ANDROID APK'S HACKEN

Decompilen, manipulieren, backdooren

WORUM GEHT ES?

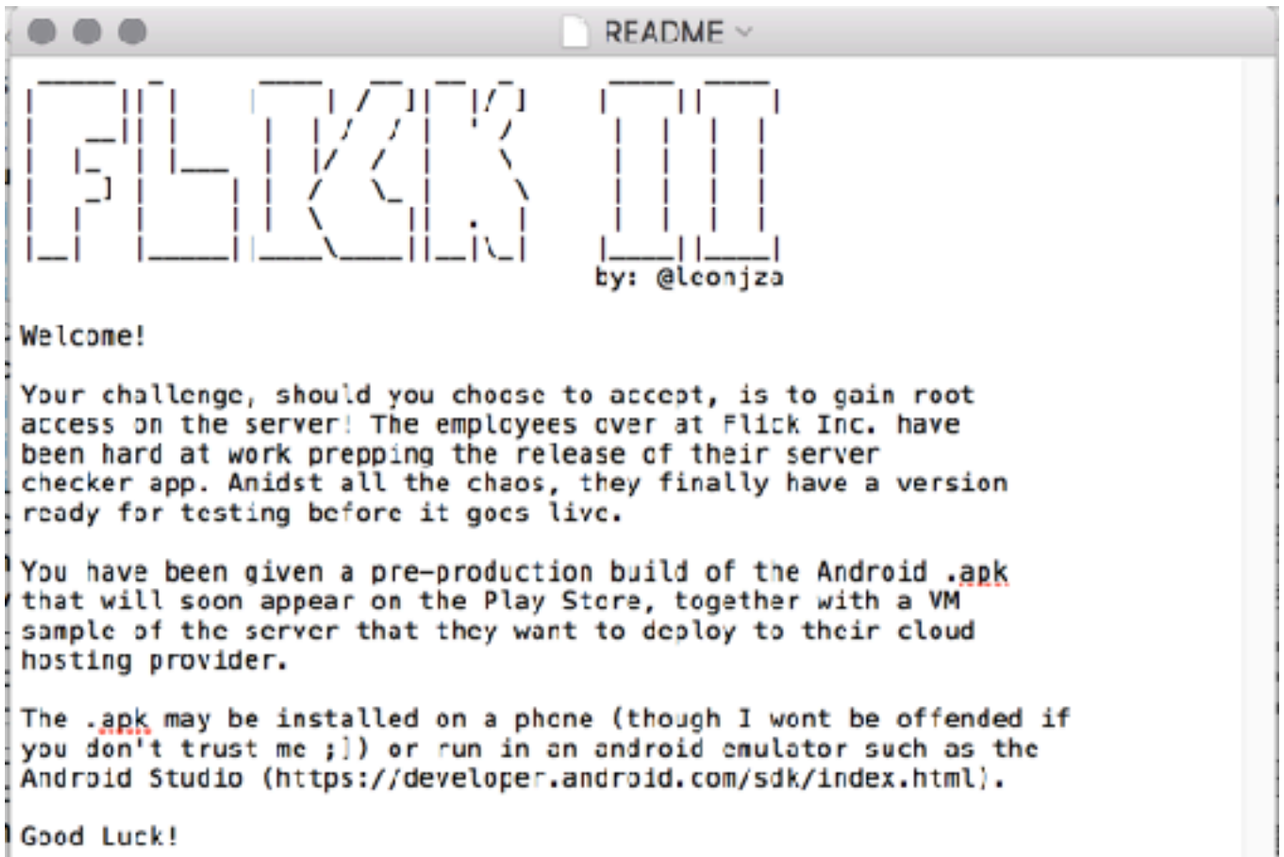
- Android APKs sind häufig in Java programmiert
- Es gibt auch in C entwickelte Android Applikationen
- Java APKs lassen sich einfach in SMALI Code decompilen
- Java APKs lassen sich darüber hinaus auch gut in Java zurück decompilen

VIELE ANWENDUNGSFÄLLE

- Demo:
  - Manipulation des Programmverhalten
  - Certificate Pinning deaktivieren

DEMO 1: CERTIFICATE PINNING DEAKTIVIEREN

- Beispiel: VM FlickII von Vulnhub:
- https://www.vulnhub.com/entry/flick-2,122/
- Step-by-Step writeup in meinem Blog:
- https://itunsecurity.wordpress.com/2016/12/25/vulnhub-flickii-a-different-approach-walkthrough-part1/



