안드로이드 루팅 탐지 우회

Frida 후킹

211014 이유경

Contents

1. 루팅이란?
   - 1-1. 루팅 정의
   - 1-2. 루팅 탐지 분석
   - 1-3. 루팅 탐지 기술
   - 1-4. 루팅 탐지 우회 대응 방법
2. 후킹이란?
   - 2-1. 후킹 정의
   - 2-2. Frida란?
3. 루팅 탐지 우회
   - 3-1. Manifest.xml
   - 3-2. onCreate()
   - 3-3. 루팅 탐지 함수
   - 3-4. 루팅 탐지 우회

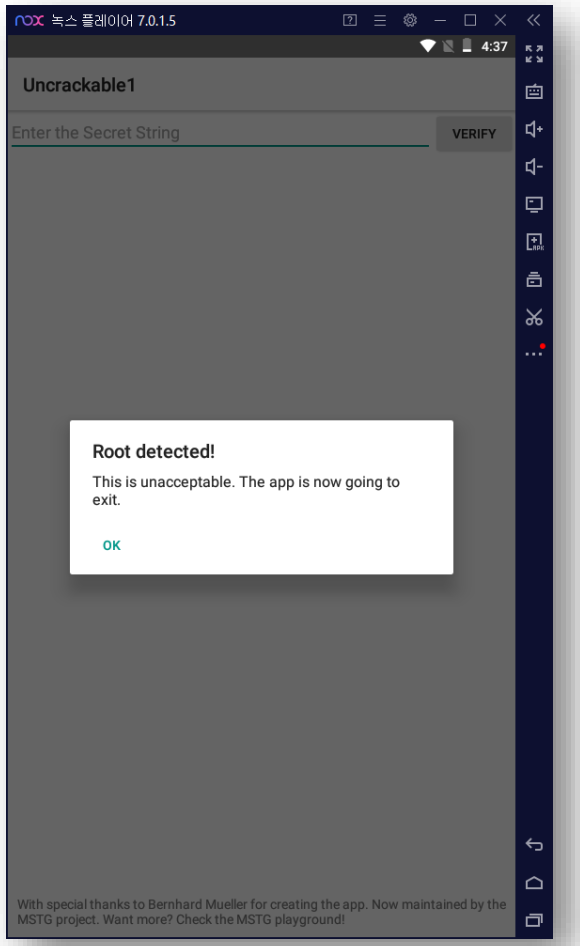
1-1. 루팅 정의

안드로이드 운영체제 상에서 **최상위 권한(root)을 얻음**으로

해당 기기의 생산자 또는 판매자 측에서 걸어 놓은 제약을

해제하는 행위

1-2. 루팅 탐지 분석



루팅 탐지된 화면 ("Root detected!")

↓

OK 버튼 누르면 어플리케이션 종료

↓

어플리케이션이 종료되는 코드를 확인하기 위해

jadx를 통해 디컴파일하여 분석

1-3. 루팅 탐지 기술

1. Su 탐지
2. 프로세스 리스트 탐지
3. 루팅 관련 어플 설치 여부 확인
4. 폴더 권한 확인
5. 테스트키 확인

1-3. 루팅 탐지 기술

1. Su 탐지
2. 프로세스 리스트 탐지
3. 루팅 관련 어플 설치 여부 확인
4. 폴더 권한 확인
5. 테스트키 확인

우회

후킹을 통해 정상적인 앱을 실행했을 때
반환되는 반환 값으로 변조를 하거나,
특정 조건이나 값에 따른
루틴을 처리하지 않고 우회하도록 하여
루팅 검증을 우회하는 방법

1-3. 루팅 탐지 우회 대응 방법

루팅을 방지하는 부분의 코드가 노출되지 않게 한다.

↓

코드 난독화 혹은 apk 무결성 검증 등을 실행

2-1. 후킹 정의

특정 함수 코드를 **가로채서** 원하는 행위를 한 뒤
원래의 코드로 돌려주는 기법

- 함수 호출을 가로채서 모니터링 가능
- 완전히 다른 함수로 대체하는데 사용 가능

후킹은 앱이 실행되면서 특정 메소드에 hook을 걸어
본인 마음대로 메소드의 리턴 값, 메소드의 내용을 패치 할 수 있다.

2-2. Frida란?

Python 기반의 라이브러리로 Native App을 Hooking 하여
어플리케이션을 분석할 수 있게 해 주는 툴

3-1. Manifest.xml

```
package="owasp.mstg.uncrackable1"
```

```
android:name="sg.vantagepoint.uncrackable1.MainActivity"
```

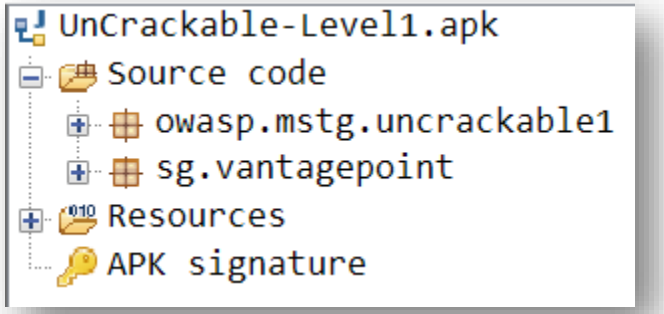
① apk easy tool로 Uncrackable-Level1 디컴파일

