What Mobile Ads Know About Mobile Users

Sooel Son

joint work with
Daehyeok Kim and Vitaly Shmatikov

Overview

- Background
  - Mobile advertising library
  - Attack model: malicious advertiser
- Information available to the attacker
  - Local file resources in Android devices
    - **Inference attack** via local resource oracle
    - **Direct information leakage attack**
    - **Proposed defenses**
  - User trajectories
- Summary

1.8 million

apps in Google Play Store

source: AppBrain

41% include at least one mobile advertising library

source: AppBrain

Every third

ad-supported app includes multiple advertising libraries

source: Shekhar et al. (USENIX Security 2012)

Web


YOUR AD HERE
Web browser

Mobile

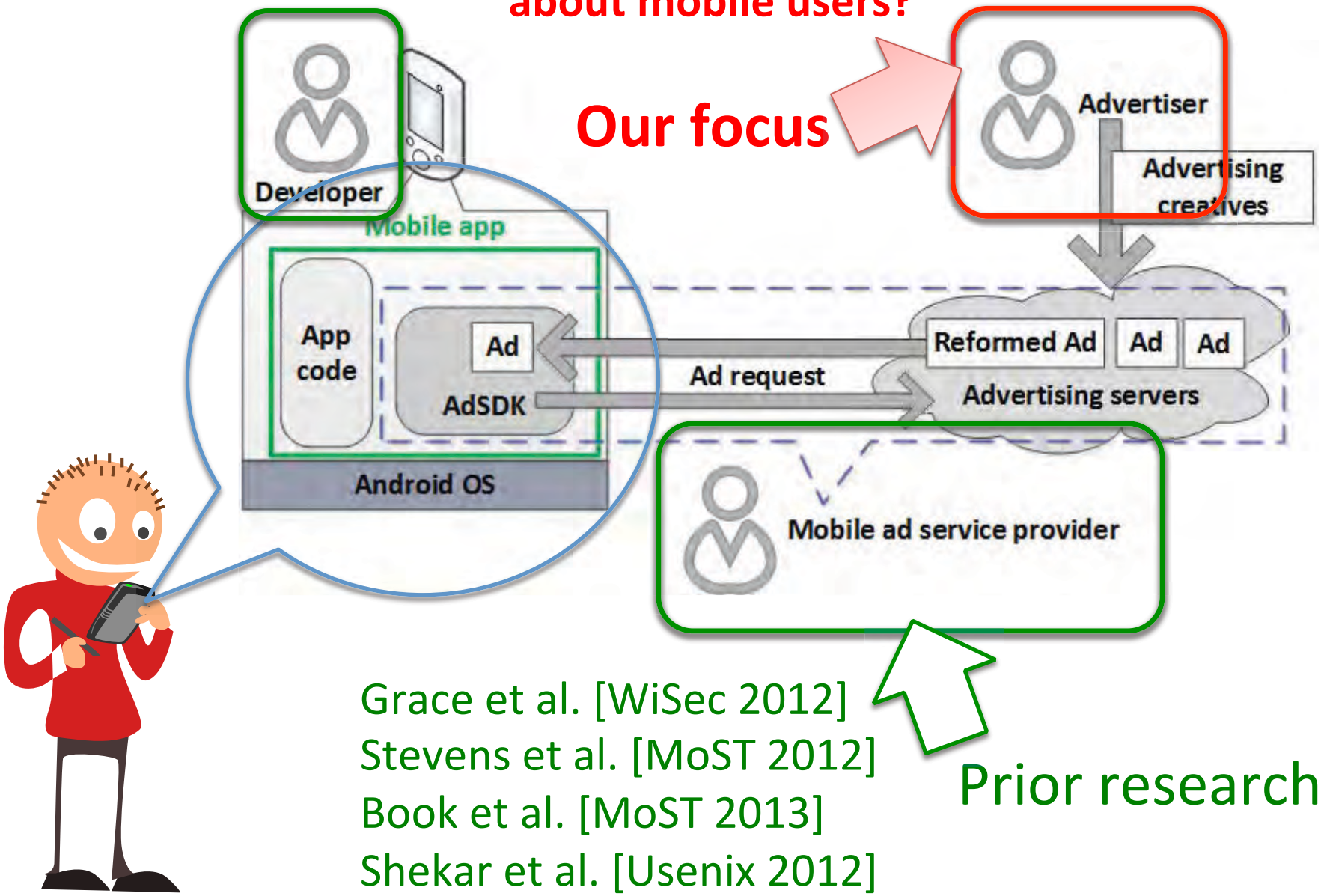

YOUR AD HERE
admob
AdMarvel
mopub
airpush
Ad library
Mobile app

Mobile


YOUR AD HERE
admob
AdMarvel
mopub
airpush
Ad library
Mobile app


5:54
search dictionary
Word of the Day
July 23, 2015
hierogram
noun [hahy-er-uh-gram]
a sacred symbol, as an emblem, pictograph, or the like.
Slideshows
Speak with Savoir-faire: 10 French Loanwords
hone in vs. home in
Blog
Hone In vs. Home In
amazon
Download the free Amazon app
Get notified when your order arrives

What can malicious advertisers learn about mobile users?



Grace et al. [WiSec 2012]
Stevens et al. [MoST 2012]
Book et al. [MoST 2013]
Shekar et al. [Usenix 2012]
....

