

Android SSL Pinning Bypass

Hazırlayan

Ümmü Derya Çelik

SSL Pinning Bypass Nedir?

Certificate Pinning olarak da bilinen SSL pinning bir güvenlik önlemi sistemidir. Uzak sunucu ile uygulama arasında, olası güvenlik tehditlerine karşı sunucu sertifikalarını tekrar tekrar doğrulayarak güvenli ve şifreli iletişim kanalları sağlar. Bu sistem önemli bilgilerimizin, kullanıcı adı, şifre, banka bilgileri gibi, sunucu ile haberleşme sırasında aradaki adam(MITM) tarafından ele geçirilmemesi için bir önlemdir.

SSL Nasıl Çalışır?

- İstemci sunucuya bağlanır ve sunucu kendisini tanımlamasını ister.
- Sunucu istemciye sertifika gönderir (ortak anahtar içerir Public Key)
- Client bu sertifikanın geçerli olup olmadığını kontrol eder. Varsa, istemci bir simetrik anahtar (oturum anahtarı - Session Key) oluşturur, ortak anahtarla şifreyi sunucuya geri gönderir.
- Sunucu şifreli simetrik anahtarı alır, kendi özel şifresi ile şifresini çözer, sonra müşteriye onay paketi gönderir.
- Müşteri ACK'yi alır ve oturumu başlatır.

SSL kullanarak, istemci yalnızca geçerli sertifikaya sahip güvenilir kaynaklardan bağlantıya izin verecektir. Sorun istemci ile sunucu arasında sunucu gibi davranan bir saldırganın varlığı olduğunda başlamaktadır.

SSL pinning bypass, uygulama ve sunucu arasındaki güvenliği ve gizliliği sağlayan bu sertifikaların suiistimal edilip uygulamanın saldırganın kendi belirlediği sertifikaya güvenmesiyle oluşan güvenlik zafiyetidir.

Bu tür saldırılara Ortadaki Adam Man-In-The-Middle saldırıları denmektedir.

SSL Pinning Bypass saldırısı yapabilmek için dört farklı seçenek var. Bunlar;

- 1) Custom sertifika Ekleme
- 2) Gömülü sertifikanın üzerine yazmak
- 3) Frida ile müdahale
- 4) Custom sertifika doğrulama kodunun reverse edilmesi

Bu yazıda Frida ile SSL Pinning Bypass uygulama adımlarına değinilecektir.

Frida ile SSL Pinning Bypass

1. Adım: Yapacağımız işlemler için gerekli araçları ve programları bilgisayarımıza yüklemek ile başlayalım.

Gerekli araçlar ve programlar; Frida, frida tools, Python3, Android Plotform tools ve Trafiği incelemek için Burpsuite programı.

```
pip install Frida
pip install objection
pip install frida-tools

sudo apt update && sudo apt install android-sdk
sudo apt-get install android-tools-adb
```