↓

② 디컴파일된 파일 중, AndroidManifest.xml에서

package명과 MainActivity 확인

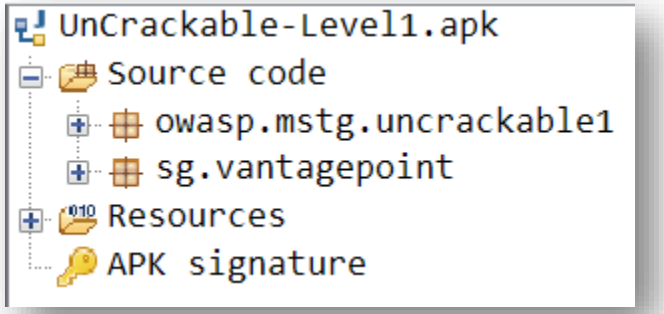
3-2. onCreate()

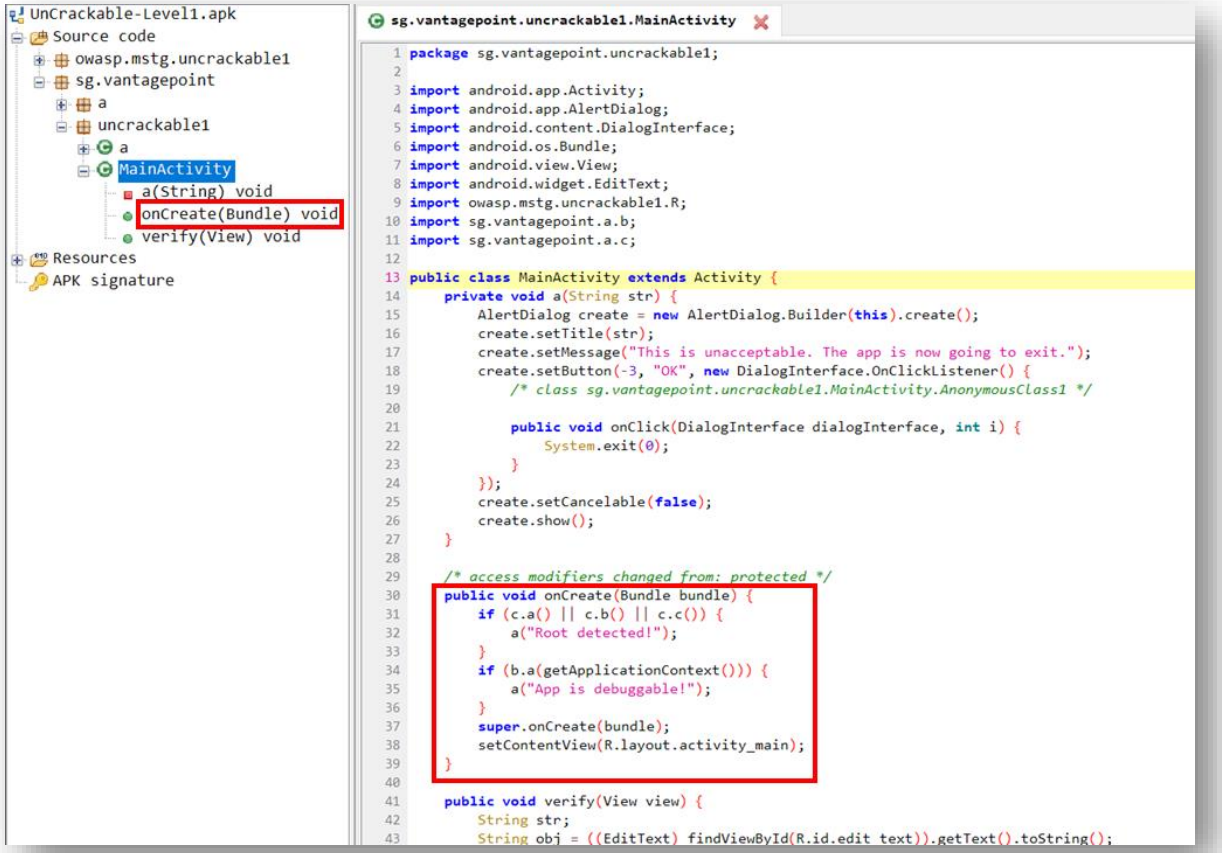


MainActivity의 경로: *sg.vantagepoint.uncrackable1.MainActivity*

3-2. onCreate()

MainActivity의 경로: *sg.vantagepoint.uncrackable1.MainActivity*





3-2. onCreate()

onCreate: 가장 먼저 실행되는 함수

```
public void onCreate(Bundle bundle) {
    if (c.a() || c.b() || c.c()) {
        a("Root detected!");
    }
    if (b.a(getApplicationContext())) {
        a("App is debuggable!");
    }
    super.onCreate(bundle);
    setContentView(R.layout.activity_main);
}
```