Prior research

Advertising services

- Large businesses
  - AdMob (Google),
    Mopub (Twitter),
    AirPush, many others
- Provide **AdSDK libraries** to 100,000s of developers
- Millions of $ in revenue
- Reputation at stake

Advertisers

- Lots of fly-by-night operators
- Ads resold via auctions, brokers, exchanges
- No reputation at stake, no accountability
- Dynamic filtering and sanitization are hard

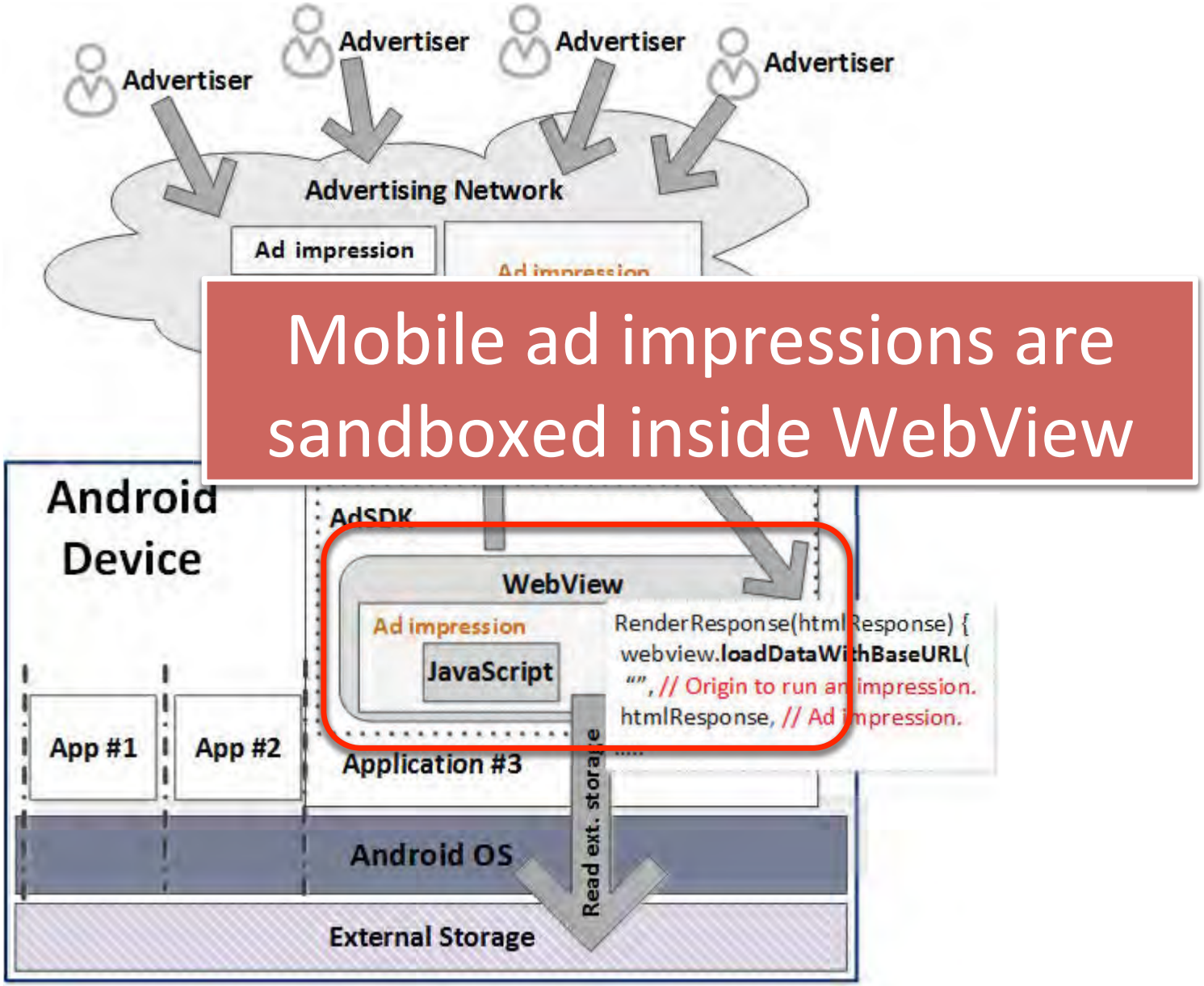
Ad libraries must protect users from malicious advertising

