.png*</code>	<code>Icon-Production-120.png*</code>	<code>PlugIns/</code>	<code>it.lproj/</code>	<code>sk.lproj/</code>
<code>Default-fbapi@2x~ipad.png*</code>	<code>Icon-Production20x20@2x.png*</code>	<code>Settings.bundle/</code>	<code>ja.lproj/</code>	<code>sv.lproj/</code>
<code>Default-fbapi@2x~iphone.png*</code>	<code>Icon-Production20x20@2x~ipad.png*</code>	<code>Sys.dylib</code>	<code>ko.lproj/</code>	<code>th.lproj/</code>
<code>Default-fbapi@3x~iphone.png*</code>	<code>Icon-Production20x20@3x.png*</code>	<code>_CodeSignature/</code>	<code>libloader/</code>	<code>tr.lproj/</code>
<code>Default-fbapi-ipad.png*</code>	<code>Icon-Production20x20~ipad.png*</code>	<code>assets/</code>	<code>libuasharedanalytics.dylib</code>	<code>updateAvatarState.js*</code>
<code>Default-fbauth-568h@2x~iphone.png*</code>	<code>Icon-Production29x29@2x.png*</code>	<code>clash_units.plist*</code>	<code>libuasharedanalyticsflurry.dylib</code>	<code>vi.lproj/</code>
<code>Default-fbauth@2x~ipad.png*</code>	<code>Icon-Production29x29@2x~ipad.png*</code>	<code>cs.lproj/</code>	<code>libuasharedtools.bundle/</code>	<code>zh-Hans.lproj/</code>
<code>Default-fbauth@2x~iphone.png*</code>	<code>Icon-Production29x29@3x.png*</code>	<code>da.lproj/</code>	<code>libuasharedtools.dylib</code>	<code>zh-Hant-HK.lproj/</code>
<code>Default-fbauth@3x~iphone.png*</code>	<code>Icon-Production29x29~ipad.png*</code>	<code>de.lproj/</code>	<code>libuasharedtoolsgoogle.dylib</code>	<code>zh-Hant.lproj/</code>
<code>Default-fbauth-ipad.png*</code>	<code>Icon-Production40x40@2x.png*</code>	<code>diskstores.json*</code>	<code>main.jsbundle*</code>	

```
[iPhone-5:/Application/Facebook.app root# chmod +x Facebook
[iPhone-5:/Application/Facebook.app root# ./Facebook
Killed: 9
[iPhone-5:/Application/Facebook.app root# uicache
```

```
[iPhone-5:/Application/Facebook.app root#
```

iOS (iPhone OS) Dosya ve Dizin Yapısı

Kopyalama işlemi başarılı bir şekilde tamamlandıktan sonra Şekilde görüldüğü üzere cihaza SSH bağlantısı yapılarak bağlanıyoruz. Daha sonra **cd /Application/Facebook.app** ile kopyaladığımız dizini giderek **chmod +x Facebook** komutu ile çalışma yetkisi veriyoruz. Son olarakta **uicache** komutu ile işlemi tamamlıyoruz. iFunBox haricide IPA dosyası ile uygulama kurulumunu SSH üzerinden bu şekilde gerçekleştirebilirsiniz.

iOS (iPhone OS) Dosya ve Dizin Yapısı

```
~ — ssh root@192.168.1.31
```

```
Last login: Wed Jul 11 22:34:05 on ttys000
[Ahmets-MacBook-Pro:~ ahmet$ ssh root@192.168.1.31
[root@192.168.1.31's password:
[iPhone-5:~ root# pwd
/var/root
[iPhone-5:~ root# cd /
[iPhone-5:/ root# ls
Applications/ Library/ User@ boot/ dev/ lib/ private/ tmp@ var@
Developer/ System/ bin/ cores/ etc@ mnt/ sbin/ usr/
[iPhone-5:/ root# cd private/
[iPhone-5:/private root# ls
etc/ system_data/ var/
[iPhone-5:/private root# cd var/
[iPhone-5:/private/var root# ls
Keychains/ MobileSoftwareUpdate/ db/ keybags/ log/ networkd/ spool/ wireless/
Managed\ Preferences/ backups/ empty/ lib/ logs/ preferences/ stash@
MobileAsset/ cache/ folders/ local/ mobile/ root/ tmp/
MobileDevice/ containers/ installd/ lock/ msgs/ run/ vm/
```

iOS (iPhone OS) Dosya ve Dizin Yapısı

Görüldüğü üzere SSH üzerinden Jailbreak yapılmış cihaza bağlanıldığında /var/root dizininde olduğu görülmektedir. Ls komutu ile dizinler listelendiğinde Application, Developer, Library ve private gibi bir çok dizin görülmektedir. Cihaz root haklarında çalıştığı için tüm dosya ve dizinlere erişilebilmektedir.

iOS (iPhone OS) Dosya ve Dizin Yapısı

```
~ — ssh root@192.168.1.31
[ iPhone-5:/ root# ls
 Applications/ Library/ User@ boot/ dev/ lib/ private/ tmp@ var@
 Developer/ System/ bin/ cores/ etc@ mnt/ sbin/ usr/
 [ iPhone-5:/ root# cd Applications/
 [ iPhone-5:/Applications root# ls
 AACredentialRecoveryDialog.app/ Home.app/ ScreenSharingViewService.app/
 AccountAuthenticationDialog.app/ HomeUIService.app/ ServerDocuments.app/
 AdSheet.app/ InCallService.app/ Setup.app/
 AppStore.app/ Magnifier.app/ SharedWebCredentialViewService.app/
 AskPermissionUI.app/ MailCompositionService.app/ SharingViewService.app/
 Bridge.app/ Maps.app/ SiriViewService.app/
 Calculator.app/ MessagesNotificationViewService.app/ SocialUIService.app/
 Camera.app/ MessagesViewService.app/ SoftwareUpdateUIService.app/
 CheckerBoard.app/ MobileCal.app/ Stocks.app/
 Compass.app/ MobileMail.app/ StoreDemoViewService.app/
 CompassCalibrationViewService.app/ MobileNotes.app/ StoreKitUIService.app/
 Contacts.app/ MobilePhone.app/ TV.app/
 CoreAuthUI.app/ MobileSMS.app/ TencentWeiboAccountMigrationDialog.app/
 Cydia.app/ MobileSafari.app/ Tips.app/
 DDActionsService.app/ MobileSlideShow.app/ TrustMe.app/
 DataActivation.app/ MobileStore.app/ Utilities/
 DemoApp.app/ MobileTimer.app/ VideoSubscriberAccountViewService.app/
 Diagnostics.app/ Music.app/ Videos.app/
 DiagnosticsService.app/ MusicUIService.app/ VoiceMemos.app/
 FaceTime.app/ News.app/ WatchListViewService.app/
 FacebookAccountMigrationDialog.app/ Passbook.app/ Weather.app/
 Family.app/ PassbookUIService.app/ Web.app/
 Feedback\ Assistant\ iOS.app/ PhotosViewService.app/ WebApp1.app/
 FieldTest.app/ Podcasts.app/ WebContentAnalysisUI.app/
 FindMyFriends.app/ PreBoard.app/ WebSheet.app/
 FindMyiPhone.app/ Preferences.app/ iAdOptOut.app/
 Fitness.app/ Print\ Center.app/ iBooks.app/
 GameCenterUIService.app/ Reminders.app/ iCloud.app/
 HashtagImages.app/ SLGoogleAuth.app/ iCloudDriveApp.app/
 Health.app/ SLYahooAuth.app/ SafariViewService.app/
 HealthPrivacyService.app/
 iPhone-5:/Applications root# ]
```

iOS (iPhone OS) Dosya ve Dizin Yapısı

System Partition /Applications dizinidir. Temel OS uygulama binary'leri, işletim Sistemi ve işletim sistemi uygulamaları bulunur. Read Only partition dır.

```
~ — ssh root@192.168.1.31
iPhone-5:/private/var/mobile/Containers/Data/Application root# pwd
/private/var/mobile/Containers/Data/Application
iPhone-5:/private/var/mobile/Containers/Data/Application root# ls -la
total 0
drwxr-xr-x  56 mobile  mobile  1904 Jul 11 00:42 .
drwxr-xr-x  8 root   mobile  272 Jan  6 2018 ..
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 013DE3C1-822B-4775-9747-CCCDE5482ABD/
drwxr-xr-x  6 mobile  mobile  238 Jul 10 11:50 01EDDEB4A-FB4F-4A70-9DED-97C86D2FE4F1/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 052C6C08-E909-4763-9B1E-96D322DFA331/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 1BC9C974-1117-4943-AFC0-ECEB944E6EC8/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 206F3234-AC6A-4AB1-A77A-299B0D77E669/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 21BD1C80-FCB2-4460-8E5A-A7A1014A77DD/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 236868A1-8C1B-4C15-BC82-3B91049645C5/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 2C8BF8A3-8027-4FC1-8EF1-C148810C4EE6/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 3593864B-A241-48F1-8003-8388BEFB2F31/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 3B7A9F2F-D3F1-41B9-B293-BF320C6E6606/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 3CCS56C03-E4C0-4CE5-ABE5-89D88977EA03/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 3D63871D-74B8-491C-976B-1485D36E10D9/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 4F9E117C-7289-4DE3-8C28-011BFCB18D15/
drwxr-xr-x  5 mobile  mobile  204 Jul 11 00:42 52F3E883-CB11-45EE-B786-742FADA38A10/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 53D88BC64-C087-49EC-BB80-502C82D998BDF/
drwxr-xr-x  6 mobile  mobile  238 Jul 10 17:39 576D26FE-0764-4F1A-9093-58B96608CA41C/
drwxr-xr-x  6 mobile  mobile  238 Mar  1 15:29 593C33F2-1E26-4617-95F9-C91862542905/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 64471C58-639A-4A9C-90C9-6BF142F325E0/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 65B9729A-26A4-4095-ADA2-9639EB9595D9/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 6C4D52FB-1C11-4963-BDA4-1C387614DD6E/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 7023EFB7-5809-45E1-87D4-89B471887EBB/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 73EBC8A1-56FC-450B-8430-E28000E44072/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 85950178-F28D-44C8-9CF4-7F5198700986/
drwxr-xr-x  5 mobile  mobile  204 Jul 10 15:54 85B1E4BE-33BA-4DCC-86BA-38AB4C1BF030/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 897C9F9C-2D48-4038-8704-7816F2E96C0C/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 904EFD7F-2790-4517-8B73-99FA0ECD7005/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 915B2C19-3799-410F-8F88-04DEE3EBFB87/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 9B9B0DC1-F2EE-47DB-BDD1-FB1E80FB7033/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 A0E2786A-002F-4DB3-ABD8-FA75E0705EC5/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 A179B3CB-E669-47FF-8074-AB71FB4BC572/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 A38C04D4-7ABA-481D-9B00-DB41421AF42C/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 A49C0A1D-0949-450E-90F0-23AB6697BFC4/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 A78DF6C7-EADA-40C1-9EBD-A02B27278F09/
drwxr-xr-x  5 mobile  mobile  204 Jul 10 16:45 AD623F51-BFB3-449F-AC39-14FA7E77D7C7/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 AF2EACB7-C2BF-45E8-91B3-ABCC69F8A08C/
drwxr-xr-x  5 mobile  mobile  204 Mar  1 08:19 B76CCE12-8659-48C2-9666-DE8AB2FA6DC5/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 BAE89FBB-BAC6-45D7-B13D-51D4C557FDE0/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 C0502DC9-C7FC-4F69-949E-00D7F93562EB/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 C0643592-F96E-43FF-8BC1-15B6231BAE38/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 C6A33939-3718-4C4B-B500-CBBD101BC916/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 D84429F5-FF3D-4123-9BA2-674690C7ED20/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 DB0CAEA2-93C1-44DD-AE13-AB7C67B1A1B6/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 DC3C4AC7-4F18-4F43-AC43-C52E4F763837/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 DE9ED2B8-B54F-43A2-A8E7-7A7A655067DE/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 EB25CCC1-9F4A-4570-A1C4-4FF0694387DD/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 EC6167B1-5CDF-4C87-8C6E-1FEE377F1B9A/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 ECBEBA466-611B-477D-912F-69D6A2B5E962/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 EE9D40E8-C581-41C2-AC10-F29C069CA452/
drwxr-xr-x  5 mobile  mobile  204 Jan  6 2018 EEDC084C-73C4-466D-9EDE-FF7702C108D7/
```