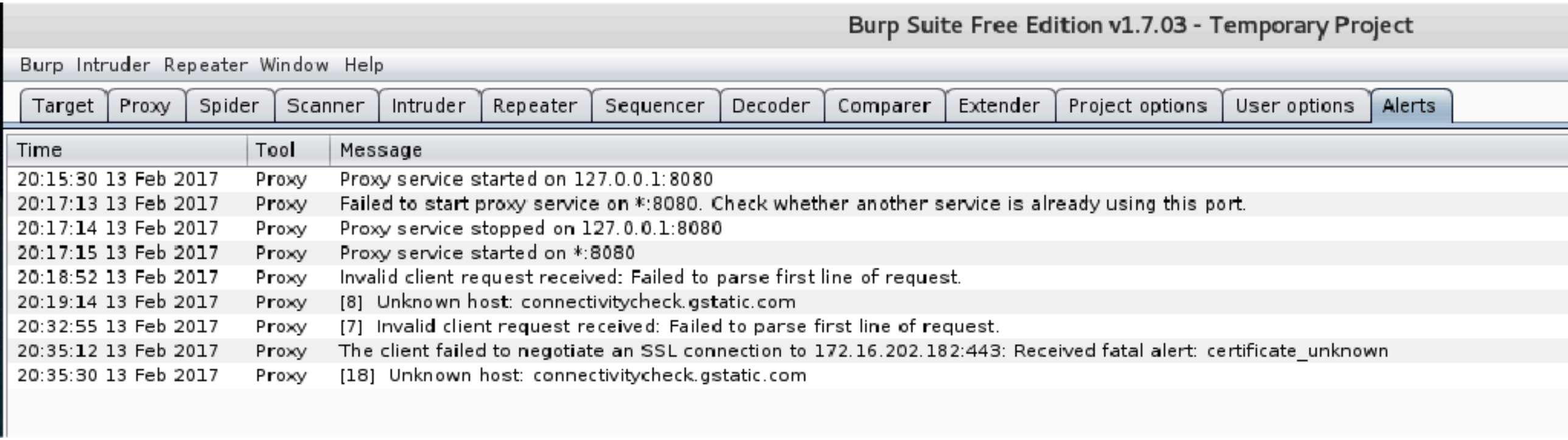
DEMO 1: CERTIFICATE PINNING DEAKTIVIEREN

- Step1:
- VM Hochfahren und untersuchen
- Android Studio Installieren und Emulator hochfahren:

```
$ ~/Library/Android/sdk/platform-tools/adb install ~/Desktop/vortrag-workdir/
flick-check-dist.apk
* daemon not running. starting it now on port 5037 *
* daemon started successfully *
1341 KB/s (1109803 bytes in 0.807s)
   pkg: /data/local/tmp/flick-check-dist.apk
Success
```