Komutlarıyla Linux işletim sistemimize bu paketleri yüklüyoruz.

```
root@kali:~/home/derya# pip install frida
WARNING: Value for scheme.platlib does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/lib/python3.9/dist-packages
sysconfig: /usr/lib/python3.9/site-packages
WARNING: Value for scheme.purelib does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/lib/python3.9/dist-packages
sysconfig: /usr/lib/python3.9/site-packages
WARNING: Value for scheme.headers does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/include/python3.9/UNKNOWN
sysconfig: /usr/include/python3.9/UNKNOWN
WARNING: Value for scheme.scripts does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/bin
sysconfig: /usr/bin
WARNING: Value for scheme.data does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local
sysconfig: /usr
WARNING: Additional context:
user = False
home = None
root = None
prefix = None
Collecting frida
  Downloading frida-15.0.14.tar.gz (9.1 kB)
Requirement already satisfied: click in /usr/lib/python3/dist-packages (from frida) (5.0.0)
Building wheels for collected packages: frida
  Building wheel for frida (setup.py) ... done
  Created wheel for frida: filename=frida-15.0.14-cp39-cp39-linux_x86_64.whl size=22164099 sha256=7a18bd3abccffafae45
  Stored in directory: /root/.cache/pip/wheels/84/95/c7/d642f3ff295616c4285d41551885bc3dcf5ad241982ce62cb8
Successfully built frida
Installing collected packages: frida
WARNING: Value for scheme.headers does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/include/python3.9/frida
sysconfig: /usr/include/python3.9/frida
Successfully installed frida-15.0.14
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system pa
ckage manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
WARNING: You are using pip version 21.2.1; however, version 21.2.4 is available.
You should consider upgrading via the '/usr/bin/python3.9 -m pip install --upgrade pip' command.
```

```
root@kali:~/home/derya# pip install objection
WARNING: Value for scheme.platlib does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/lib/python3.9/dist-packages
sysconfig: /usr/lib/python3.9/site-packages
WARNING: Value for scheme.purelib does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/lib/python3.9/dist-packages
sysconfig: /usr/lib/python3.9/site-packages
WARNING: Value for scheme.headers does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/include/python3.9/UNKNOWN
sysconfig: /usr/include/python3.9/UNKNOWN
WARNING: Value for scheme.scripts does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/bin
sysconfig: /usr/bin
WARNING: Value for scheme.data does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local
sysconfig: /usr
WARNING: Additional context:
user = False
home = None
root = None
prefix = None
Collecting objection
  Downloading objection-1.11.0.tar.gz (327 kB)
  327 kB 551 kB/s
Requirement already satisfied: click in /usr/lib/python3/dist-packages (from objection) (7.1.2)
Collecting delegator.py
  Downloading delegator.py-0.1.1-py2.py3-none-any.whl (5.0 kB)
Requirement already satisfied: flask in /usr/lib/python3/dist-packages (from objection) (1.1.2)
Collecting frida-tools<=0.0.0
  Downloading frida-tools-10.2.1.tar.gz (40 kB)
  40 kB 695 kB/s
Requirement already satisfied: frida<=14.0.0 in /usr/local/lib/python3.9/dist-packages (from objection) (15.0.14)
Collecting litecli<=1.2.0
  Downloading litecli-1.6.0-py2.py3-none-any.whl (47 kB)
  47 kB 725 kB/s
Requirement already satisfied: prompt-toolkit<4.0.0, >=3.0.3 in /usr/lib/python3/dist-packages (from objection) (3.0.1)
```

```
root@kali:~/home/derya# pip install frida-tools
WARNING: Value for scheme.platlib does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/lib/python3.9/dist-packages
sysconfig: /usr/lib/python3.9/site-packages
WARNING: Value for scheme.purelib does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/lib/python3.9/dist-packages
sysconfig: /usr/lib/python3.9/site-packages
WARNING: Value for scheme.headers does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/include/python3.9/UNKNOWN
sysconfig: /usr/include/python3.9/UNKNOWN
WARNING: Value for scheme.scripts does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local/bin
sysconfig: /usr/bin
WARNING: Value for scheme.data does not match. Please report this to <https://github.com/pypa/pip/issues/10151>
distutils: /usr/local
sysconfig: /usr
WARNING: Additional context:
user = False
home = None
root = None
prefix = None
Requirement already satisfied: frida-tools in /usr/local/lib/python3.9/dist-packages (10.2.1)
Requirement already satisfied: prompt-toolkit<4.0.0, >=3.0.0 in /usr/lib/python3/dist-packages (from frida-tools) (3.0.1)
Requirement already satisfied: pygments<3.0.0, >=2.0.2 in /usr/lib/python3/dist-packages (from frida-tools) (2.7.1)
Requirement already satisfied: colorama<1.0.0, >=0.2.7 in /usr/lib/python3/dist-packages (from frida-tools) (0.4.4)
Requirement already satisfied: frida<16.0.0, >=15.0.0 in /usr/local/lib/python3.9/dist-packages (from frida-tools) (15.0.14)
Requirement already satisfied: setuptools in /usr/lib/python3/dist-packages (from frida<16.0.0, >=15.0.0->frida-tools) (52.0.0)
WARNING: Running pip as the 'root' user can result in broken permissions and conflicting behaviour with the system pa
ckage manager. It is recommended to use a virtual environment instead: https://pip.pypa.io/warnings/venv
WARNING: You are using pip version 21.2.1; however, version 21.2.4 is available.
You should consider upgrading via the '/usr/bin/python3.9 -m pip install --upgrade pip' command.
```

Komutların çıktısı görsellerdeki gibidir.