iOS (iPhone OS) Dosya ve Dizin Yapısı

User Partition Private/var dizinidir. App dizinleri benzersiz (unique) klasör isimleri verilmektedir. Tüm 3. party app binary'leri **/private/var/mobile/Containers/Dara/Application** dizininde tutulur.

~ — ssh root@192.168.1.31

```
iPhone-5:/private/var/mobile/Containers/Data/Application root# find /private/var/mobile/Containers/Data/Application/ -name *mail*
/private/var/mobile/Containers/Data/Application/3B7A0F2F-D3F1-41B9-B293-BF320C6E6606/Library/Caches/Snapshots/com.apple.mobilemail
/private/var/mobile/Containers/Data/Application/3B7A0F2F-D3F1-41B9-B293-BF320C6E6606/Library/Caches/com.apple.mobilemail
/private/var/mobile/Containers/Data/Application/3B7A0F2F-D3F1-41B9-B293-BF320C6E6606/Library/Preferences/com.apple.mobilemail.plist
/private/var/mobile/Containers/Data/Application/3B7A0F2F-D3F1-41B9-B293-BF320C6E6606/Library/SyncedPreferences/com.apple.mobilemail-com.apple.mail.conversationRelevanceRule.plist
/private/var/mobile/Containers/Data/Application/3B7A0F2F-D3F1-41B9-B293-BF320C6E6606/Library/SyncedPreferences/com.apple.mobilemail-com.apple.mail.favorites.plist
/private/var/mobile/Containers/Data/Application/3B7A0F2F-D3F1-41B9-B293-BF320C6E6606/Library/SyncedPreferences/com.apple.mobilemail-com.apple.mail.listUnsubscribeInfo.plist
/private/var/mobile/Containers/Data/Application/3B7A0F2F-D3F1-41B9-B293-BF320C6E6606/Library/SyncedPreferences/com.apple.mobilemail-com.apple.mail.senderRelevanceRule.plist
/private/var/mobile/Containers/Data/Application/3B7A0F2F-D3F1-41B9-B293-BF320C6E6606/Library/SyncedPreferences/com.apple.mobilemail-com.apple.mail.threadinfo.plist
/private/var/mobile/Containers/Data/Application/3B7A0F2F-D3F1-41B9-B293-BF320C6E6606/Library/SyncedPreferences/com.apple.mobilemail-com.apple.mail.vipsenders.plist
/private/var/mobile/Containers/Data/Application/3B7A0F2F-D3F1-41B9-B293-BF320C6E6606/Library/SyncedPreferences/com.apple.mobilemail.plist
iPhone-5:/private/var/mobile/Containers/Data/Application root#
```

iOS (iPhone OS) Dosya ve Dizin Yapısı

find /private/var/mobile/Containers/Data/Application/ -name *Aranacak_Uygulama* komutu ile Application dizini altında bulunan UDID dizin isimlerinden anlaşılamayan hangi uygulamanın hangi dizinde olduğu tespit edilebilmektedir. Yüklenen uygulamanın cihaz üzerinde tutulan verilerini ve dosyalarını bu şekilde bularak incelenebilmektedir.

Cihaz üzerine bağlanarak uygulama dosyası tespit edildikten sonra xml, json, txt ve veritabanı dosyalarına hassas veri, güvenlik açığı oluşturabilecek bilgiler Android üzerinde anlatılan teknikler ile aynı şekilde incelenmelidir.

Uygulama dosyası üzerinde bulunan sqlite3 veritabanı dosyaları, local dosyalar, log dosyaları gibi dosyalar kontrol edilmelidir. Android dünyasından farklı olarak Keychain ve Property List (.plist) dosyaları incelenmelidir. Keychain de kayıt ulaşılabilen bir kimlik bilgisi var mı?

Bunun için cihaza SSH ile bağlanılarak <http://github.com/ptoomey3/Keychain-Dumper/archive/master.zip> adresindeki Keychain-Dumper isimli araç cihaza wget ile indirilerek zip dosyasından çıkartılarak çalıştırılarak Keychain verileri incenebilir. Property List (.plist) dosyalarında hassas veriler açık bir şekilde tutuluyor mu ayrıca bunlarda kontrol edilmelidir.

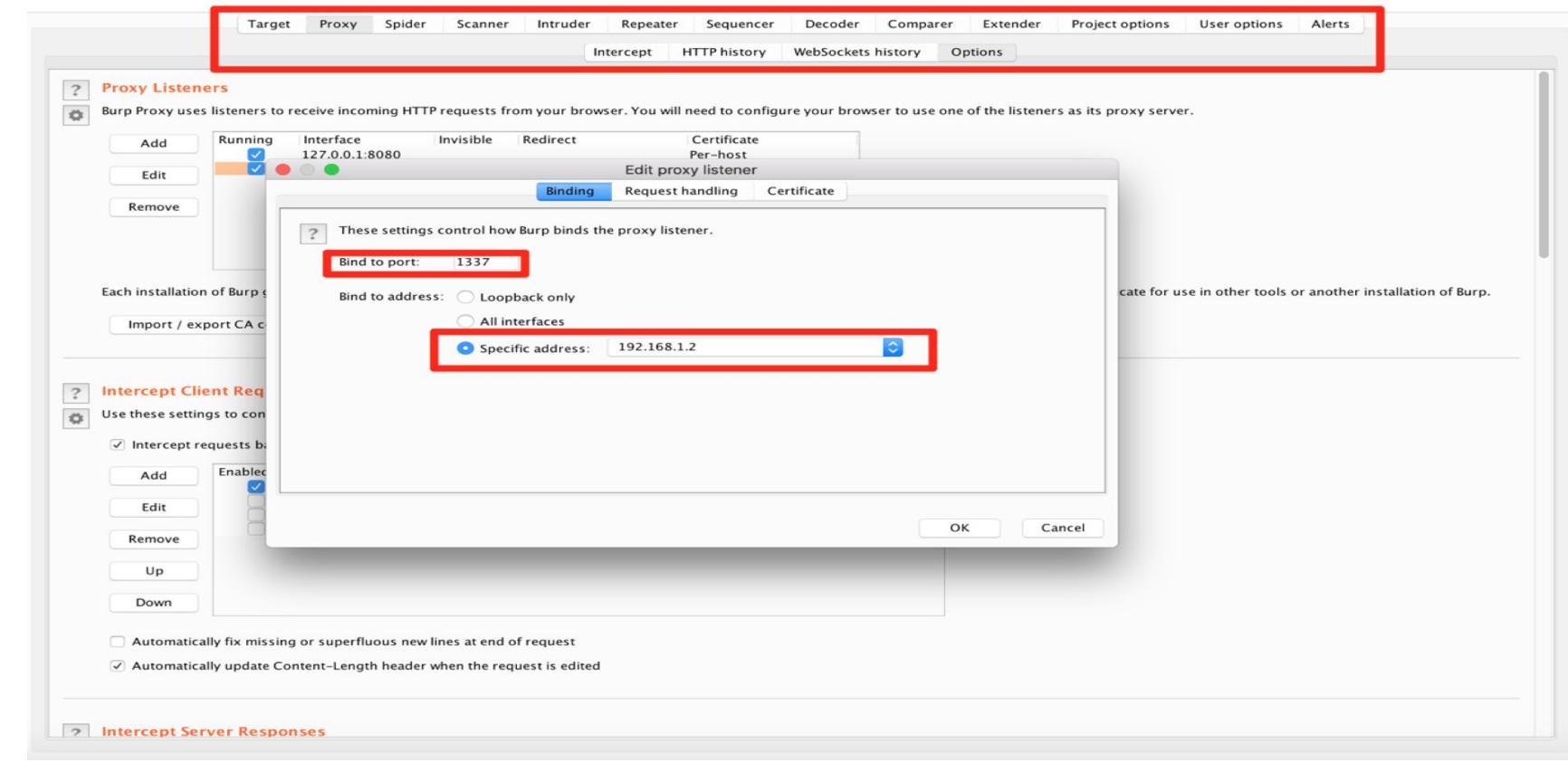
iOS (iPhone OS) BurpSuite Bağlantısı ve Sertifika Yükleme

Android cihazlar için Burp Suite proxy yazılımı üzerinden cihaz üzerindeki trafiği dinlemiştik. Gerekli bağlantıları yapmış ve android cihaza burp suite yazılımının kendi sertifikasını kurmuştuk.