If문을 통해 a, b, c 메서드로 루팅 탐지

```
public void onCreate(Bundle bundle) {
    if (c.a() || c.b() || c.c()) {
        a("Root detected!");
    }
    if (b.a(getApplicationContext())) {
        a("App is debuggable!");
    }
    super.onCreate(bundle);
    setContentView(R.layout.activity_main);
}
```

3-3. 루팅 탐지 함수

```
public class c {
    public static boolean a() {
        for (String str : System.getenv("PATH").split(":")) {
            if (new File(str, "su").exists()) {
                return true;
            }
        }
        return false;
    }

    public static boolean b() {
        String str = Build.TAGS;
        return str != null && str.contains("test-keys");
    }

    public static boolean c() {
        for (String str : new String[]{"/system/app/Superuser.apk", "/s
            if (new File(str).exists()) {
                return true;
            }
        }
        return false;
    }
}
```

루팅하면 su 바이너리 파일 생성됨.

su가 탐지되면 루팅됐다고 판단

기기를 루팅하면 "test-keys"라는 태그 값이 등록됨.

해당 값이 존재하면 루팅됐다고 판단

/system/app 폴더에 루팅에 쓰이는 apk가 있는지 확인

apk가 존재하면 루팅됐다고 판단

3-4. 루팅 탐지 우회

```
1  //test.js
2
3  console.log("[+] Running Hook");
4  Java.perform(function() {
5      var hook = Java.use("sg.vantagepoint.a.c");
6      hook.a.implementation = function() {
7          console.log("[+] Hooking a()");
8          return false;
9      }
10     hook.b.implementation = function() {
11         console.log("[+] Hooking b()");
12         return false;
13     }
14     hook.c.implementation = function() {
15         console.log("[+] Hooking c()");
16         return false;
17     }
18 });
```

Java.perform(): 불러오는 값에 함수를 정의해서 java Method 후킹

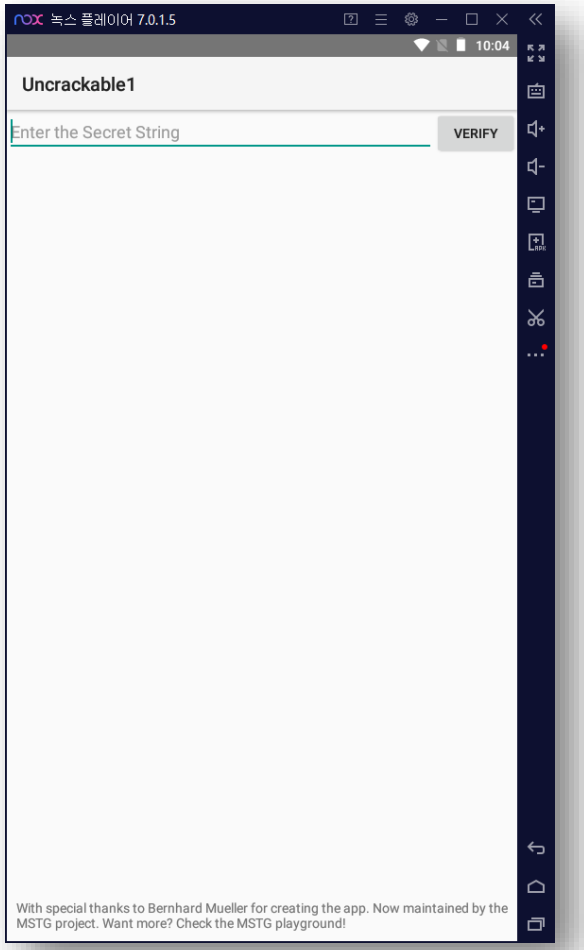
Java.use(): 후킹할 class에 대한 객체를 받아올 수 있음

.implementation: sg.vantagepoint.a.c 메서드를 재정의

3-4. 루팅 탐지 우회

frida.exe -U -f owasp.mstg.uncrackable1 -l D:₩dla₩frida_test₩test.js --no-pause

```
D:₩Program Files₩Python₩Python39₩Scripts>frida.exe -U -f owasp.mstg.uncrackable1 -l D:₩dla₩frida_test₩test.js --no-pause
     ____
    / _  |   Frida 15.1.1 - A world-class dynamic instrumentation toolkit
   | (_| |
    > _  |   Commands:
t  /_/ |_|       help      -> Displays the help system
   . . . .       object?   -> Display information about 'object'
t  . . . .       exit/quit -> Exit
   . . . .
t  . . . .   More info at https://frida.re/docs/home/
Spawning `owasp.mstg.uncrackable1`...
[+] Running Hook
Spawned `owasp.mstg.uncrackable1`. Resuming main thread!
[SM-G965N::owasp.mstg.uncrackable1]-> [+] Hooking a()
[+] Hooking b()
[+] Hooking c()
[SM-G965N::owasp.mstg.uncrackable1]->
```



THANK YOU

QUESTION