Android AdSDK Software Stack

- App developers include AdSDKs, add permissions for AdSDKs, repackage apps

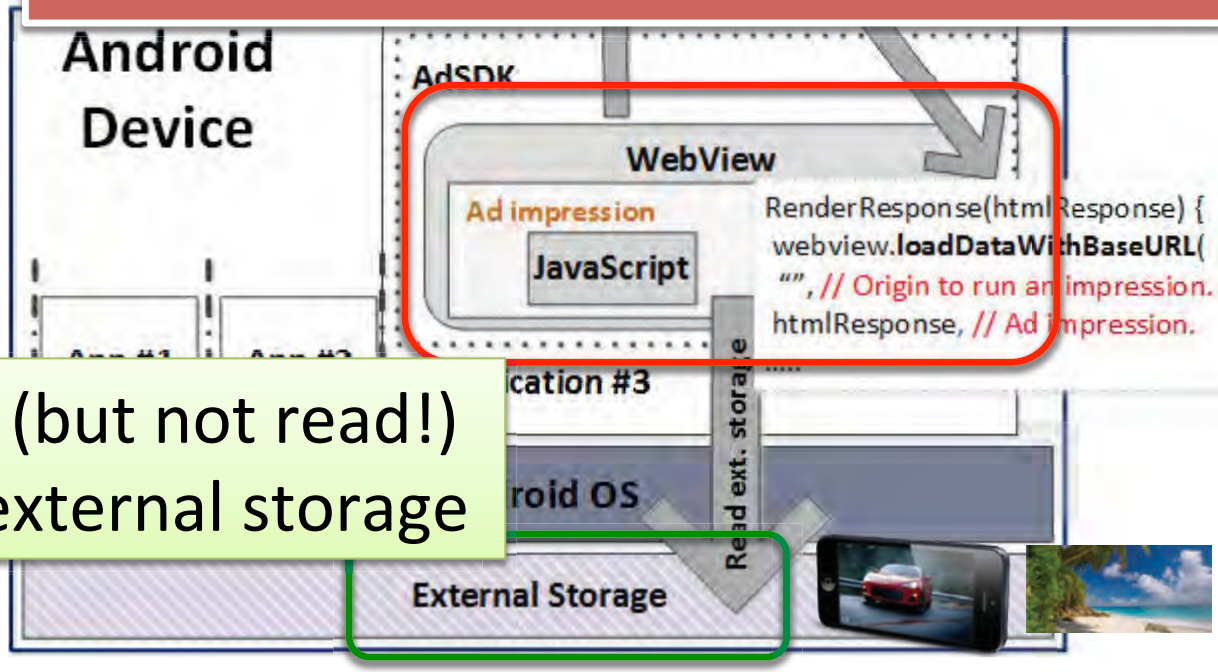
App and AdSDK share the same privileges

App and Ad should NOT share the same privileges


Advertiser
Advertiser
Advertiser
Advertiser
Advertising Network
Ad impression
Mobile ad impressions are sandboxed inside WebView
Android Device
AdSDK
WebView
Ad impression
JavaScript
RenderResponse(htmlResponse) {
webview.loadDataWithBaseURL(
"", // Origin to run an impression.
htmlResponse, // Ad impression.
App #1
App #2
Application #3
Android OS
Read ext. storage
External Storage

Standard Web same origin policy:

JavaScript in a mobile ad cannot read or write content from other origins



… can load (but not read!) files from external storage

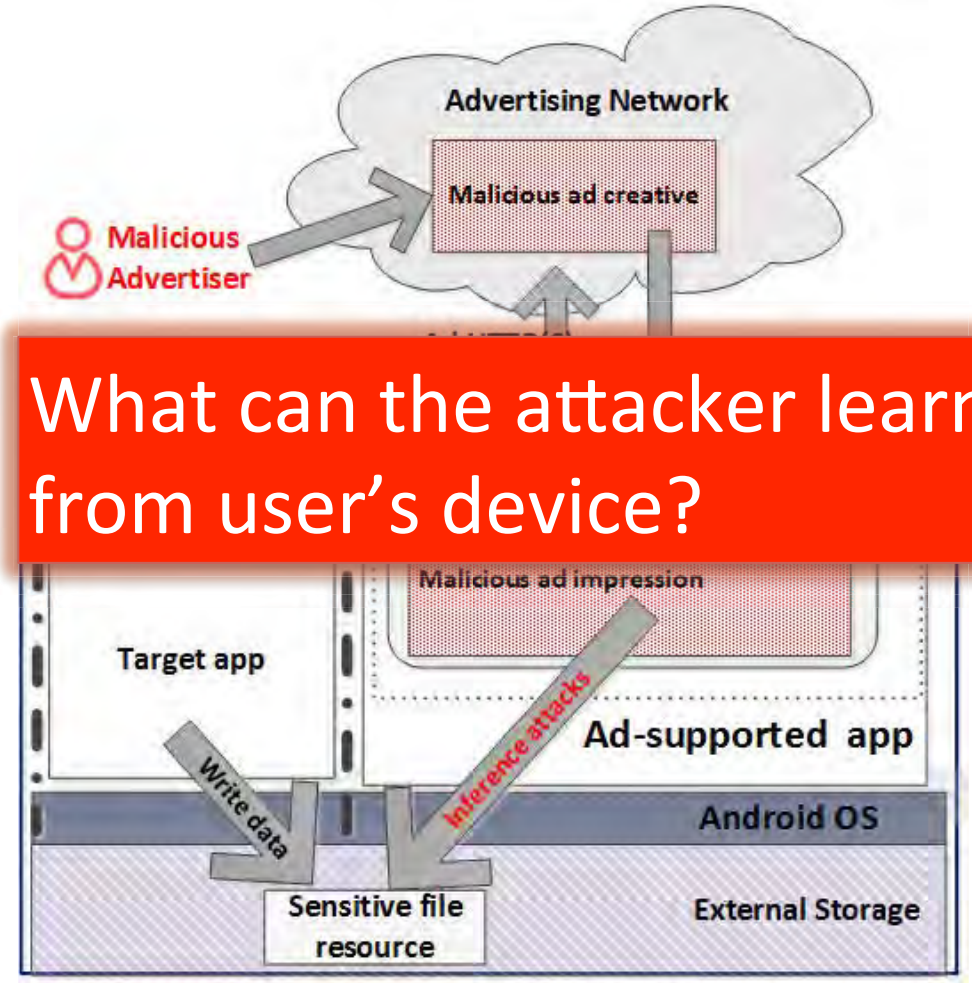
Android External Storage

- Can be read by any app with appropriate permissions
- Media-rich mobile ads require access to external storage to cache images, video