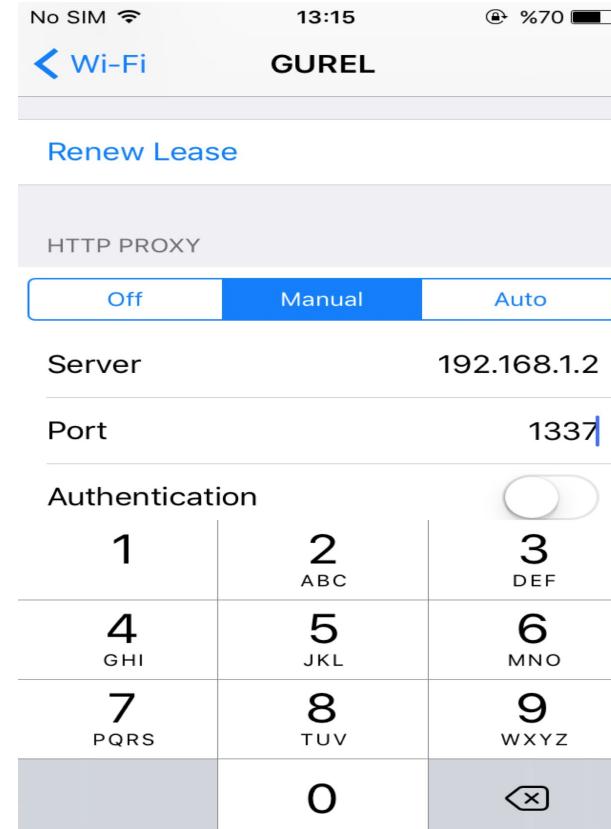
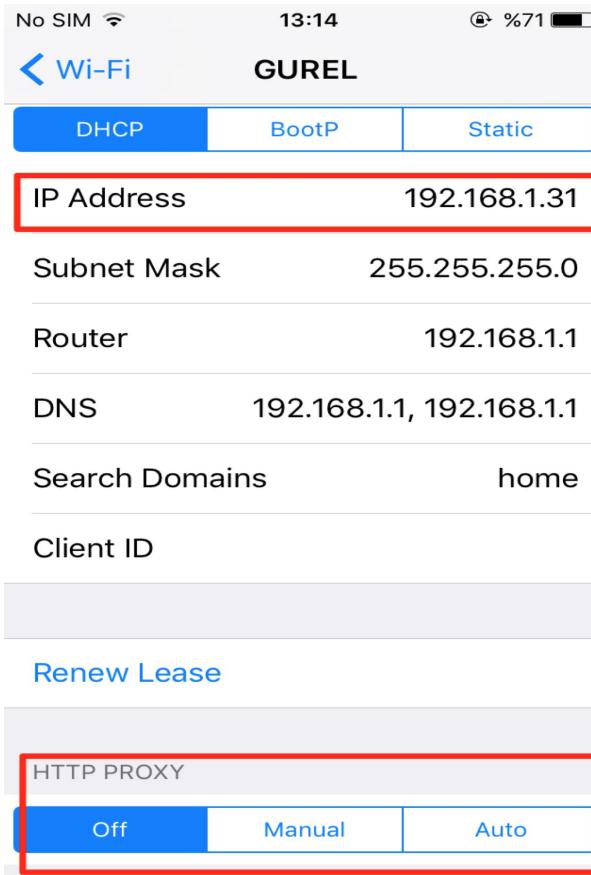
Aynı işlemi şu an iOS cihazlar için gerçekleştireceğiz. İlk olarak daha önceki bölümlerde detaylı olarak anlattığımız Burp Suite yazılımını indirerek başlıyoruz.

Burp Suite yazılımını <https://portswigger.net/burp> adresinden ücretsiz olarak indirebilirsiniz. İndirdikten sonra yazılımımızı açarak Proxy sekmesi üzerinden Options'a gelerek Add butonuna basarak IP adresi ve port bilgisini giriyoruz ve Ok butonuna basarak bu işlemi tamamlıyoruz. Unutulmamalıdır ki bilgisayarımı ve iOS cihazımız aynı ağa bağlı olmalıdır.

iOS (iPhone OS) BurpSuite Bağlantısı ve Sertifika Yükleme



iOS (iPhone OS) BurpSuite Bağlantısı ve Sertifika Yükleme

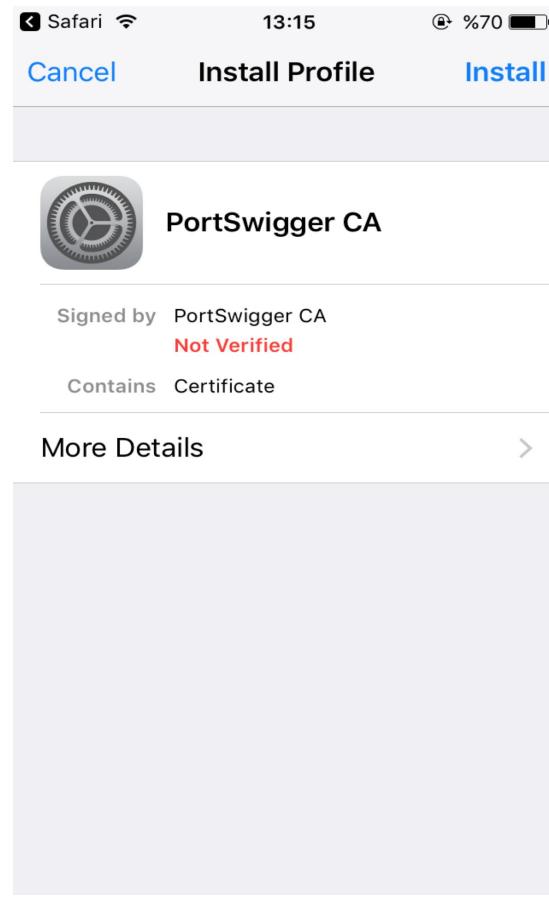
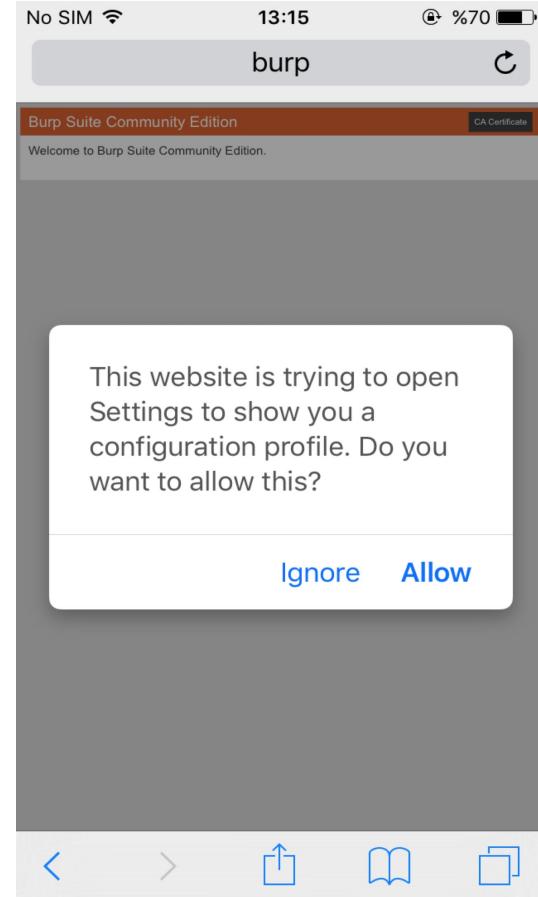
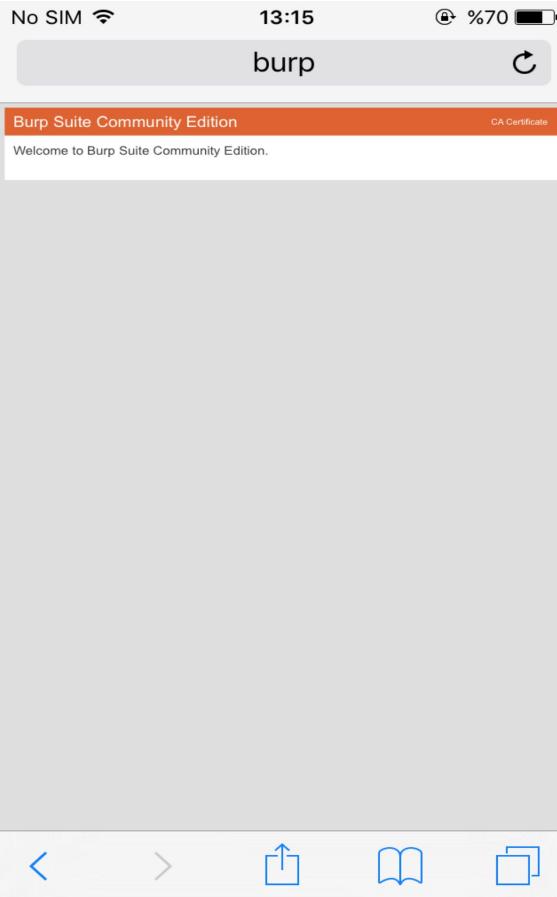