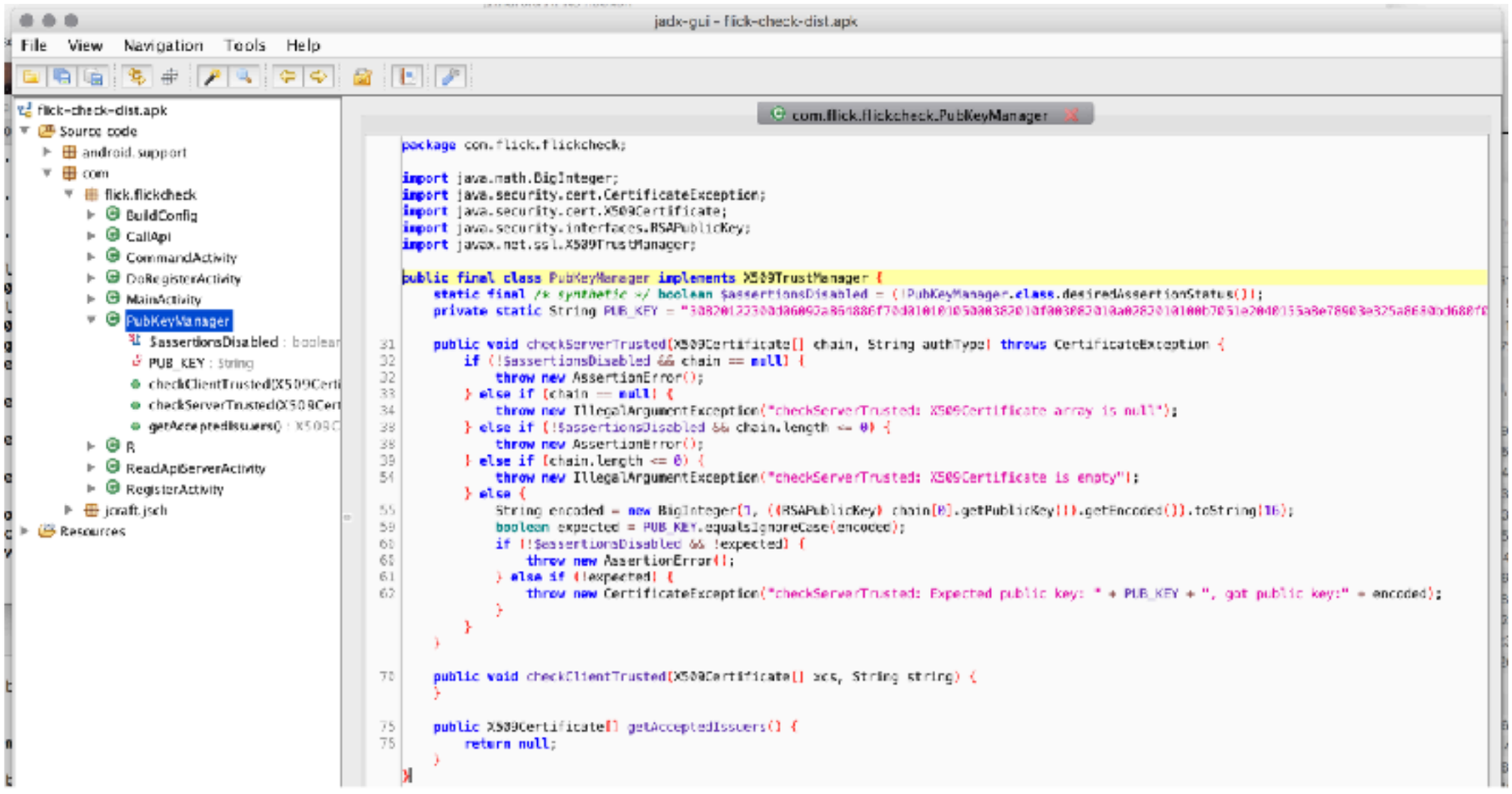
DEMO 1: CERTIFICATE PINNING DEAKTIVIEREN

- Step2:
- Problem: App mag nicht von Burp geMit(M)let werden!



DEMO 1: CERTIFICATE PINNING DEAKTIVIEREN

- Step3:
- Mit JadX rausfinden warum nicht!
- Certificate Pinning!



DEMO 1: CERTIFICATE PINNING DEAKTIVIEREN

HEY, U! YOUR KUNG FU NO GOOD!

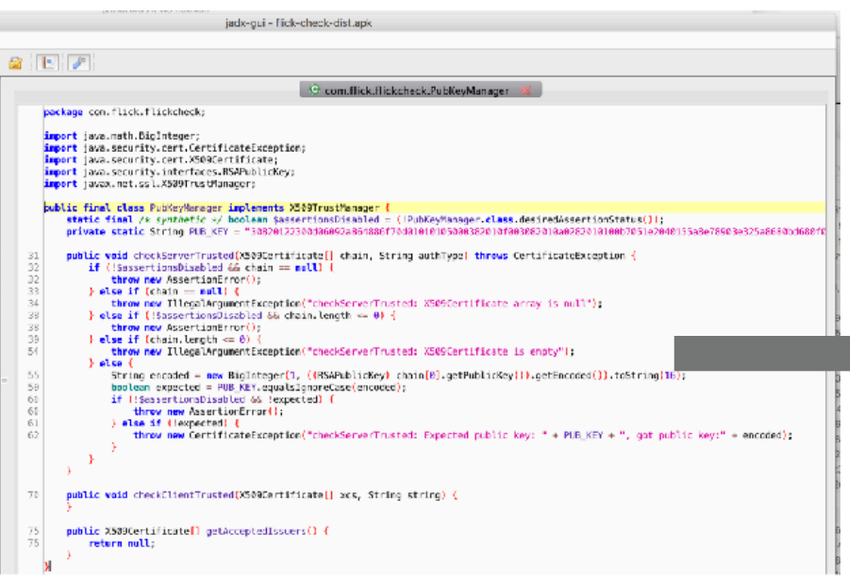
DEMO 1: CERTIFICATE PINNING DEAKTIVIEREN