2. Adım: Adb ile emülatörün bağlantısını kuralım.

```
(root@derya)-[/home/derya]
# adb connect 192.168.229.120:5555
* daemon not running; starting now at tcp:5037
* daemon started successfully
connected to 192.168.229.120:5555

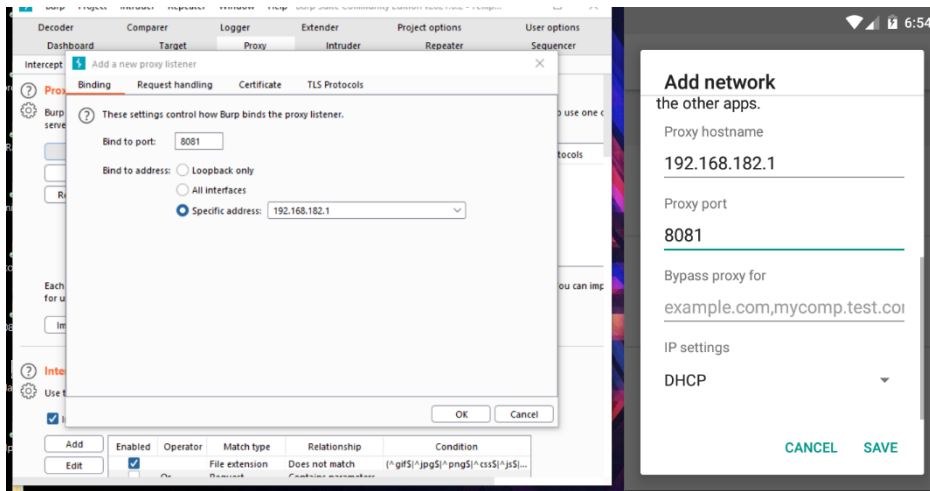
(root@derya)-[/home/derya]
# adb devices
List of devices attached
192.168.229.120:5555    device
```

Cihazımızın ip adresini girip connect komutuyla bağlantı kurduk.

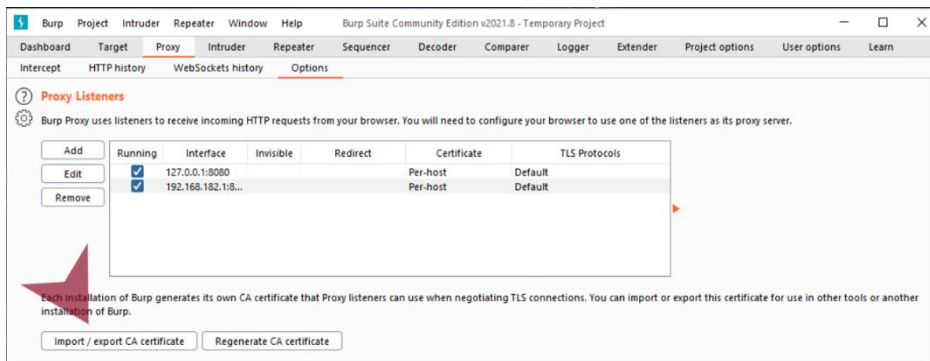
Adb devices komutuyla adb ile bağlantı kuran cihazları görebiliriz.

Adb ile başarıyla bağlantı kuruldu.

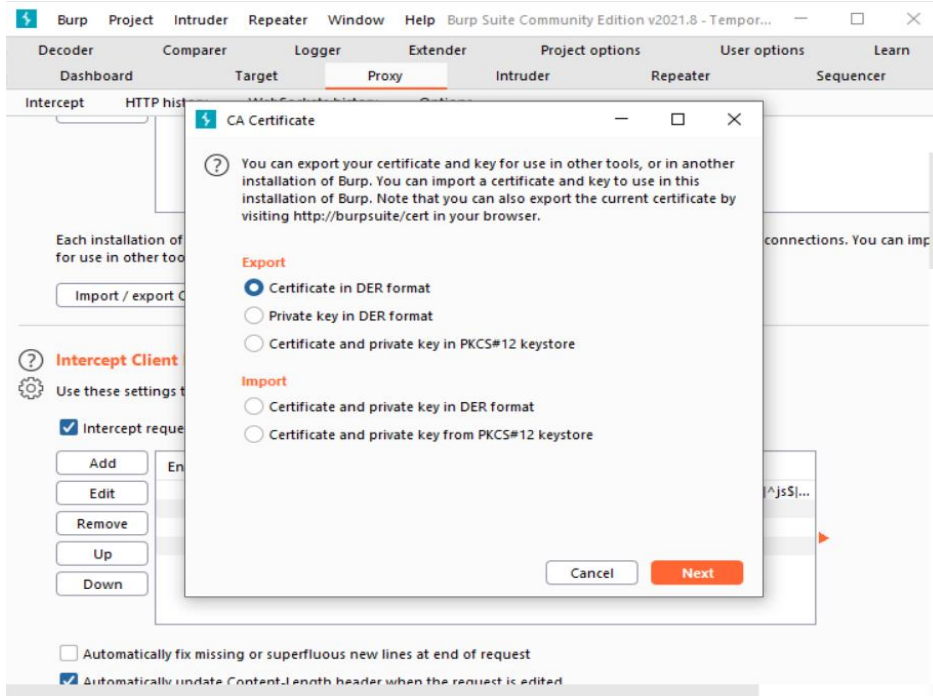
3. Adım: Burpsuite ile android cihazın bağlantısını kuralım.