iOS (iPhone OS) BurpSuite Bağlantısı ve Sertifika Yükleme

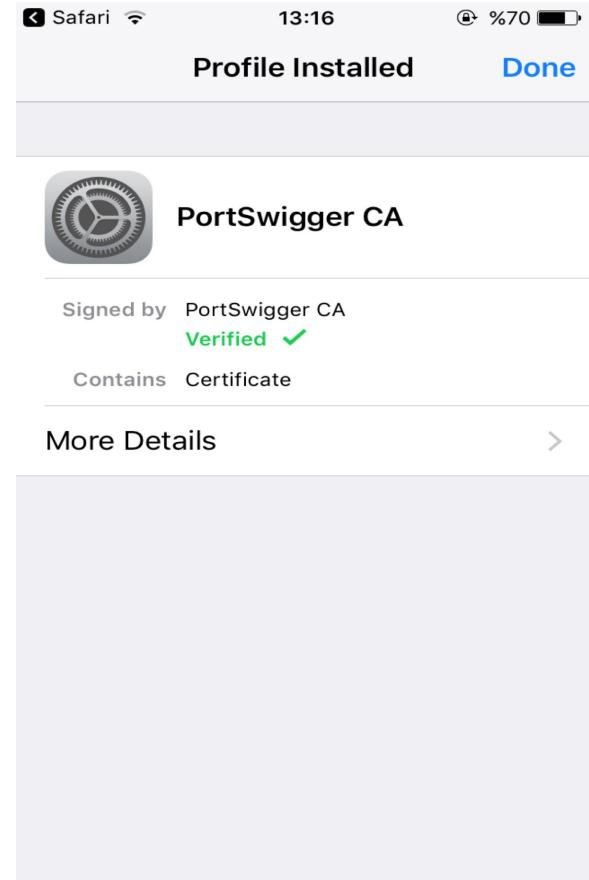
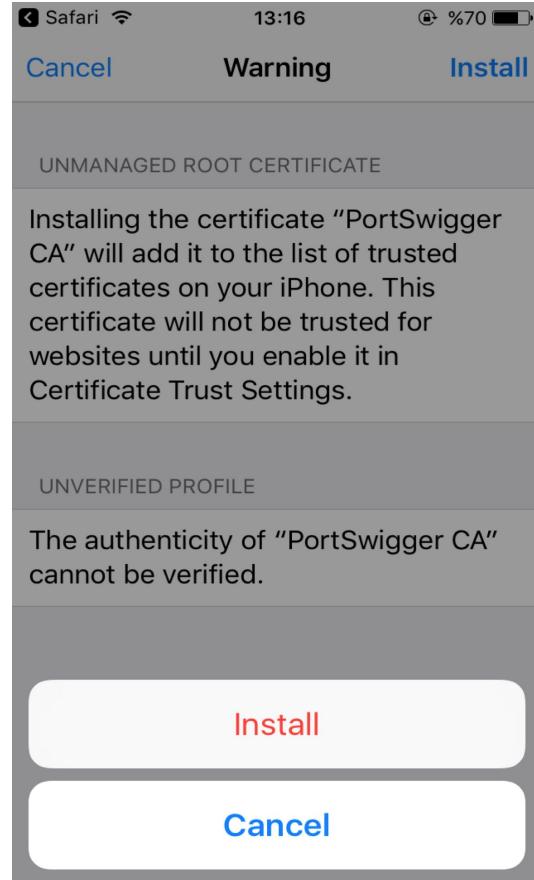
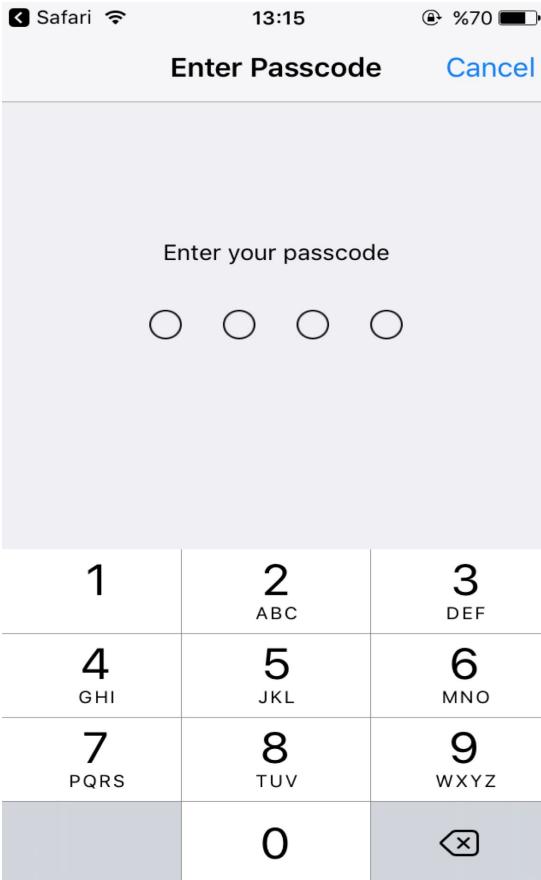
Settings -> Wi-Fi tıklanarak bağlı olunan ağ ayarları gelmektedir. Üst kısımdan iOS cihazımızın ip adresini öğrenebilirsiniz.

Alt tarafta bulunan HTTP Proxy yazan kısımdan Off durumunda Manual butonuna tıklayarak dinleme moduna aldığımız IP adresi ve port bilgisini gireceğiz.

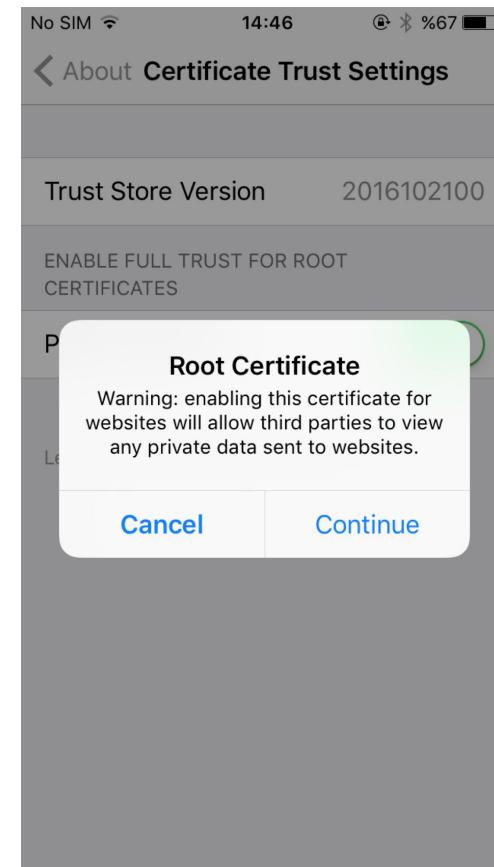
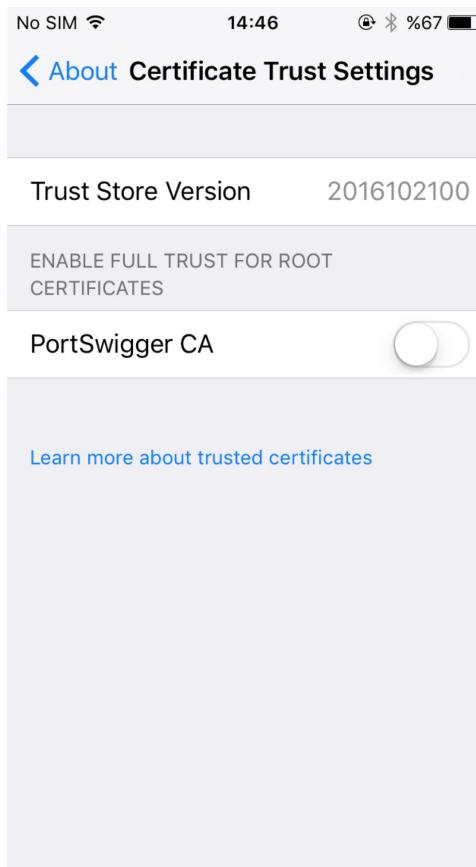
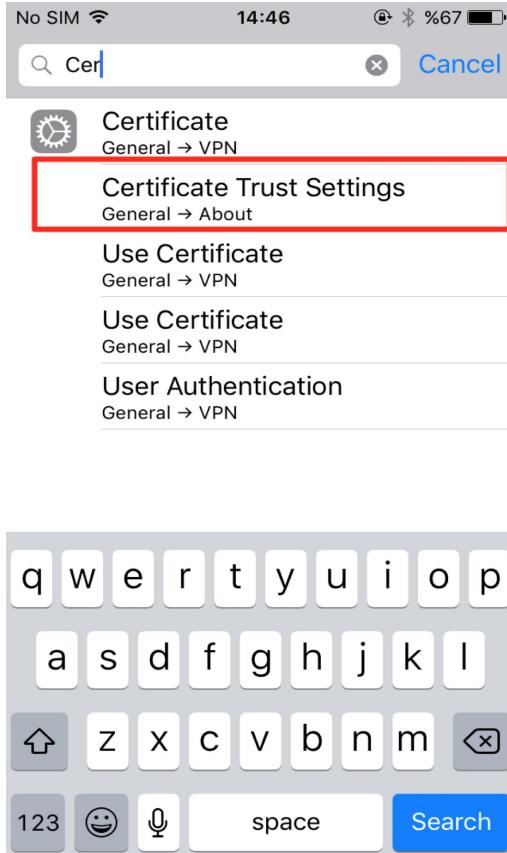
iOS (iPhone OS) BurpSuite Bağlantısı ve Sertifika Yükleme



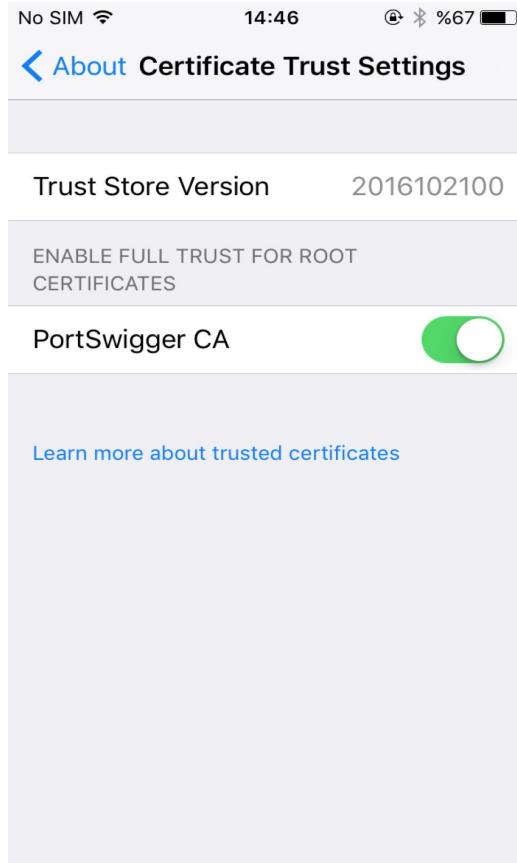
iOS (iPhone OS) BurpSuite Bağlantısı ve Sertifika Yükleme



iOS (iPhone OS) BurpSuite Bağlantısı ve Sertifika Yükleme



iOS (iPhone OS) BurpSuite Bağlantısı ve Sertifika Yükleme



iOS (iPhone OS) BurpSuite Bağlantısı ve Sertifika Yükleme

Mobil tarayıcıdan yapılan ahmet gruel araması Burp Suite aracında görülmektedir. Ayarlar tamamlanıp sertifika başarılı bir şekilde yüklendiğinde artık iOS cihazımızın trafiği Burp Suite üzerinde HTTP History kısmında görüntülenecektir.