- Step4:
- App mit apktool decompilen:

```
$ ~/Desktop/hackchallenge\ 2016/challenge2/apktool d flick-check-dist.apk
I: Using Apktool 2.2.1 on flick-check-dist.apk
I: Loading resource table...
I: Decoding AndroidManifest.xml with resources...
I: Loading resource table from file: ~/Library/apktool/framework/1.apk
I: Regular manifest package...
I: Decoding file-resources...
I: Decoding values */* XMLs...
I: Baksmaling classes.dex...
I: Copying assets and libs...
I: Copying unknown files...
I: Copying original files...
```

DEMO 1: CERTIFICATE PINNING DEAKTIVIEREN

- Step5:
- Entsprechende Smali Code Datei öffnen:




```
$ head flick-check-dist/smali/com/flick/flickcheck/PubKeyManager.smali
.class public final Lcom/flick/flickcheck/PubKeyManager;
.super Ljava/lang/Object;
.source "PubKeyManager.java"

# interfaces
.implements Ljavax/net/ssl/X509TrustManager;


# static fields
.field static final synthetic $assertionsDisabled:Z

$ head flick-check-dist/smali/com/flick/flickcheck/PubKeyManager.smali  -n
50
==> flick-check-dist/smali/com/flick/flickcheck/PubKeyManager.smali <==
.class public final Lcom/flick/flickcheck/PubKeyManager;
.super Ljava/lang/Object;
.source "PubKeyManager.java"

# interfaces
.implements Ljavax/net/ssl/X509TrustManager;
```

DEMO 1: CERTIFICATE PINNING DEAKTIVIEREN

- Step6:
- Logik des Certificate Managers verstehen:

```
public final class PubKeyManager implements X509TrustManager {
    static final /* synthetic */ boolean $assertionsDisabled = (!PubKeyManager.class.desiredAssertionStatus());
    private static String PUB_KEY = "30820122300d06092a864886f70d0101010500038201 0f003082010a0282010100b7051e2040155a8e78

31  public void checkServerTrusted(X509Certificate[] chain, String authType) throws CertificateException {
32      if (!$assertionsDisabled && chain == null) {
32          throw new AssertionError();
33      } else if (chain == null) {
34          throw new IllegalArgumentException("checkServerTrusted: X509Certificate array is null");
38      } else if (!$assertionsDisabled && chain.length <= 0) {
38          throw new AssertionError();
39      } else if (chain.length <= 0) {
54          throw new IllegalArgumentException("checkServerTrusted: X509Certificate is empty");
        } else {
```