iab. MRAID 2.0

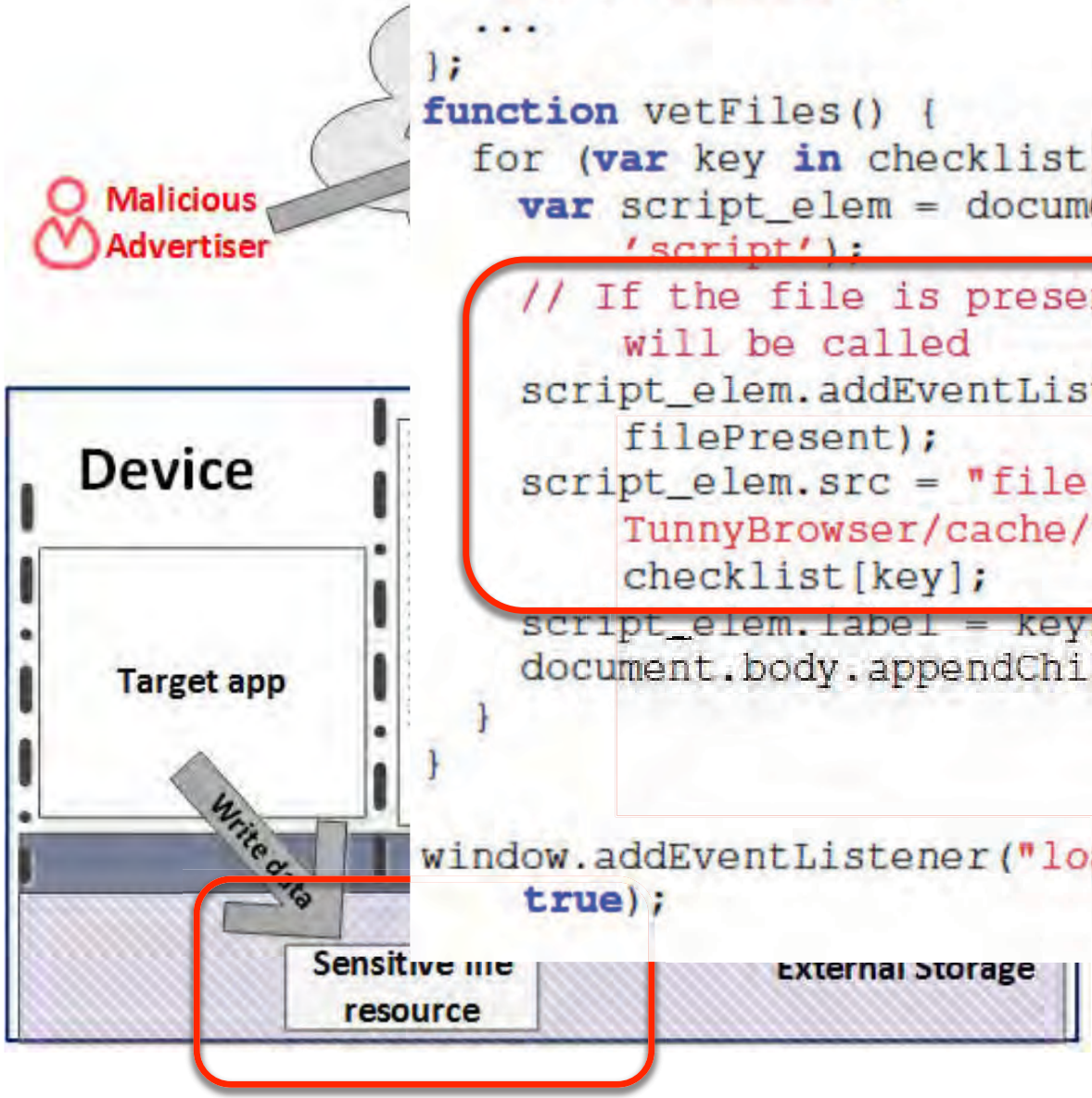
- Very weak access control for external storage
  - Any app can read any other app's files
  - But mobile ads are not apps. **Same origin policy = untrusted JavaScript cannot read ext-storage files … but can attempt to load them**

Attack Model



Malicious advertiser

- Cannot install apps
- Cannot observe user's network traffic
- Only payload: Ads