Proxy ayarlarından dinleyeceğimiz IP ve Port adreslerini girip kaydedelim



Şekilde ok ile de gösterilen Import / export CA certificate butonuna tıklayalım.

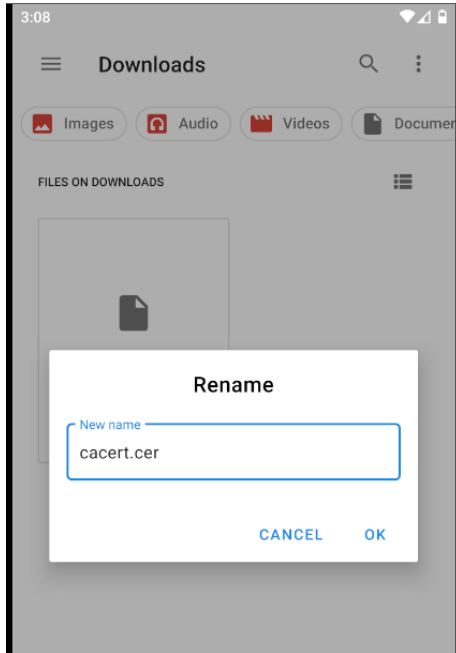


Devamında bu pencere çıkacak. Buradan da DER format seçip Burpsuite sertifikasını kaydedelim.

Kaydettiğimiz bu sertifikayı android cihazımıza yükleyip sertifikayı kaydedelim.

```
(root@derya)-[/home/derya]
# adb push /home/derya/Desktop/cacert.der /sdcard/Download
* daemon not running; starting now at tcp:5037
* daemon started successfully
```