Mobil uygulamaların haberleştiği web servis adresleri, istekleri tamamını buradan ulaşip Web Güvenliği ve Android Güvenliği kısmında anlattığımız testler iOS uygulama güvenliği içinde tekrarlanabilir.

Eğer yüklenen uygulamada SSL Pinning var ise bu durumda istekler Burp Suite üzerinde görünmeyecektir. Bunun içinde Android sistemlerde yaptığımız gibi iOS ortamındada SSL Pinning atlatma yöntemleri mevcuttur.

SSL Kill Switch uygulaması ile SSL Pinning atlatacağız.

iOS (iPhone OS) BurpSuite Bağlantısı ve Sertifika Yükleme

Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts

Intercept **HTTP history** WebSockets history Options

Filter: Hiding CSS, image and general binary content

#	Host	Method	URL	Params	Edited	Status	Length	MIME type	Extension	Title	Comment	SSL	IP
50	https://p52-keyvalueservice...	POST	/sync	✓		200	1783	XML			✓	17.248.147.10	
49	https://p52-keyvalueservice...	POST	/sync	✓		200	1515	XML			✓	17.248.147.10	
48	https://p52-keyvalueservice...	POST	/sync	✓		200	1783	XML			✓	17.248.147.14	
40	https://www.google.com.tr	GET	/search?q=ahmet+gurel&gbv=1&se...	✓		200	37936	HTML		ahmet gurel - Google'd...	✓	216.58.212.3	
39	https://www.google.com.tr	GET	/search?source=hp&ei=CxxIW-jYN4...	✓		200	227286	HTML			✓	216.58.212.3	
38	https://p52-keyvalueservice...	POST	/sync	✓		200	1515	XML			✓	17.248.147.14	
37	https://p52-keyvalueservice...	POST	/sync	✓		200	1783	XML			✓	17.248.147.14	
36	https://p52-keyvalueservice...	POST	/sync	✓		200	1515	XML			✓	17.248.147.14	
35	https://p52-sharedstreams.ic...	POST	/11958080182/sharedstreams/getc...	✓		200	1280	XML			✓	17.248.147.54	
34	https://itunes.apple.com	GET	/WebObjects/MZStore.woa/wa/com....			200	3667532	XML			✓	104.86.229.18	
33	https://itunes.com	GET	/version			302	749	HTML		302 Found		✓	17.178.96.29

Request Response

Raw Params Headers Hex

```
GET /search?q=ahmet+gurel&gbv=1&sei=IIpIW66DMIPL6ATzwLPACQ HTTP/1.1
Host: www.google.com.tr
Referer: https://www.google.com.tr/
Accept-Encoding: gzip, deflate
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Cookie: CGIC=Ij90ZXh0L2hbWwSYXbwGljYXRpb24veGh0bWwreGlsLGFWcGxy2F0aW9uL3htbDtxPTAuOSwqLy07cT0wLjg; 1P_JAR=2018-07-13-11;
ANID=AHWqTUuCM_ZYWoO-T602NmnhTCkqEOKES5VUyEry9NPY9Z_Q3z29sofC5U99;
NID=134=yNb6oJVPO9aNBm-xE0qum8M0aOT79rUxlebDoakKkqQTf5EVhkLlouDXRtgeeBzsHBRGwRo7jdEJc8eCs6Bp8IjrWxCJHsRJhUlkerj2PEmLQGZAsYus758oRRgbmKooTvForH10
8KdB9TL9-2Sk7chAsA41CH4ZHt-p8hwSKylsM9xK58i9MSRmFxpi4EyXkz8rPg
Accept-Language: en-us
Connection: close
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 10_3_3 like Mac OS X) AppleWebKit/603.3.8 (KHTML, like Gecko) Version/10.0 Mobile/14G60
Safari/602.1
```

? < + > Type a search term 0 matches

iOS (iPhone OS) SSL Kill Switch 2 ile SSL Pinning Atlatma

Android uygulama güvenliği kısmında SSL Sabitleme (Pinning) güvenlik önlemini atlatmak için XPosed modüllerini kullanmıştık.

iOS platformunda Jailbreak yapılmış cihaza SSL Kill Switch uygulaması SSH üzerinden kurularak uygulamalarda bulunan

SSL Sabitleme (Pinning) güvenlik önlemi atlatılabilirmektedir. SSL Kill Switch 2 uygulaması github üzerinden indirilebilir.

iOS (iPhone OS) SSL Kill Switch 2 ile SSL Pinning Atlatma

The screenshot shows a GitHub repository page for 'nabla-c0d3 / ssl-kill-switch2'. The repository is described as a 'Blackbox tool to disable SSL certificate validation - including certificate pinning - within iOS and OS X Apps'. It has 47 commits, 2 branches, 6 releases, and 5 contributors. The latest commit is from April 13. The commit history lists various changes such as fixing package names, updating project settings, and adding license files. The repository has 67 stars, 767 forks, and 142 issues.

nabla-c0d3 / ssl-kill-switch2

Blackbox tool to disable SSL certificate validation - including certificate pinning - within iOS and OS X Apps

47 commits | 2 branches | 6 releases | 5 contributors

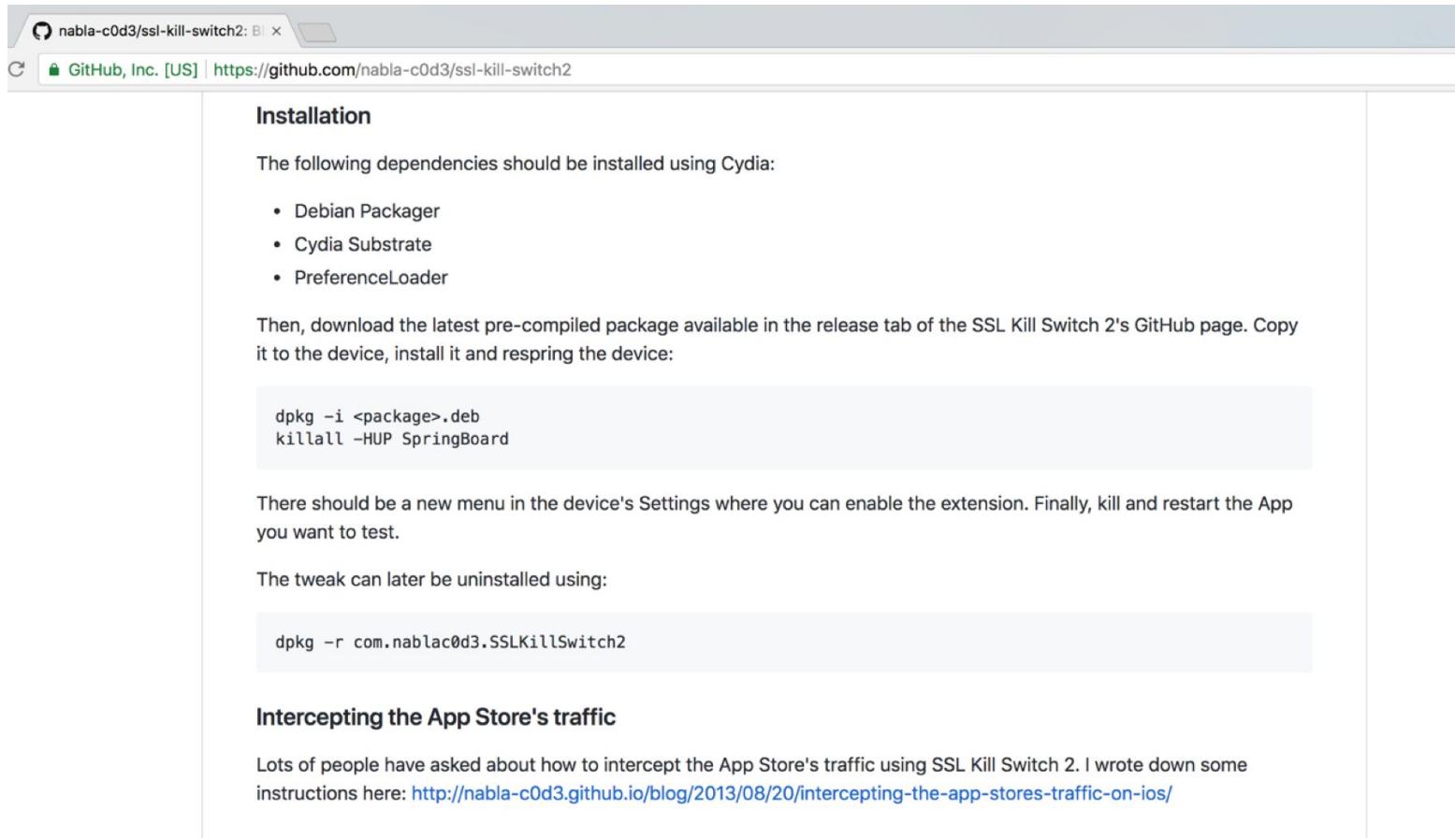
Branch: master | New pull request | Create new file | Upload files | Find file | Clone or download

Latest commit 4f79d0f on Apr 13

Commit	Description	Time Ago
SSLKillSwitch.xcodeproj	Update project settings	3 months ago
SSLKillSwitch	Tweak comments	3 months ago
SSLKillSwitchTests	Update project settings	3 months ago
layout	Fix package name for new theos requirements	3 months ago
.gitignore	Ignore theos build files	3 years ago
.travis.yml	Fix travis	3 months ago
BH2012_MobileCertificatePinning.pdf	Add BH presentation	3 years ago
LICENSE	Add license	3 years ago
Makefile	Force a device respring after install	3 years ago
README.md	Fix version number	3 months ago
SSLKillSwitch2.plist	Add iTunes daemon to bundle filter	2 years ago

README.md

iOS (iPhone OS) SSL Kill Switch 2 ile SSL Pinning Atlatma



The following dependencies should be installed using Cydia:

- Debian Packager
- Cydia Substrate
- PreferenceLoader

Then, download the latest pre-compiled package available in the release tab of the SSL Kill Switch 2's GitHub page. Copy it to the device, install it and respring the device:

```
dpkg -i <package>.deb  
killall -HUP SpringBoard
```

There should be a new menu in the device's Settings where you can enable the extension. Finally, kill and restart the App you want to test.