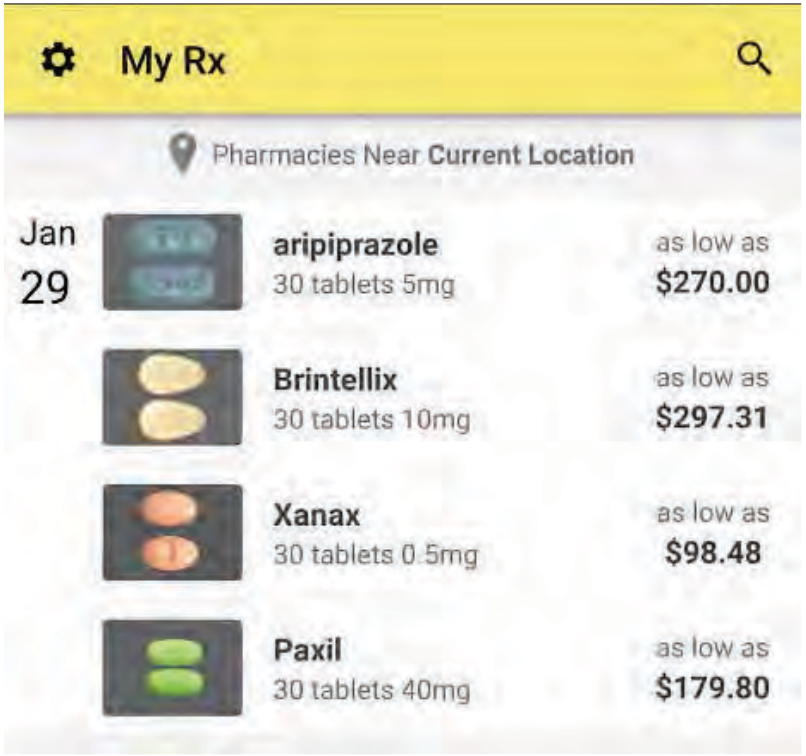


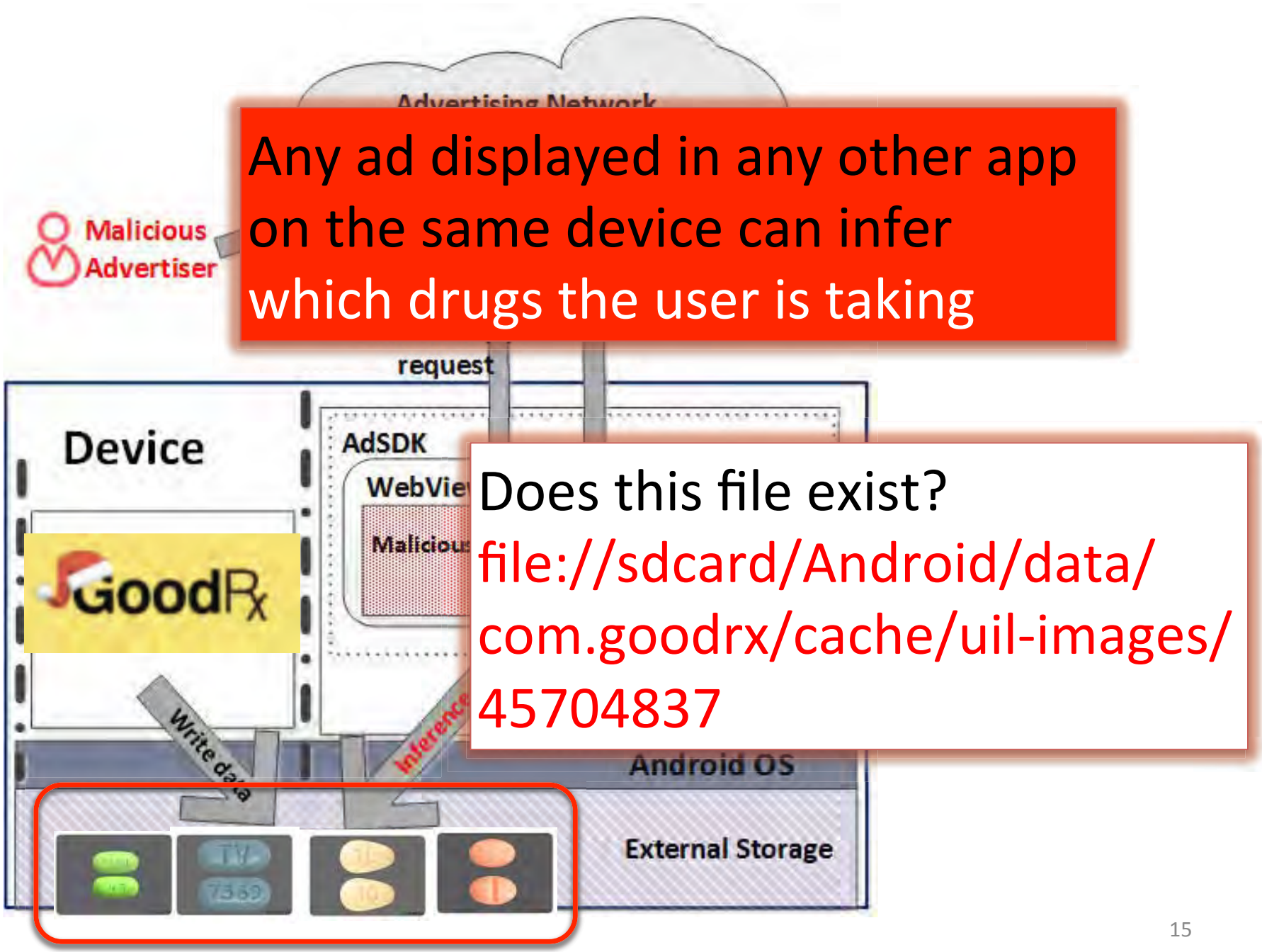

```
var checklist = {
  'DMV': '1645feb7',
  ...
};
function vetFiles() {
  for (var key in checklist) {
    var script_elem = document.createElement(
        'script');
    // If the file is present, filePresent
       will be called
    script_elem.addEventListener("load",
        filePresent);
    script_elem.src = "file:///sdcard/
        TunnyBrowser/cache/webviewCache/" +
        checklist[key];
    script_elem.label = key;
    document.body.appendChild(script_elem);
  }
}

window.addEventListener("load", vetFiles,
    true);
```