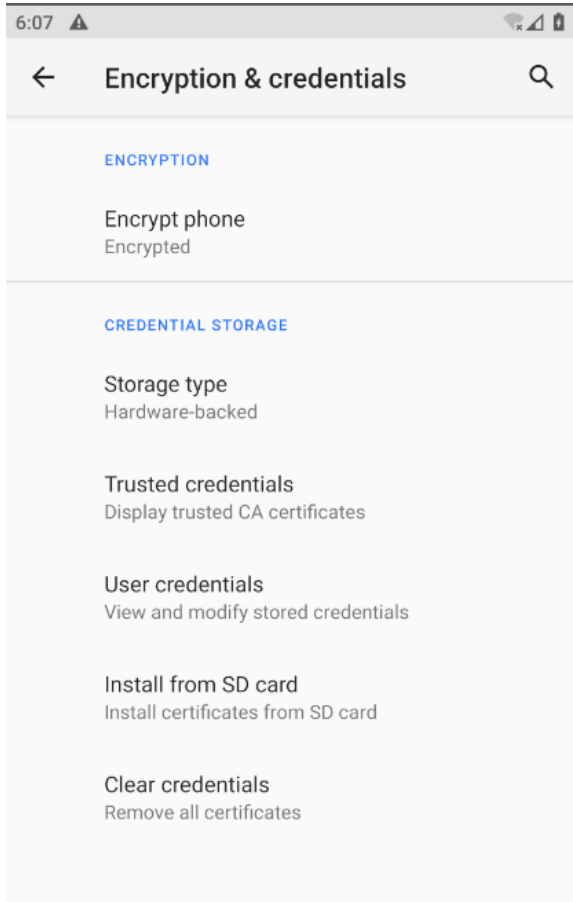
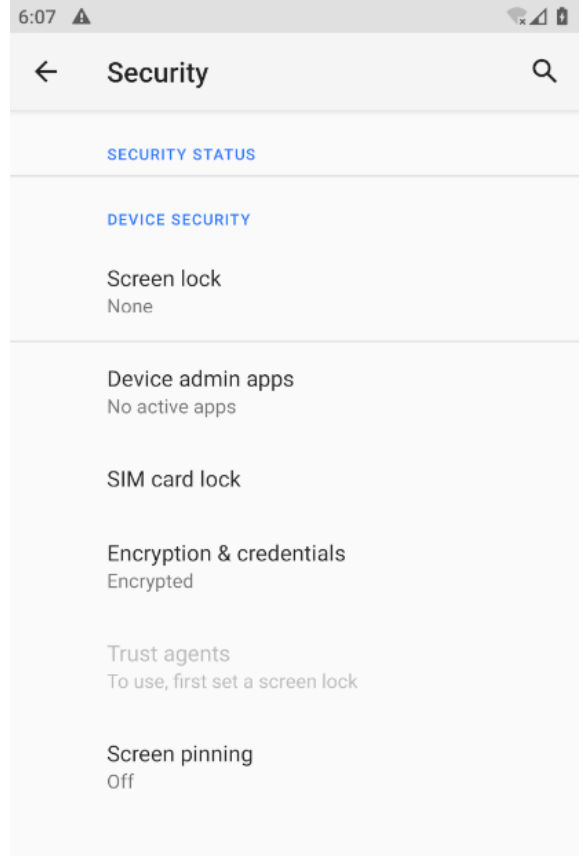
Bu komutla sertifika dosyasını cihazın indirilenler dosyasına attık.



Cihazda indirilenler klasörüne gidip sertifika dosyasını okuyabilmesi için .der olan uzantısını .cer yapıp kaydedelim

Sertifikayı kaydetme aşamasına geçtik.

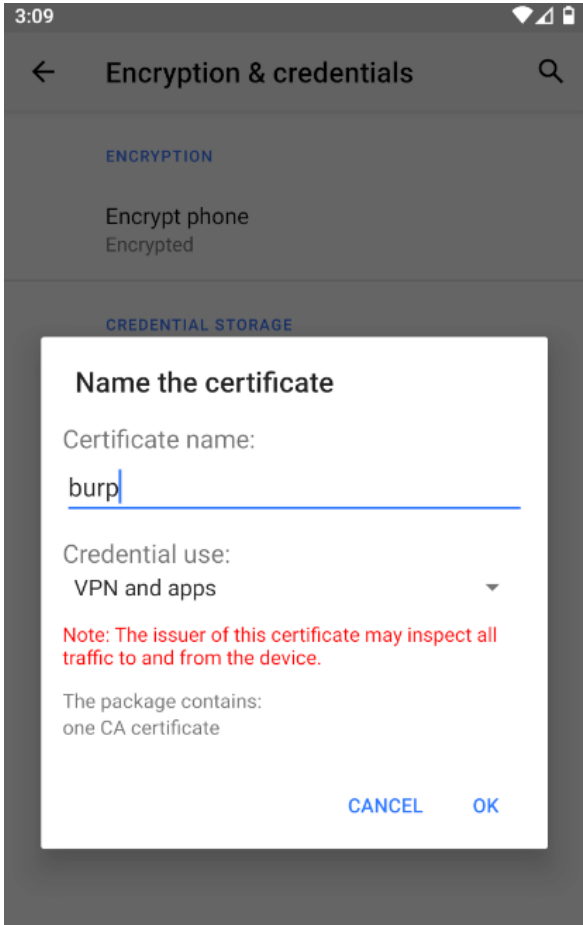
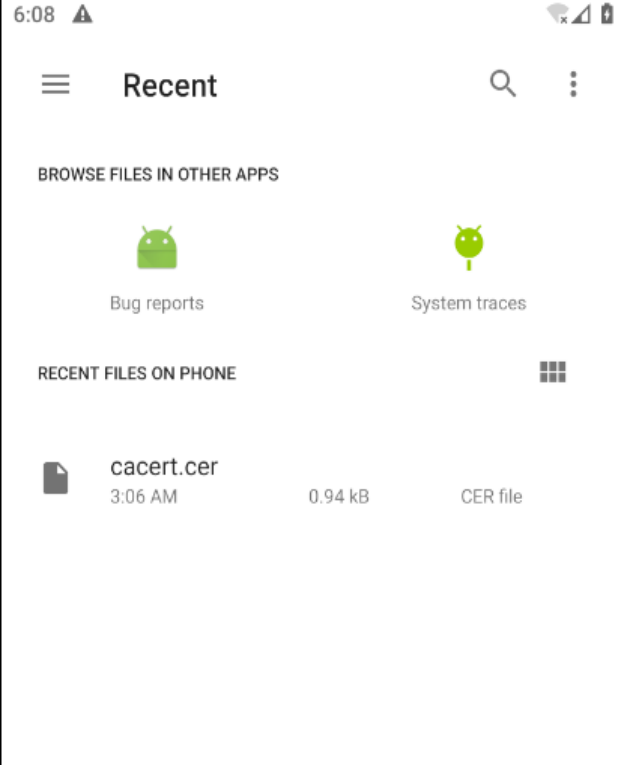
Ayarlardan güvenlik penceresini açalım.
Encryption & credentials seçeneğine
girelim.