The tweak can later be uninstalled using:

```
dpkg -r com.nablaC0d3.SSLKillSwitch2
```

Intercepting the App Store's traffic

Lots of people have asked about how to intercept the App Store's traffic using SSL Kill Switch 2. I wrote down some instructions here: <http://nabla-c0d3.github.io/blog/2013/08/20/intercepting-the-app-stores-traffic-on-ios/>

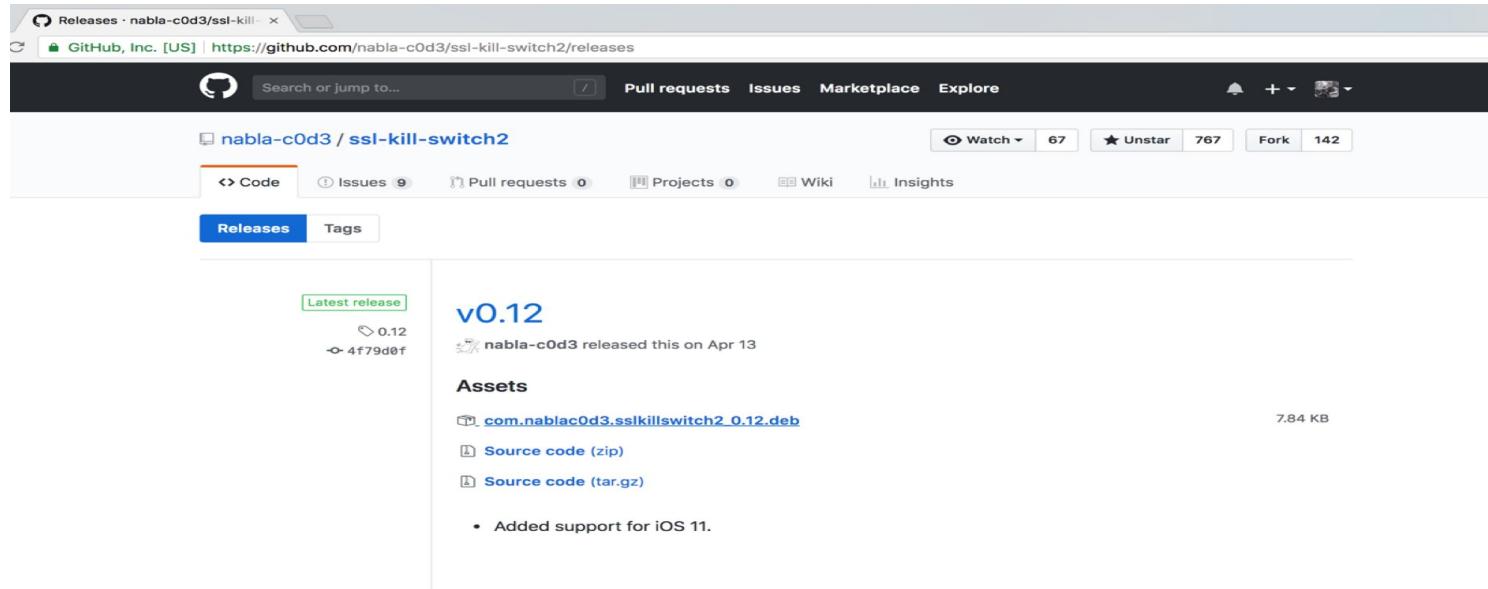
iOS (iPhone OS) SSL Kill Switch 2 ile SSL Pinning Atlatma

Kurulum sayfasında uygulamayı kurmadan önce kurulması gereken uygulamalar bulunmaktadır. iOS SSL Kill Switch yükleyebilmek için kurulması gereken üç araç vardır.

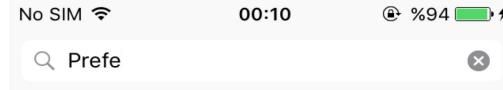
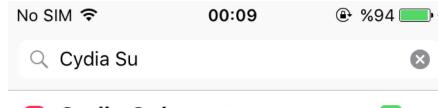
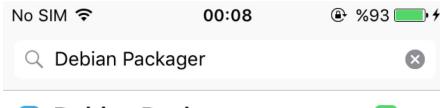
Dpkg: .deb paketlerini yüklemek için,

Cydia Substrate: Fonksiyonları patch'lemek için,

PreferenceLoader: iOS Ayarlar menüsüne ekleme yapabilmek için gereklidir.



iOS (iPhone OS) SSL Kill Switch 2 ile SSL Pinning Atlatma



Preference Tag Clear Theme

Preference WithMe

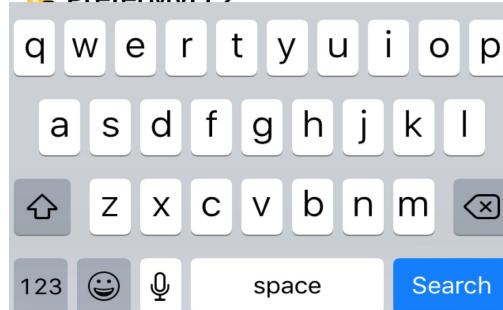
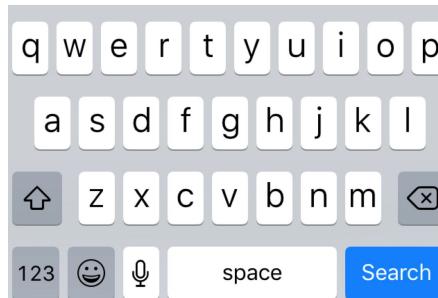
PreferenceLoader ✓

PreferenceTag (iOS 7)

PreferenceTag2 (iOS 8)

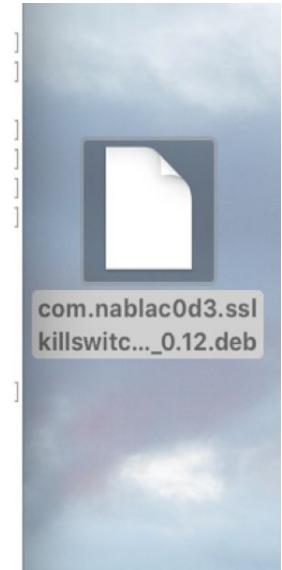
PreferenceTag3 (iOS 9)

PreferMyFi ↗



iOS (iPhone OS) SSL Kill Switch 2 ile SSL Pinning Atlatma

```
Ahmets-MacBook-Pro:Desktop ahmet$ scp com.nabla0d3.sslkillswitch2_0.12.deb root@192.168.1.31:/tmp  
[root@192.168.1.31's password:  
com.nabla0d3.sslkillswitch2_0.12.deb  
Ahmets-MacBook-Pro:Desktop ahmet$ ssh root@192.168.1.31  
[root@192.168.1.31's password:  
iPhone-5:~ root# cd /tmp  
iPhone-5:/tmp root# dpkg -i com.nabla0d3.sslkillswitch2_0.12.deb  
Selecting previously unselected package com.nabla0d3.sslkillswitch2.  
(Reading database ... 1216 files and directories currently installed.)  
Preparing to unpack com.nabla0d3.sslkillswitch2_0.12.deb ...  
Unpacking com.nabla0d3.sslkillswitch2 (0.12-3+debug) ...  
Setting up com.nabla0d3.sslkillswitch2 (0.12-3+debug) ...  
iPhone-5:/tmp root# killall -HUP SpringBoard  
iPhone-5:/tmp root#
```



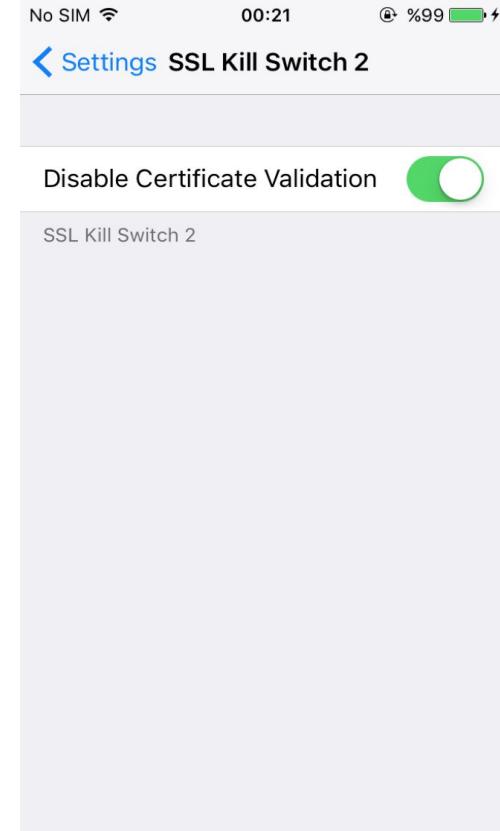
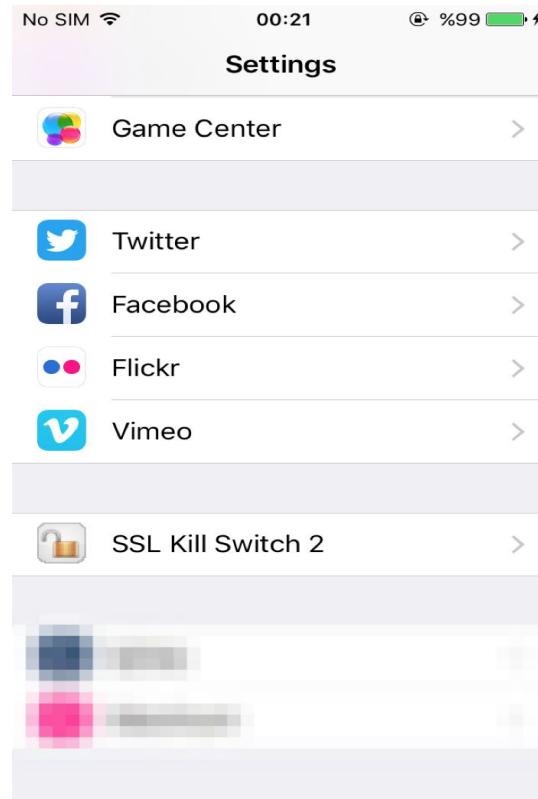
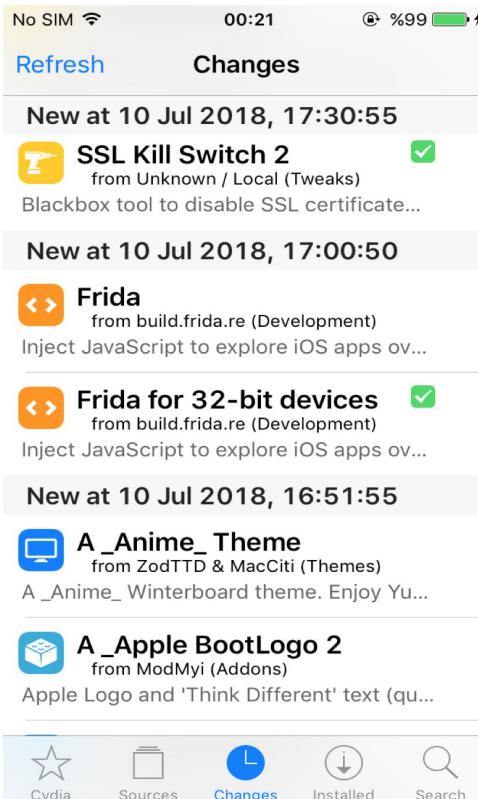
iOS (iPhone OS) SSL Kill Switch 2 ile SSL Pinning Atlatma