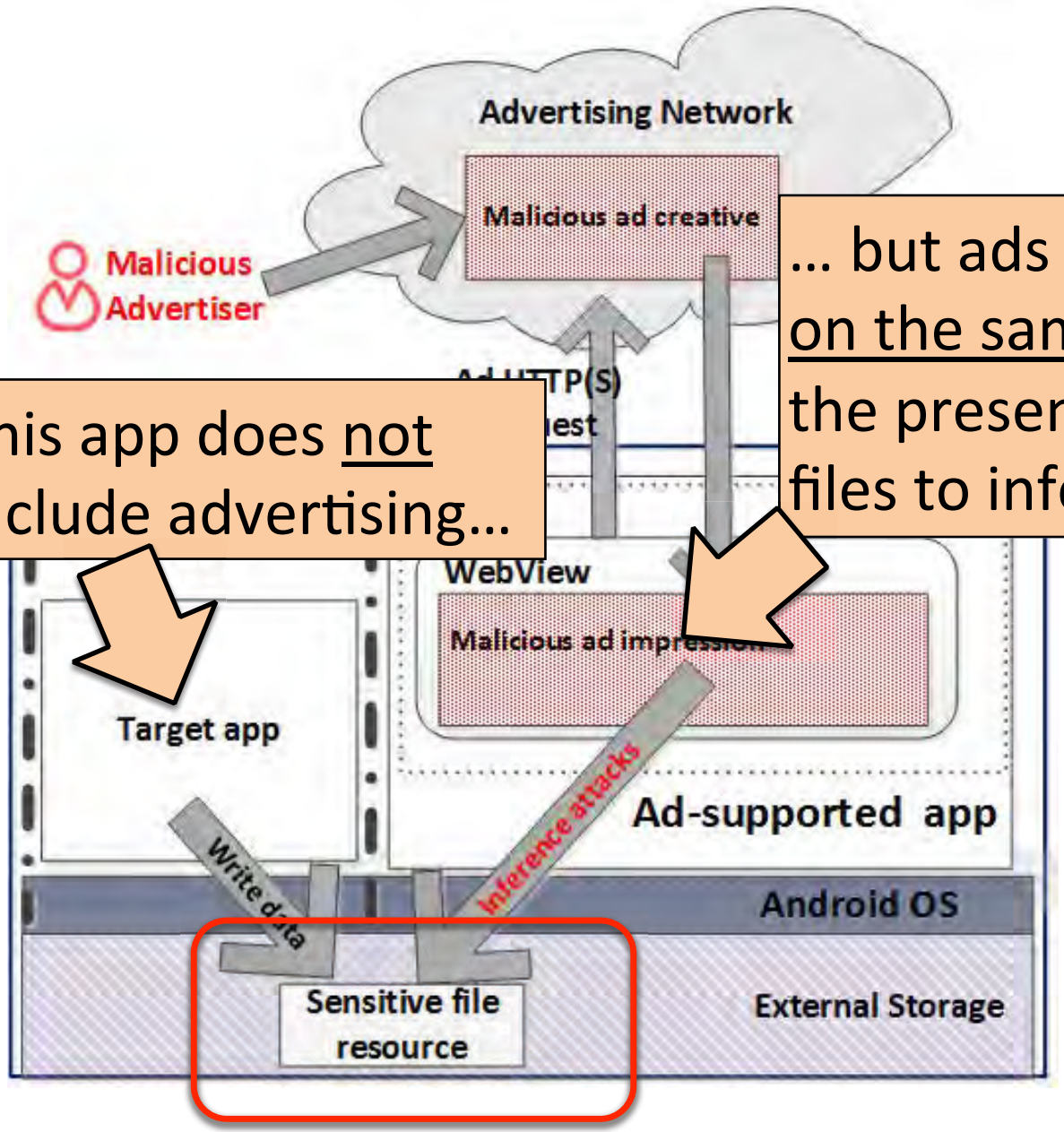
App for finding pharmacies, comparing drug prices (1 to 5 million installs in Google Play Store)

Bookmark functionality:

thumbnail images of drugs that the user searched for cached in external storage




Advertising Network
Malicious Advertiser
Any ad displayed in any other app on the same device can infer which drugs the user is taking
request
Device
AdSDK
WebVie
Malicious
GoodRx
Does this file exist?
file://sdcard/Android/data/com.goodrx/cache/uil-images/45704837
Write data
Inference
Android OS
External Storage



This app does not include advertising…

… but ads shown in any app on the same device can use the presence of its cached files to infer user's secrets

Does not violate same origin policy

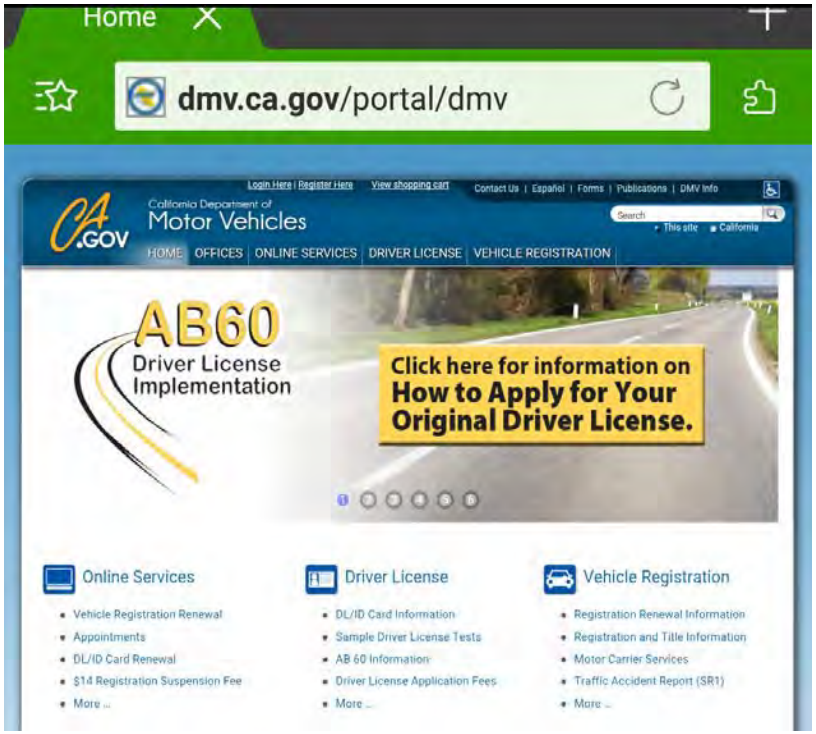
Why this Inference is Possible?