install from sdcard' a tıklayalım.

Yüklü olan sertifikalar hali hazırda görünmekte.

Cacert.cer dosyasını seçiyoruz.



Sertifikaya bir ad verip kaydediyoruz. Böylece sertifika kaydetme işlemi tamamlandı.

Burpsuite' le de tam bağlantıyı sağlamış olduk.

4. Adım: Android cihaza frida-server kurulumunu yapalım.

Bu linkten sistemimize uygun olan [frida-server](#) dosyasını indirelim. İndirdikten sonra dosya adını frida-server olarak kısaltmak işlem kolaylığı sağlayacaktır.

Sisteminizin hangi frida-server türüne uygun olduğunu öğrenmek için terminale bu kodu girebilirsiniz;

```
adb shell getprop ro.product.cpu.abi
```

adb push komutuyla indirdiğimiz paketi android cihaza gönderelim.

```
(root@derya)-[/home/derya]
# adb push /home/derya/Desktop/frida-server /data/local/tmp
/home/derya/Desktop/frida-server/: 1 file pushed. 16.1 MB/s (46347700 bytes in 2.738s)
```

Uygulamanın sorunsuz çalışabilmesi için izinlerini verelim.

```
(root@derya)-[/home/derya]
# adb shell chmod 777 /data/local/tmp/frida-server
```

Frida Server kurulumu da tamamlandı. Çalışıp çalışmadığını kontrol edelim.

```
(root@derya)-[/home/derya]
# adb shell /data/local/tmp/frida-server &
[1] 3029
```

Bu şekilde bir çıktı alıyorsak uygulama sorunsuz çalışıyor demektir.

5. Adım: Kullanılacak olan SSL Pinning scriptini cihazımıza yükleyelim.

Bu scripte, <https://codeshare.frida.re/@pcipolloni/universal-android-ssl-pinning-bypass-with-frida/> linkinden ulaşabilirsiniz.

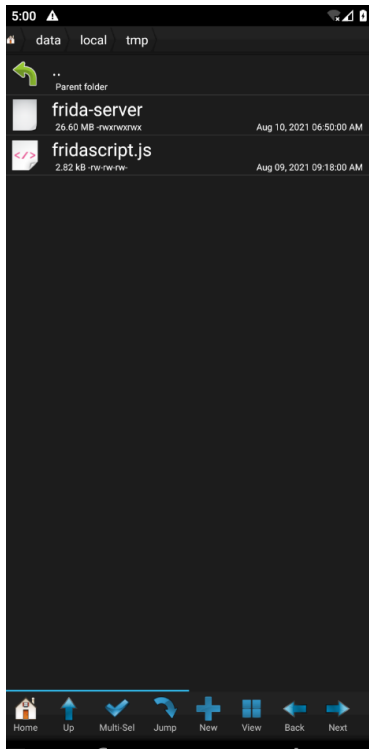
Boş bir text dosyası oluşturup bu linkteki kodları kopyalayıp “fridascript.js” adıyla kaydedelim.