SSH bağlantısı ile cihazımıza bağlanıyoruz ve cd komutu ile /tmp dizinine geçiş yapıyoruz.

Github readme sayfasında yazan kurulum komutları olan linux sistemleri .deb uzantılı paketleri kurmamıza yarayan dpkg –i komutunu kullanarak deb paketimizi kuruyoruz.

Son olarak **killall -HUP SpringBoard** komutu ile kurulumu tamamlıyoruz.

iOS (iPhone OS) SSL Kill Switch 2 ile SSL Pinning Atlatma



iOS (iPhone OS) Frida Kurulumu ve Bağlantısı

Kurulumuna ve kullanımına geçmeden önce kısaca Frida'dan bahsetmek gerekirse Frida dinamik olarak çalışan uygulamanın çalışma anında müdahale edip bizim değişikler yapmamızı sağlayan bir araç takımıdır.

Python kurulu bilgisayarlarda pip ile kolaylıkla bilgisayara kurulmaktadır. Bu tek başına yeterli değildir.

Frida'nın agent uygulamasında test cihazına/emulatöre kurmamız gerekmektedir.
sudo pip install Frida komutu ile kurulumu gösterilmiştir.

Çıkan çıktıda zaten daha önceden kurulduğu uyarısı verilmiştir. İlk defa kuranlarda böyle bir mesaj çıkmayacaktır.

iOS (iPhone OS) Frida Kurulumu ve Bağlantısı

```
[Ahmet's-MacBook-Pro:~ ahmet$ sudo pip install frida
[Password:
The directory '/Users/ahmet/Library/Caches/pip/http' or its parent directory is not owned by the current user and the cache has been disabled. Please check the permissions and owner of that directory. If executing pip with sudo, you may want sudo's -H flag.
The directory '/Users/ahmet/Library/Caches/pip' or its parent directory is not owned by the current user and caching wheels has been disabled. Check the permissions and owner of that directory. If executing pip with sudo, you may want sudo's -H flag.
Requirement already satisfied: frida in /usr/local/lib/python2.7/site-packages (11.0.13)
Requirement already satisfied: colorama<1.0.0,>=0.2.7 in /usr/local/lib/python2.7/site-packages (from frida) (0.3.9)
Requirement already satisfied: prompt-toolkit<2.0.0,>=0.57 in /usr/local/lib/python2.7/site-packages (from frida) (1.0.15)
Requirement already satisfied: pygments<3.0.0,>=2.0.2 in /usr/local/lib/python2.7/site-packages (from frida) (2.2.0)
Requirement already satisfied: wcwidth in /usr/local/lib/python2.7/site-packages (from prompt-toolkit<2.0.0,>=0.57->frida) (0.1.7)
Requirement already satisfied: six>=1.9.0 in /usr/local/lib/python2.7/site-packages (from prompt-toolkit<2.0.0,>=0.57->frida) (1.11.0)
[Ahmet's-MacBook-Pro:~ ahmet$ frida-ls-devices
[Id          Type      Name
-----      -----
local       local    Local System
c848e7dc012b8b807fffa2b2b77201fd968a2263 tether   iPhone
tcp         remote   Local TCP
Ahmet's-MacBook-Pro:~ ahmet$ ]]
```

iOS (iPhone OS) Frida Kurulumu ve Bağlantısı

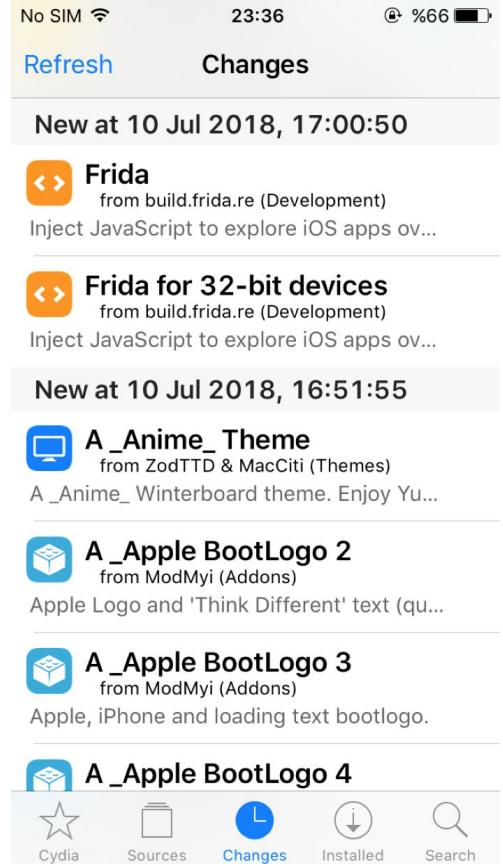
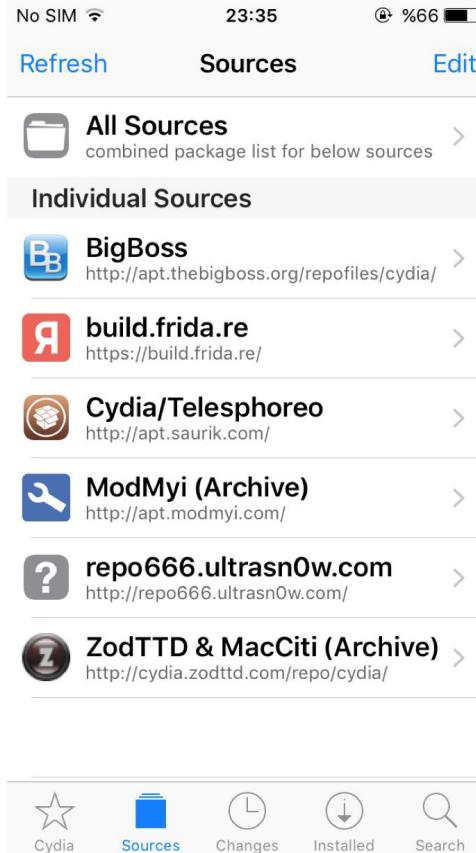
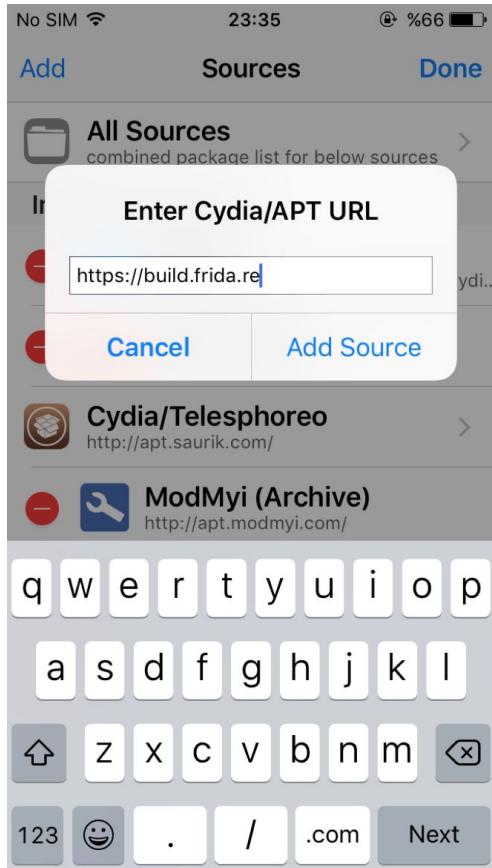
Frida kurulduktan sonra **frida-ls-device** komutu ile USB ile bağlı olan cihazlar listelenenecektir.

Gelelim şimdi Jailbreak yapılmış iOS cihaza Frida agent kurulumuna Cydia Uygulama marketine Frida'nın source adresi eklenmektedir.

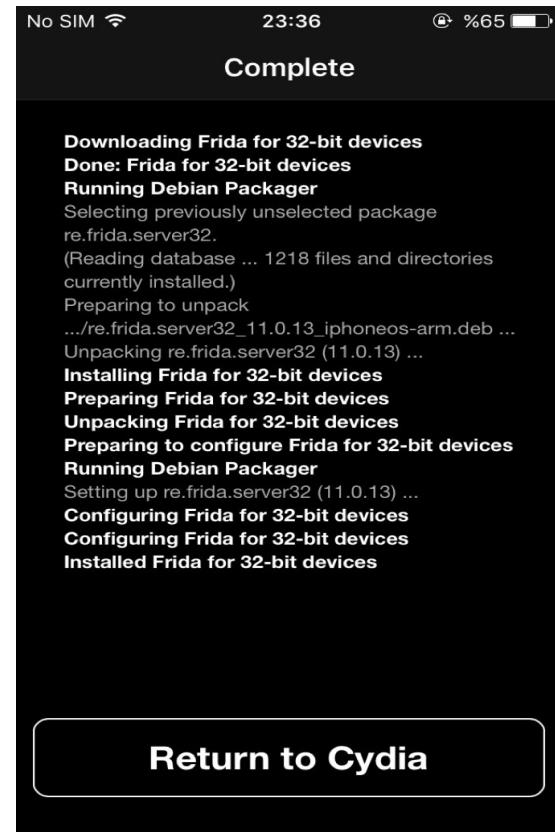
Doğrudan Cydia da aratıp kurulamamaktadır. Cydia açılarak aşağıdaki menüden Sources'a tıklanmalıdır.

