

Mobile application investigation and exposure analysis

Mobile Exposure Graph

Inspect an APK without executing it; connect permissions, code, infrastructure traces, findings, and evidence locations in one relationship graph. Compare two versions and export a hash-bound evidence bundle.

Accurate product boundary

This first release is not EDR. It does not provide a device agent, live telemetry, automatic blocking, or a malware verdict. The file stays in the browser and is never executed.

01 Local intake

02 Bounded static review

03 Evidence binding

04 Graph + diff

05 Verifiable output

Candidate APK

The current or questioned version. This is the primary artifact in the graph.

1

Choose APK

Choose a file or drop it here

APK, up to 100 MB, never uploaded

No file selected

Baseline APK (optional)

An older or known version, used only for version-difference analysis.

2

Choose baseline APK

Choose a file or drop it here

APK, up to 100 MB, never uploaded

No file selected

3

Clear all

Reviewing an APK without running it, with Mobile Exposure Graph

I open an Android app file but I do not run it. I see its permissions, code traces and addresses on one graph, and every finding next to the place in the file it came from.

1 What the tool is for

An APK review usually ends as a long table. I wanted to see in one place which file a permission, an address or a code trace sits in. This tool ties every finding to its evidence. If you give it two versions, it also shows what changed between them.

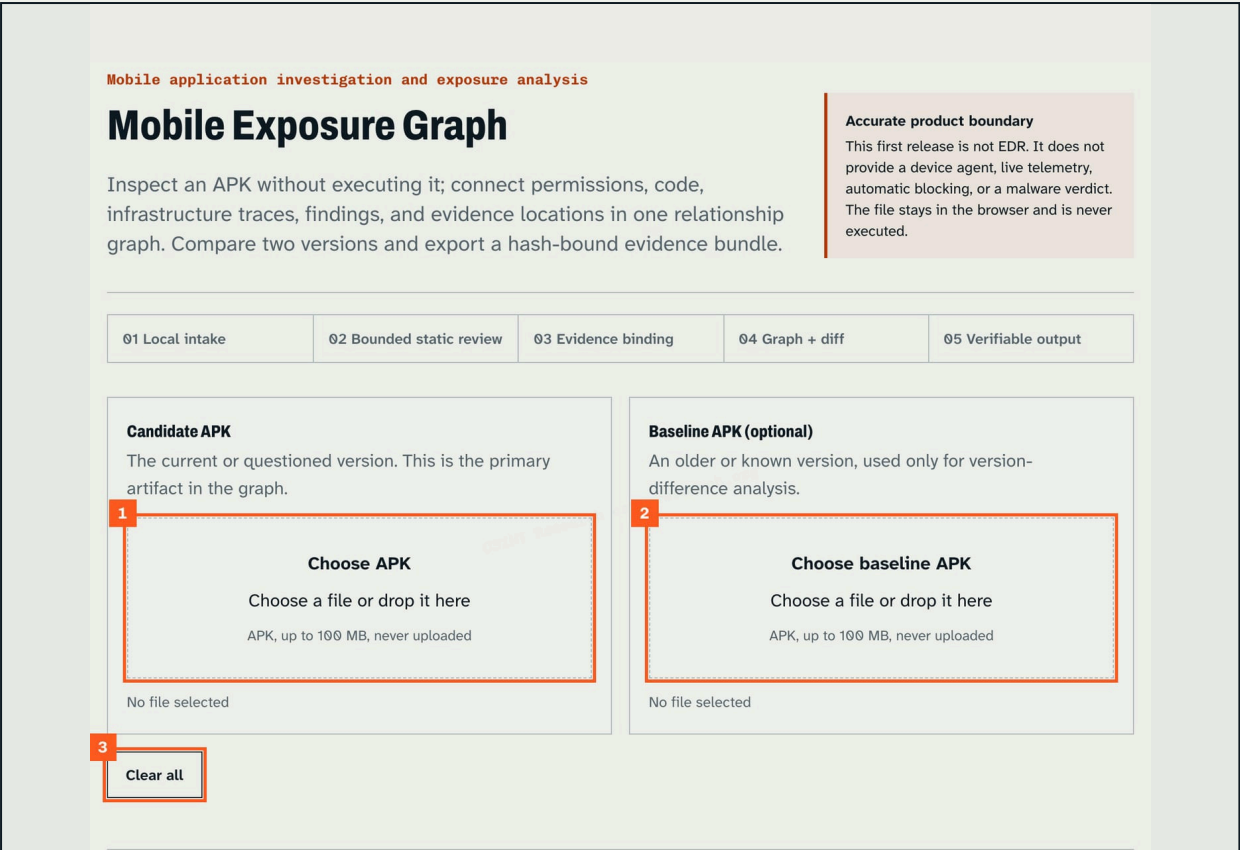
Everything happens in your browser. The APK is not sent to a server and is never run. The tool only reads what is inside the file.