Şimdi bu dosyayı adb ile cihazımıza gönderelim.

```
(root@derya)-[/home/derya]
# adb push /home/derya/Desktop/fridascript.js /data/local/tmp
/home/derya/Desktop/fridascript.js: 1 file pushed. 0.0 MB/s (2771 bytes in 0.068s)
```

```
(root@derya)-[/home/derya]
# adb shell chmod 777 /data/local/tmp/fridascript.js
```

İzinlerini de verdikten sonra frida çalışmaya hazır hale geldi.



Aktardığımız dosyaları görmekteyiz.

Şimdi Frida'yı çalıştıralım.

6. Adım: Atak için hazır olduğumuza göre uygulamaya başlayalım.

```
frida-ps -U
```

```
(root@derya)-[/home/derya]
# frida-ps -U
PID  Name
-----
4567  Phone
4283  Root Browser
4196  Settings
4432  Superuser
4826  Twitter
382   adbd
1042  android.ext.services
287   android.hardware.audio@2.0-service
288   android.hardware.camera.provider@2.4-service
289   android.hardware.cas@1.1-service
291   android.hardware.configstore@1.1-service
292   android.hardware.drm@1.0-service
293   android.hardware.drm@1.2-service.clearkey
294   android.hardware.gnss@1.0-service
295   android.hardware.graphics.allocation@2.0-service
296   android.hardware.graphics.composer@2.1-service
297   android.hardware.health@2.0-service.genymotion
231   android.hardware.keymaster@3.0-service
298   android.hardware.light@2.0-service
299   android.hardware.memtrack@1.0-service
300   android.hardware.power@1.0-service
301   android.hardware.sensors@1.0-service
302   android.hardware.wifi@1.0-service
285   android.hidl.allocation@1.0-service
3636  android.process.media
286   android.system.suspend@1.0-service
253   apexd
304   ashmemd
306   audioserver
177   batteryd
337   camerad
3573  com.android.certinstaller
1081  com.android.inputmethod.latin
1105  com.android.launcher3
900   com.android.networkstack
4098  com.android.packageinstaller
1986  com.android.permissioncontroller
936   com.android.phone
1453  com.android.se
1558  com.android.smspush
4077  com.android.statementservice
803   com.android.systemui
4329  com.android.webview:sandboxed_process0:org.chromium.content.app.SandboxedPr
4580  com.android.webview:webview_service
1535  com.genymotion.genyid
```

Komutuyla cihazda bulunan uygulamalar şekildeki gibi listelenecektir.

SSL Pinning yapmak istediğimiz uygulamanın paket ismini buluyoruz. Ben örnek olarak Twitter uygulamasını seçtim.

```
frida -U -f com.twitter.android -l home/username/fridascript.js --no-paus
```

```
Frida 15.0.13 - A world-class dynamic instrumentation toolkit
Commands:
  help      -> Displays the help system
  object?   -> Display information about 'object'
  exit/quit -> Exit
  More info at https://frida.re/docs/home/
Failed to spawn: need Gadget to attach on jailed Android; its default location is: C:\Users\msi\AppData\Local\Microsoft\Windows\INetCache\frida\gadget-android-arm64.so
```

Frida Gadget paketine ihtiyacım olduğunu söyleyen bir hatayla karşılaştım. Frida gadget paketi Frida'nın enjekte edildiği ortamda çalışmasına uygun bir ortam olmadığında yama olarak kullanılabilecek yardımcı bir kütüphanedir.

Frida-server paketini android cihazımıza yüklediğimiz gibi adb push komutuyla yükleyip izinlerini tamamlıyoruz ayrıca "Location is: " devamında belirttiği konumda bilgisayarımızda da Gadget paketine sahip olmamız gerektiğini söylüyor bu yüklemeyi de tamamladık. Şimdi tekrar deneyelim;

```
Frida 15.0.13 - A world-class dynamic instrumentation toolkit
Commands:
  help      -> Displays the help system
  object?   -> Display information about 'object'
  exit/quit -> Exit
  More info at https://frida.re/docs/home/
Failed to spawn: java.lang.UnsatisfiedLinkError: dlopen failed: library "/data/data/com.twitter.android/gadget.so" needed or dlopened by "/apex/com.android.runtime/lib/libnative.so" is not accessible for the namespace "runtime"
C:\Users\msi\OneDrive\Masaüstü\Yeniklasor\adb\platform-tools>
```

Bu seferde "runtime ad alanı için yürütülebilir değil" tarzında bir hata aldım. Hatayı araştırdığımda gadget kütüphanesinin yüklenmesinin engellendiğini ve Android API 24 ve üzeri sürümler için özel kütüphane tanımlamanın engellendiğini öğrendim.