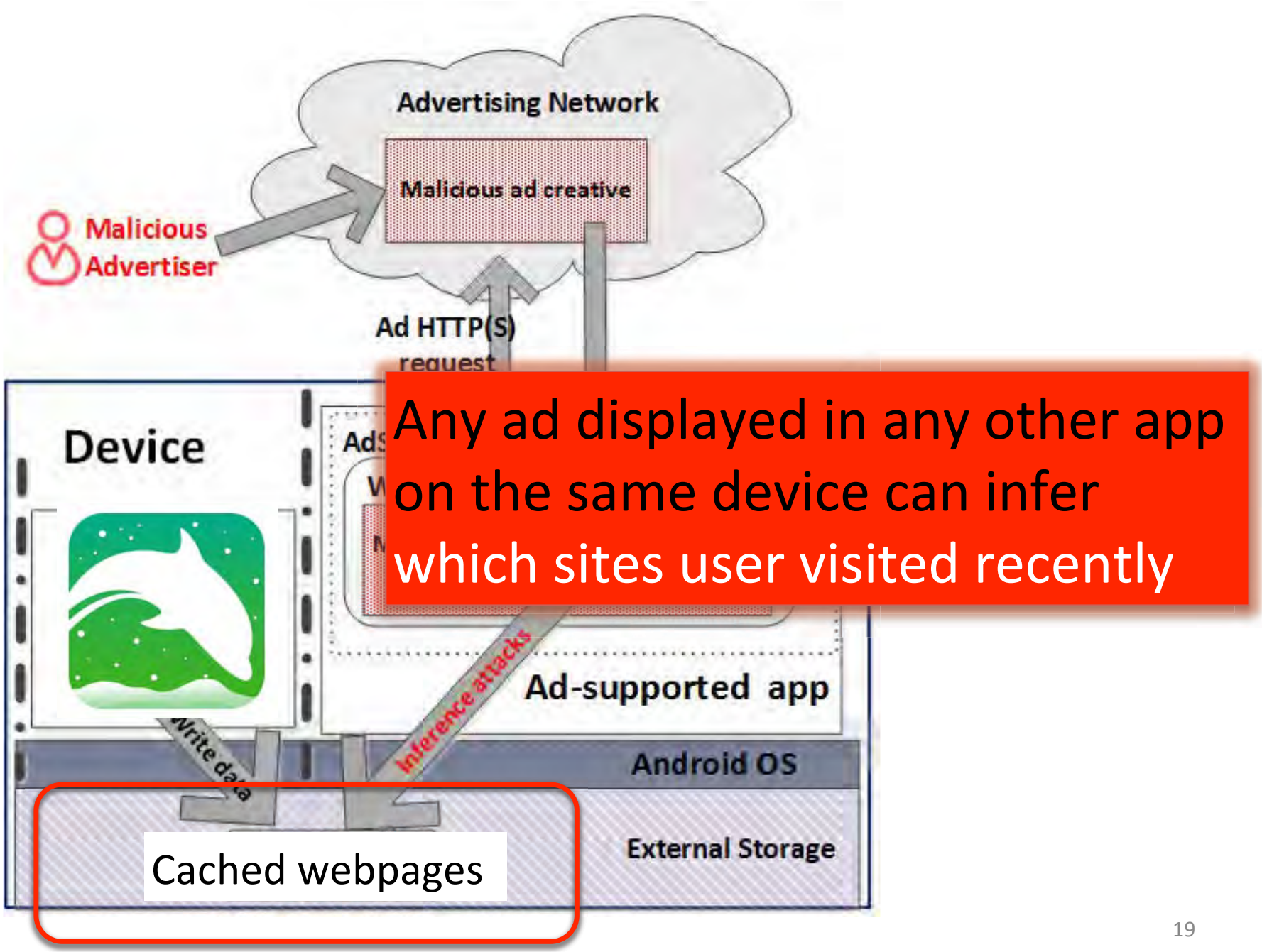
- **Read** vs. **Load** resources from different origins in JavaScript
  - **Read**: accessing actual contents of a resource.
  - **Load**: attaching a resource to the DOM object, not accessing its content.
- SOP prevents JavaScript in Ads from ***reading*** a cross-origin resource.
- However, ***loading*** a cross-origin resource is not prohibited.

Dolphin mobile browser

(50 to 100 million installs in Google Play Store)

To reduce bandwidth usage and response time, caches fetched images, HTML, and JavaScript in external storage





Any ad displayed in any other app on the same device can infer which sites user visited recently

Direct Information Leakage

- Malicious advertiser can read (not just load) all resources in external storage
- SetAllowUniversalAccessFromFileURLs
- SetAllowFileAccessFromFromURLs
  - Default is false since Android 4.0
  - Once enabled, it allows reading local resources from any file scheme URL
- D.Wu and R.Chang [ISC 2014, MoST 2015]

Our Study

- Several major Android advertising libraries

- “Local resource oracle” present in all of them
- All acknowledged the issue,
  several fixed in their latest AdSDK releases

Defenses for AdSDK developers

- Blocking any file access
  - WebSettings.SetAllowFileAccess(**false**)
  - Limit direct access to files

Defenses for AdSDK developers (2)

- Implement home-brewed ACLs

```
public WebResourceResponse shouldInterceptRequest(
                                        WebView view, String Url) {
  Uri givenUri = Uri.parse(Url);
  string givenPath = givenUri.getPath();
  if (givenPath.starsWith(JAIL_PREFIX)) {
    // If givenUrl is a subdirectory of JAIL_PREFIX, request is granted
    …
  }
}
```