Daha sonra ise sağ üst kısımdan Edit'e basılmalı ve sonrasında sol üst kısımda bulunan Add butonuna basılarak <https://build.frida.re> adresi Soruce olarak girilmelidir.

iOS (iPhone OS) Frida Kurulumu ve Bağlantısı



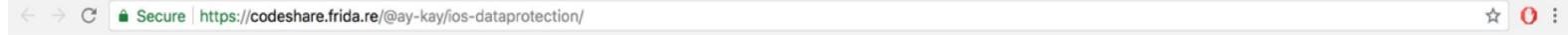
iOS (iPhone OS) Frida Kurulumu ve Bağlantısı



iOS (iPhone OS) Frida Kurulumu ve Bağlantısı

```
|Ahmet's-MacBook-Pro:~ ahmet$ frida-ps -U
PID  Name
-----
1199  Cydia
1194  Photos
1406  Settings
1404  h3lix
      55 AppleIDAuthAgent
    709 AssetCacheLocatorService
      70 BTServer
     101 BlueTool
1087  CMFSyncAgent
   161 CacheDeleteAppContainerCaches
   160 CacheDeleteiTunesStore
   140 CallHistorySyncHelper
   169 CloudKeychainProxy
1098  CloudPhotoDerivativeGenerator
     23 CommCenter
   186 ContainerMetadataExtractor
   113 DuetHeuristic-BM
1062  EscrowSecurityAlert
   170 IMDPersistenceAgent
   166 KeychainSyncingOverIDSProxy
   157 MobileBackupCacheDeleteService
     78 MobileGestaltHelper
   102 MobileStorageMounter
     95 OTATaskingAgent
   143 SafariCloudHistoryPushAgent
   184 ServerFileProvider
1175  SpringBoard
     22 UserEventAgent
   100 WirelessRadioManagerd
   185 absd
     93 accountsd
   178 adid
   805 afcd
   756 afcd
   164 aggregated
   138 akd
   175 amfid
   126 appstored
     81 apsd
     66 askpermissiond
     88 aslmanager
     64 assertiond
```

iOS (iPhone OS) Frida Kurulumu ve Bağlantısı



```
1- /*  
2  * iOS Data Protection  
3  *  
4  * getDataProtectionKeysForAllPaths() - List iOS file data protection classes (NSFileProtectionKey) of an app  
5  *  
6  */  
7  
8- function listDirectoryContentsAtPath(path) {  
9  var fileManager = ObjC.classes.NSFileManager.defaultManager();  
10 var enumerator = fileManager.enumeratorAtPath_(path);  
11 var file;  
12 var paths = [];  
13  
14 while ((file = enumerator.nextObject()) != null){  
15  paths.push(path + '/' + file);  
16 }  
17  
18 return paths;  
19 }  
20  
21 function listHomeDirectoryContents() {  
22  var homePath = ObjC.classes.NSProcessInfo.processInfo().environment().objectForKey_("HOME").toString();  
23  var paths = listDirectoryContentsAtPath(homePath);  
24  return paths;  
25 }  
26  
27 function getDataProtectionKeyForPath(path) {  
28  var fileManager = ObjC.classes.NSFileManager.defaultManager();  
29  var urlPath = ObjC.classesNSURL.fileURLWithPath_(path);  
30  attributeDict = dictFromNSDictionary(fileManager.attributesOfItemAtPath_error_(urlPath.path(), NULL));  
31  return attributeDict.NSFileProtectionKey;  
32 }  
33  
34 function getDataProtectionKeysForAllPaths() {  
35  var fileManager = ObjC.classes.NSFileManager.defaultManager();  
36  var dict = {};  
37  var paths = listHomeDirectoryContents();  
38 }
```

iOS (iPhone OS) Frida Kurulumu ve Bağlantısı

The screenshot shows a web browser window titled "Frida CodeShare" displaying a project page. The URL is <https://codeshare.frida.re/@lichao890427/ios-ssl-bypass/>. The page has a header with "Person 1" and "Log In". Below the header, there's a "Frida CodeShare" link and some social sharing icons. The main content area is titled "Project: iOS SSL Bypass". It includes a note "Try this code out now by running" followed by a command-line snippet: \$ frida --codeshare lichao890427/ios-ssl-bypass -f YOUR_BINARY. The main body of the page contains a large block of Frida script code.

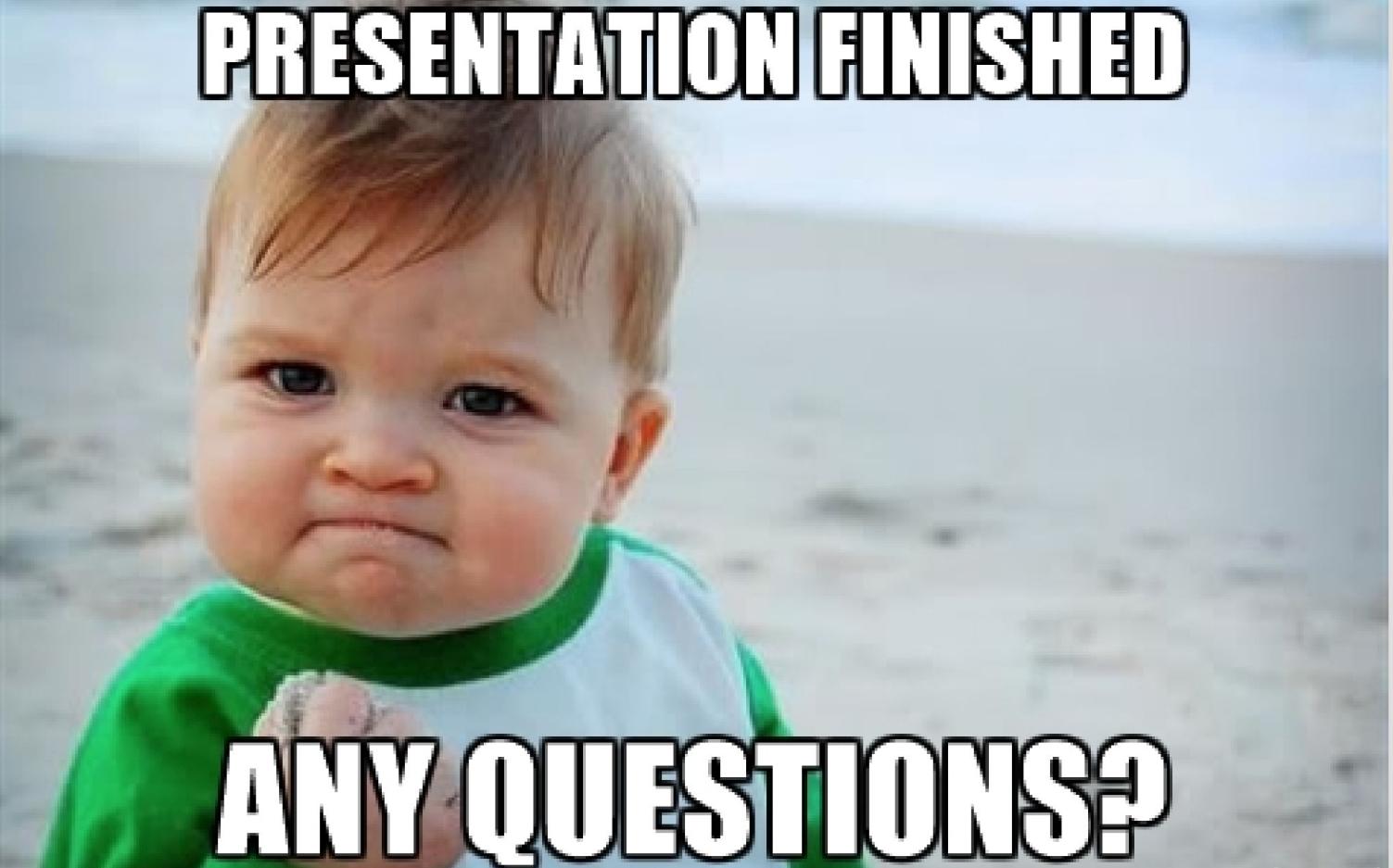
```
1 // https://github.com/lichao890427/personal_script/blob/master/Frida_script/utils.js
2 // Submit bugs on git
3
4 function forceTrustCert() {
5     Interceptor.replace(Module.findExportByName(null, 'SecTrustEvaluate'),
6         new NativeCallback(function(trust, result) {
7             Memory.writePointer(result, ptr('0x1'));
8             console.log('pass SecTrustEvaluate');
9             return 0;
10        }, 'int', ['pointer', 'pointer'])
11    );
12    if (typeof(ObjC.classes.AFSecurityPolicy) != 'undefined') {
13        Interceptor.attach(ObjC.classes.AFSecurityPolicy['- evaluateServerTrust:forDomain:'].implementation, {
14            onEnter: function(args) {
15                console.log('pass -[AFSecurityPolicy evaluateServerTrust:forDomain:]')
16            },
17            onLeave: function(retval) {
18                retval.replace(ptr('0x1'));
19            }
20        });
21        Interceptor.attach(ObjC.classes.AFSecurityPolicy['- setAllowInvalidCertificates:'].implementation, {
22            onEnter: function(args) {
23                args[2] = ptr('0x1');
24                console.log('pass -[AFSecurityPolicy setAllowInvalidCertificates:]')
25            },
26            onLeave: function(retval) {}
27        });
28        Interceptor.attach(ObjC.classes.AFSecurityPolicy['- allowInvalidCertificates'].implementation, {
29            onEnter: function(args) {
30                console.log('pass -[AFSecurityPolicy setAllowInvalidCertificates:]')
31            },
32            onLeave: function(retval) {
```

iOS (iPhone OS) Frida Kurulumu ve Bağlantısı

```
frida --codeshare ay-kay/ios-dataprotection -f YOUR_BINARY
```

```
frida --codeshare lichao890427/ios-ssl-bypass -f YOUR_BINARY
```

PRESENTATION FINISHED



ANY QUESTIONS?