Daha düşük android sürümlerinde denenmiş olsaydı süreç nasıl devam edecekti bakalım.

Aynı komutu tekrar çalıştırıyoruz.

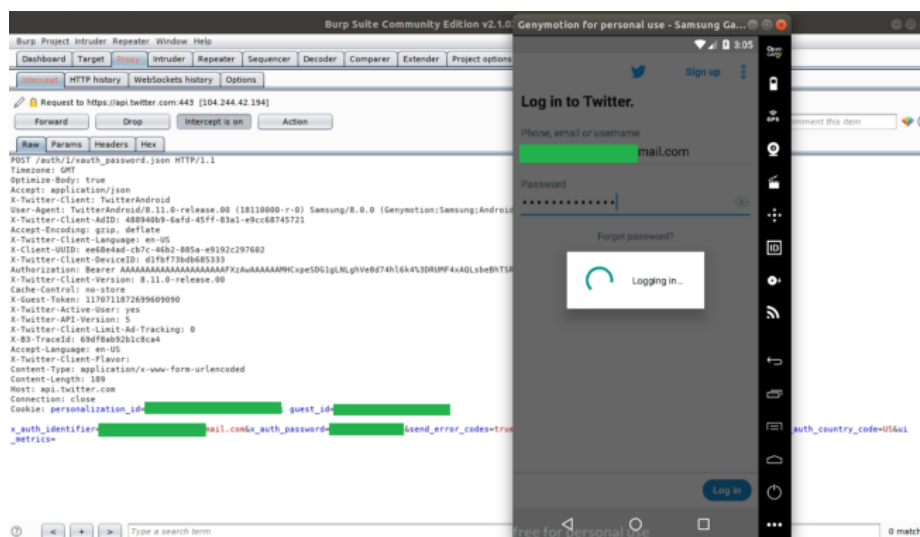
```
/~$ frida -U -i com.twitter.android --script sslpinning.js
Frida 12.6.12 - A world-class dynamic instrumentation toolkit

Commands:
  help          -> Displays the help system
  object?       -> Display information about 'object'
  exit/quit     -> Exit

More info at http://www.frida.re/docs/home/

Spawned com.twitter.android. Resuming main thread!
[Genymotion Google Pixel XL::com.twitter.android]->
[.] Cert Pinning Bypass/Re-Pinning
[+] Loading our CA...
[o] Our CA Info: CN=PortSwigger CA, OU=PortSwigger CA, O=PortSwigger, L=PortSwigger, ST=PortSwigger, C=PortSwigger
[+] Creating a KeyStore for our CA...
[+] Creating a TrustManager that trusts the CA in our KeyStore...
[+] Our TrustManager is ready...
[+] Hijacking SSLContext methods now...
[-] Waiting for the app to invoke SSLContext.init()...
[o] App invoked javax.net.ssl.SSLContext.init()...
[+] SSLContext initialized with our custom TrustManager!
```

Uygulamaya erişim sağlandı. Burpsuite uygulamasına geçip ağ trafiğini oradan izleyebiliriz.



Twitter'a giriş yapıldığında burpsuite uygulamasına girilen şifreler ve mail adresleri görüldüğü gibi yakalanmakta. SSL Pinning Bypass saldırısı başarıyla gerçekleştirildi.

SSL Pinning Nasıl Önlenir?

Sertifika Sabitleme metoduyla tamamen güvenliği sağlayamamak da saldırıya uğrama riski oranı düşürülebilir.

Sertifika sabitlemede, geliştirici SSL sertifikasının bazı bayt kodlarını uygulama koduna kodlar. Uygulama sunucuya iletişim kurduğunda, bir sertifikada aynı bayt kodunun bulunup bulunmadığını kontrol eder. Varsa, uygulama sunucuya bir istek gönderir. Bayt kodu eşleşmezse bir SSL sertifikası hatası atar. Bu teknik, bir saldırganın kendi imzasını taşıyan sertifikasını kullanmasını engeller.