A few words

APK	The install file of an Android app. It is really a ZIP with other files inside.
Manifest	AndroidManifest.xml. It lists the app's name, version, the permissions it asks for and its components.
Permission	Access an app asks for on the phone, such as camera, internet or location.
DEX	The compiled form of the app's Java or Kotlin code.
Native library	A .so file inside the APK. Code compiled straight for the processor.
Cleartext	Network traffic without encryption. The address starts with http, not https.

2 Choosing the file

You need	One .apk file, up to 100 MB.
Optional	An older or known version of the same app, for comparison.
Account	Not needed.
Where data goes	The APK is read in a separate worker process inside your browser. It is not sent to a server.



The file areas. The site ships no sample APK; both areas wait for your own file.

- 1

Candidate APK
The version you question, or the current one. The graph is drawn for this file.
- 2

Baseline APK
Optionally add an older or known version. It is used only to find the difference.
- 3

Clear
Clear all resets both areas and the result. When the file name and Analysis completed appear, the result opens further down the page.

3 Summary and review queue

This and the next sections have no screenshot. I do not publish a sample APK, and I describe the parts of the screen instead of showing the result of a synthetic file.

Number of findings	All findings, with the low and info ones counted underneath.
High and medium	Look at these first. A medium finding cannot be judged without knowing what the app is for.
Coverage	How much of the file the tool could read. If it says <i>Limited</i> , one area was left incomplete and the missing part was not treated as safe.

Pick a finding	Each card in the queue is a finding. The first card is selected at start.
Check the severity	High, medium, low or info.
Below the queue is the evidence for the selected finding.	
Confidence basis	What the finding rests on, and a percentage. Items read from the manifest follow a fixed rule (deterministic). Items from the text scan are an estimate (heuristic).
Location and value	Which file the trace is in and its value. You can open that place with your own tool and check it.
Interpretation limit	What this finding does not prove. A text trace does not mean the connection was ever made.

4 The evidence graph

From left to right, the graph's layers are file, identity, surface, code, infrastructure traces, findings and evidence. The graph is wide, so you scroll sideways to see the right-hand layers. Press a node and its links are highlighted.

Below the graph you see the selected node's name and its direct links. They show which file the node came from and which evidence supports it. The full link table opens from there too.

5 Comparing two versions

Permissions	Permissions added or removed from baseline to candidate. Plus shows added items, minus shows removed ones.
Infrastructure traces	Additions and removals in the domains, IPs and addresses found in the code.
Signature entries	A change in the fingerprint of the signature file under META-INF. The tool does not check whether a signature is valid, it only shows the difference.

A change is not a risk or an approval on its own. A new permission can be the normal result of a new feature.

6 Taking the evidence package

Download the ZIP

It holds the report (JSON and Markdown), the graph, the diff if there is one, and a manifest with the SHA-256 of every file. The original APK is not included.

Check a package

Pick a ZIP you downloaded earlier. File names, sizes and SHA-256 values are checked on your device.

The checker appears in the report section after an APK is reviewed. The Tool verification page publishes the real evidence ZIP of the F-Droid client (org.fdroid.fdroid 1.23.2) with its SHA-256; you can check that file with it.

The package is tied together with SHA-256 but it is not digitally signed. It shows the files have not changed, not who made them.

7 What it does not do

- × It does not run the app. It does not prove a permission is actually used.
- × It gives no malware verdict. No findings does not mean the file is safe.
- × It installs no agent on a phone and does no live monitoring or blocking.
- × For a deeper review, use the File and document sandbox link at the bottom of the page. Its isolated analysis runs only with separate consent; this page never sends a file there by itself.
- × The text scan has limits. In very large files part of it may not be read; then coverage says Limited.