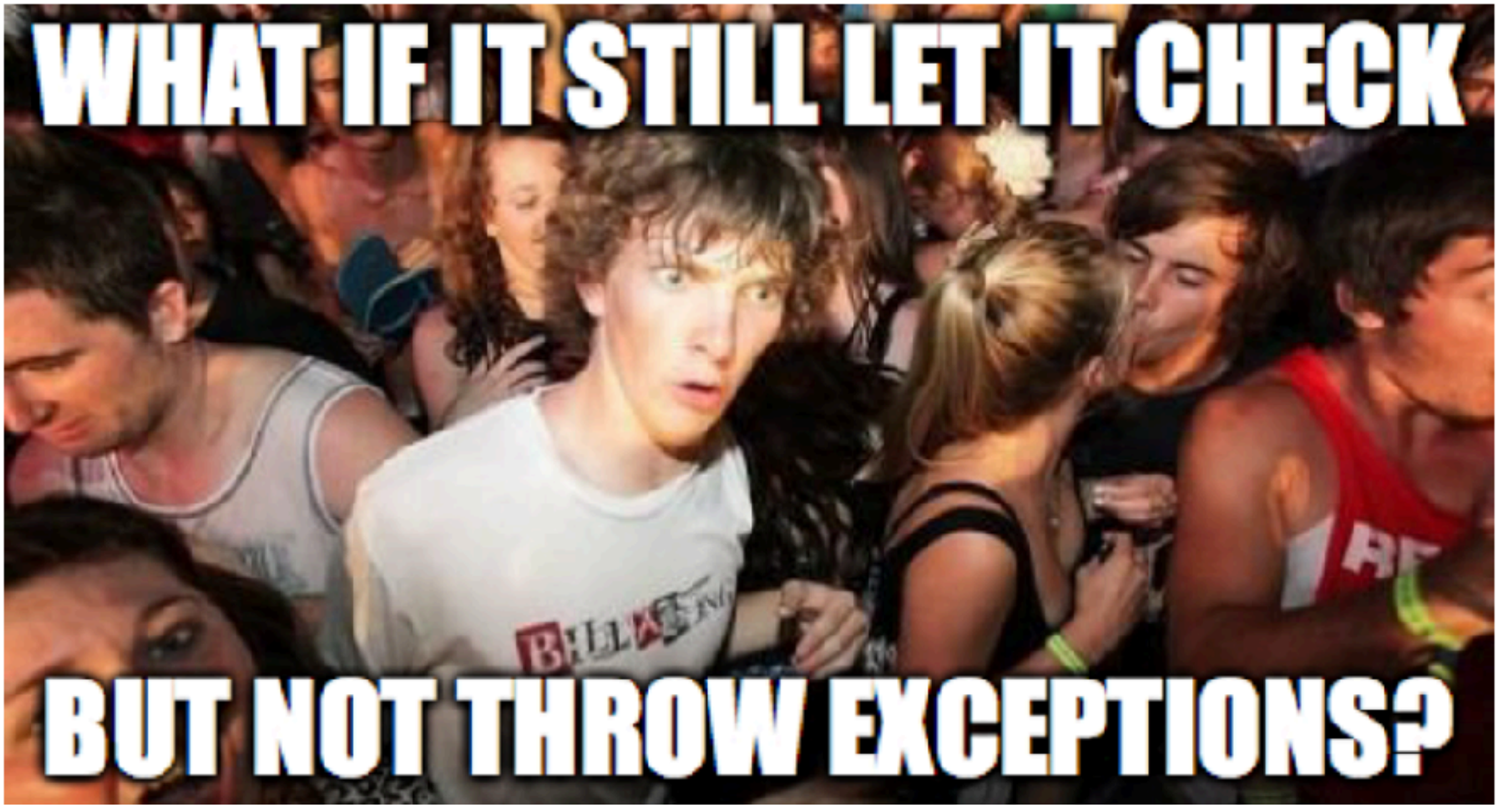
```
sget-boolean v3, Lcom/flick/flickcheck/PubKeyManager;->$assertionsDisabled:Z
if-nez v3, :cond_0
if-nez p1, :cond_0
new-instance v3, Ljava/lang/AssertionError;
invoke-direct {v3}, Ljava/lang/AssertionError;-><init>()V

throw v3

.line 33
```

DEMO 1: CERTIFICATE PINNING DEAKTIVIEREN

- Step7:
- Logik anpassen: Alle Vorkommnisse von „throw v3“ löschen



DEMO 1: CERTIFICATE PINNING DEAKTIVIEREN

- Step8:
- App wieder compilen!

```
~/Desktop/hackchallenge\ 2016/challenge2/apktool b ~/Desktop/vortrag-workdir/flick-check-dist/
I: Using Apktool 2.2.1
I: Checking whether sources has changed...
I: Smaling smali folder into classes.dex...
W: Unknown file type, ignoring: /Users/sebastianbrabetz/Desktop/vortrag-workdir/flick-check-dist/
smali/.DS_Store
W: Unknown file type, ignoring: /Users/sebastianbrabetz/Desktop/vortrag-workdir/flick-check-dist/
smali/android/.DS_Store
I: Checking whether resources has changed...
I: Building resources...
I: Building apk file...
I: Copying unknown files/dir...
```