  - ACLs based on file paths
  - Do not block other links to local resources

Tracking in Android

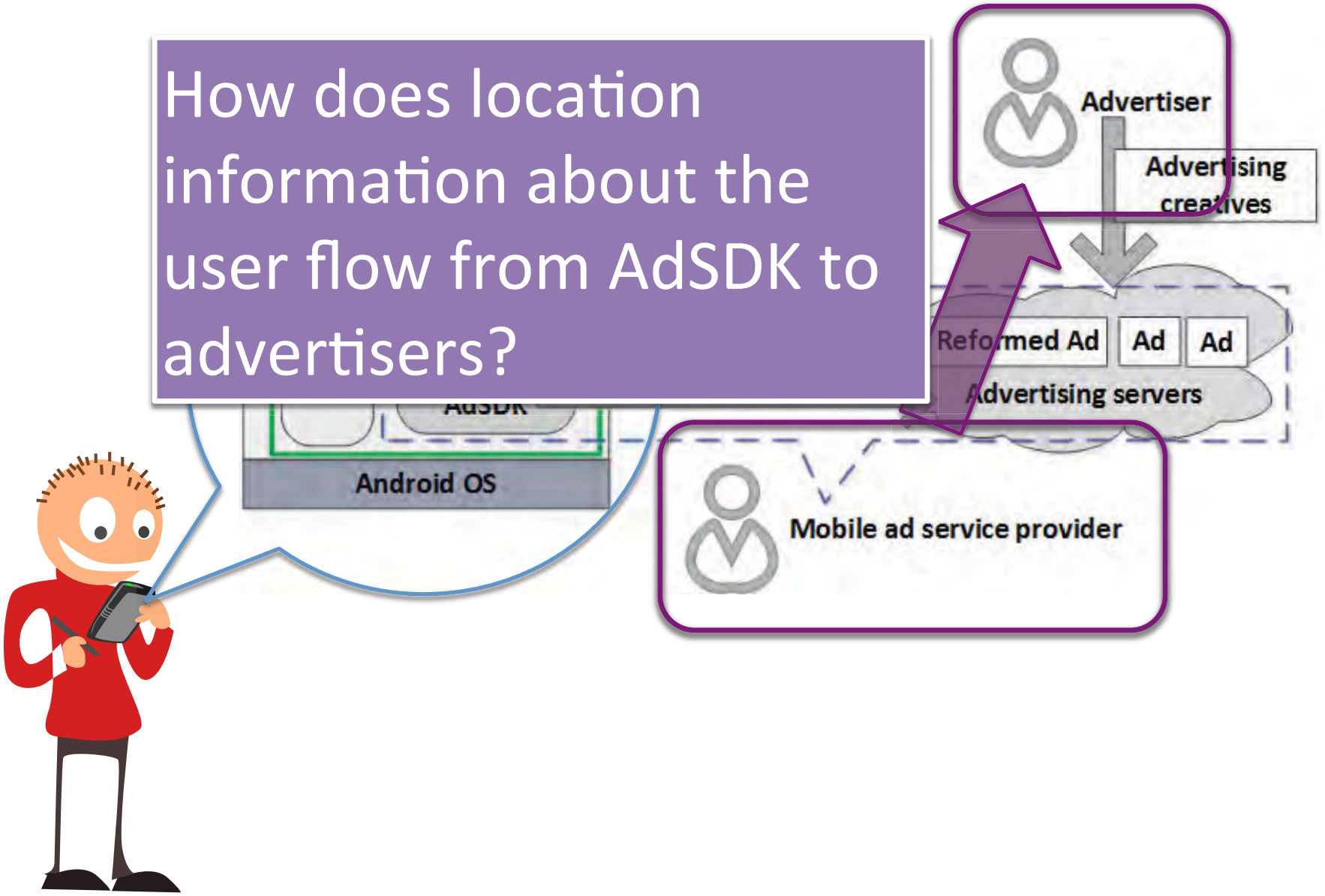
Cookies do not serve their purpose

- Permanent ID: Android ID, Mac address, IMEI, IMSI and others
- Pseudonymous ID: Google Advertising ID (GAID)
- Location data: IP address, coarse- or fine-grained GPS data

Location Data Paired with IDs

- Can infer partial user trajectory
  - **Adverting service providers**
  - **Advertisers?**

How does location information about the user flow from AdSDK to advertisers?



<Ad ID, fine grained location, time1>

<Ad ID, fine grained location, time2>

.....

Location trajectories are strong signals to identify individuals

Summary

- First study of how Android advertising services protect users from malicious advertising
- Standard Web same origin policy is no longer secure in the mobile context
  - Mere existence of a certain file in external storage can reveal sensitive information about the user
  - Direct information leakage
- Malicious advertisers may access trajectories, privacy-sensitive info and infer the identities.

Thank you.

AdSDK	Information sent to AdSDK providers (AdSDK) or advertisers (Ads)					
	Fine Loc	Android ID	H(Android ID)	GAID	Model	H(IMEI)
AdMob [4]				AdSDK	AdSDK	
MoPub [33]	AdSDK, Ads		AdSDK $^{-}$, Ads $^{-}$	AdSDK $^{+}$, Ads $^{+}$	AdSDK	
AirPush [36]	AdSDK		AdSDK, Ads	AdSDK	AdSDK, Ads	AdSDK, Ads
AdMarvel [3]		AdSDK $^{-}$	Ads $^{-}$	AdSDK $^{+}$, Ads $^{+}$	AdSDK, Ads	

$^{+}$ Information sent only if Google Play Services are present on the device.
$^{-}$ Information sent only if Google Play Services are not present on the device.

- Tested 4 popular AdSDKs by following the default guide line with FINE_LOCATION permission.
- Inconsistent information availability between AdSDK providers and advertisers across different vendors.

Flow of User's Location in MoPub