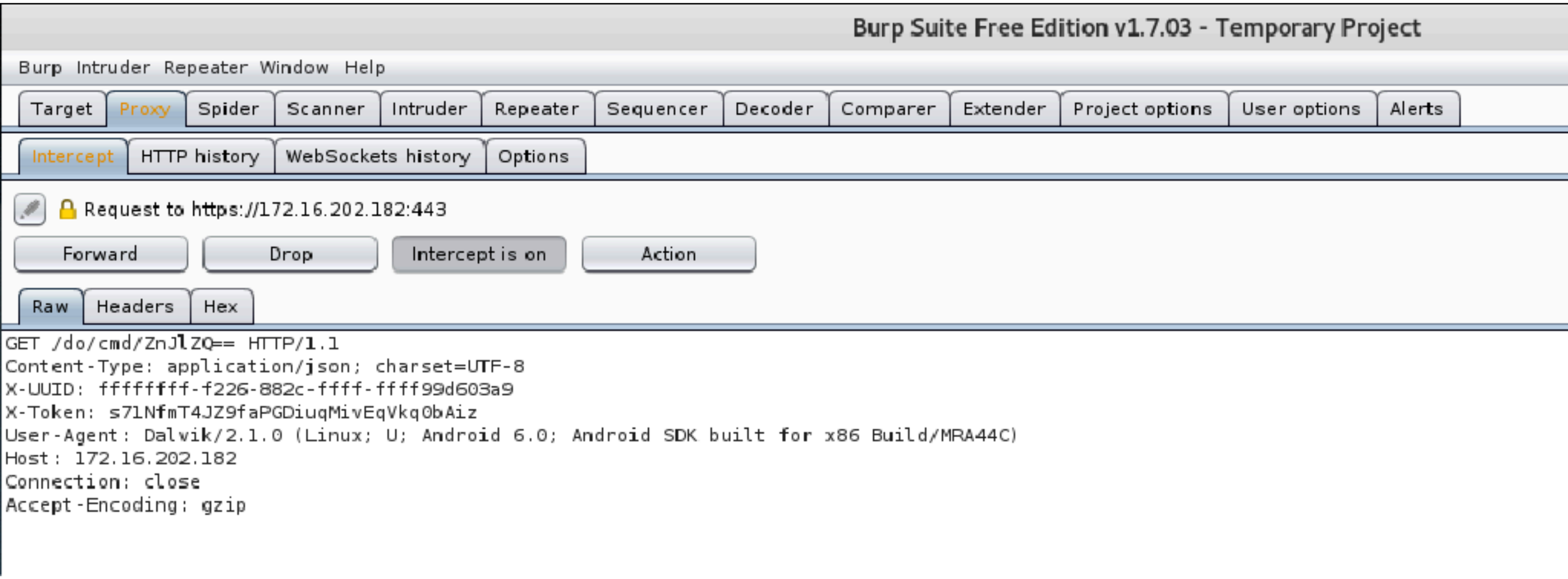
DEMO 1: CERTIFICATE PINNING DEAKTIVIEREN

- Step8:
- App signieren:
- Falls kein Keystore vorhanden unter OSX einfach anzulegen:
- *keytool -genkey -keystore (name).keystore -validity 10000 -alias (name)*

```
$ jarsigner -keystore ~/Desktop/hackchallenge\ 2016/challenge2/test.keystore -verbose /Users/
sebastianbrabetz/Desktop/vortrag-workdir/flick-check-dist/dist/flick-check-dist.apk test
Enter Passphrase for keystore:
   adding: META-INF/MANIFEST.MF
   adding: META-INF/TEST.SF
   adding: META-INF/TEST.DSA
  signing: AndroidManifest.xml
  signing: classes.dex
  signing: res/anim/abc_fade_in.xml
...
  signing: res/mipmap-mdpi-v4/ic_launcher.png
  signing: res/mipmap-xhdpi-v4/ic_launcher.png
  signing: res/mipmap-xxhdpi-v4/ic_launcher.png
  signing: resources.arsc
jar signed.
```

DEMO 1: CERTIFICATE PINNING DEAKTIVIEREN

- Step9:
- App löschen und gehax0rte Version installieren:



DEMO 1: CERTIFICATE PINNING DEAKTIVIEREN

- Step10:
- Hack the planet!

ZU GUTER LETZT: INTERESSANTE DINGE IN EXISTIERENDEN APKS FINDEN

- Hardcoded credentials:

```
<string name="private_use_pref_label">Active account</string>
<string name="s1">slkdnlvofiwelsknav</string>
<string name="s2">weoru094w8cwnvksf3</string>
<string name="s3">2-0sfls0-klnslf9vd</string>
<string name="search_authority">de.konrad.AddressSuggestionsProvider</string>
<string name="search_label">search label</string>
<string name="signup_url">https://xml.dbcarsharing-buchung.de/externe_daten/2denker/anmeldung.html?os=endroid&amp;tuser=%1$s&amp;tpass=%2ss</string>
<string name="soap_url">https://xml.dbcarsharing-buchung.de/hal2_cabserver/hal2_cabserver_3.php</string>
<string name="store_password_label">Store password</string>
<string name="technical_password">a2027c40b1</string>
<string name="technical_user">t_andoid_kon</string>
<string name="version_entry">Version</string>
<string name="version_label">Version</string>
</resources>
```