Playing with your code: a new approach to avoid potential hackers from doing exactly this!

Hugues Thiebeauld

Android Security Symposium
Vienna – 8th of March

Key question

Mobile application goes to a hacker

How do I know if this will be tough for him

Mobile application tools

focus on finding vulnerabilities, virus, malware

NowSecure™

FORTIFY® An HP Company

pradeo SECURITY SYSTEMS

WhiteHat SECURITY®

we want to know if the armour is strong

Mobile application tools

focus on finding vulnerabilities, virus, malware

we want to know if the armour is strong

NowSecure™

FORTIFY An HP Company

pradeo SECURITY SYSTEMS

WhiteHat SECURITY

Solving a critical problem

Security protections are per nature “transparent”

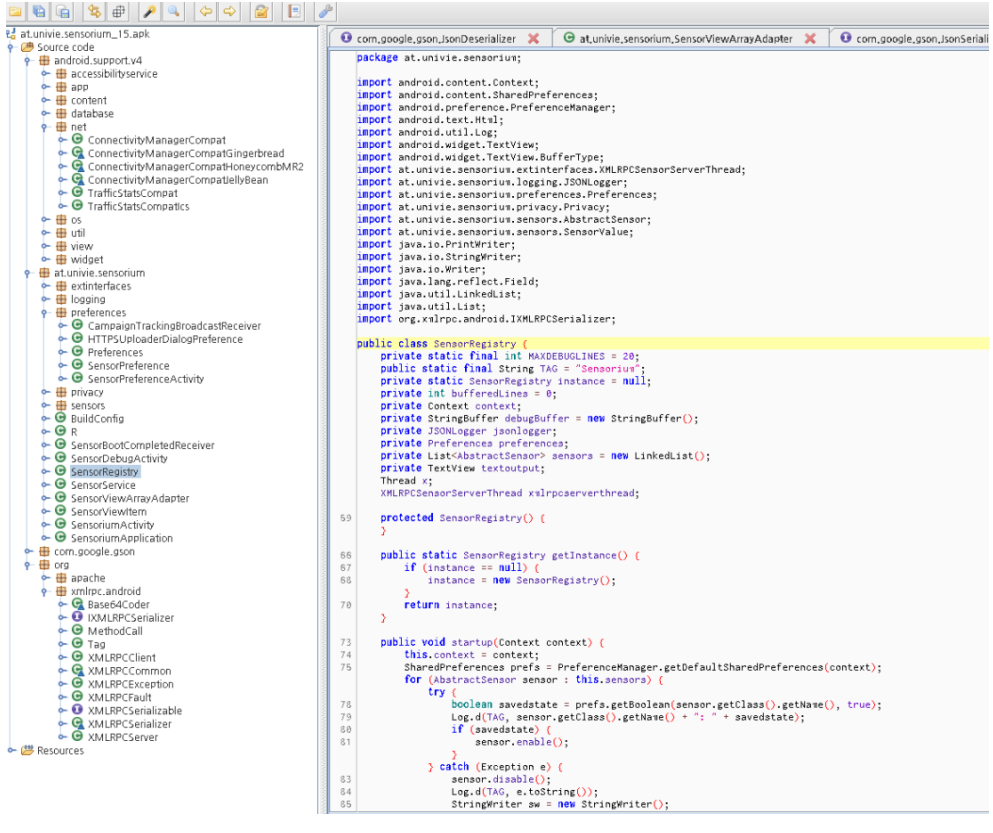
How confident the protections have been effectively enabled?

Protected mobile app

No other choice than hiring a security analyst to manually inspect

Are protections enabled at the right level?

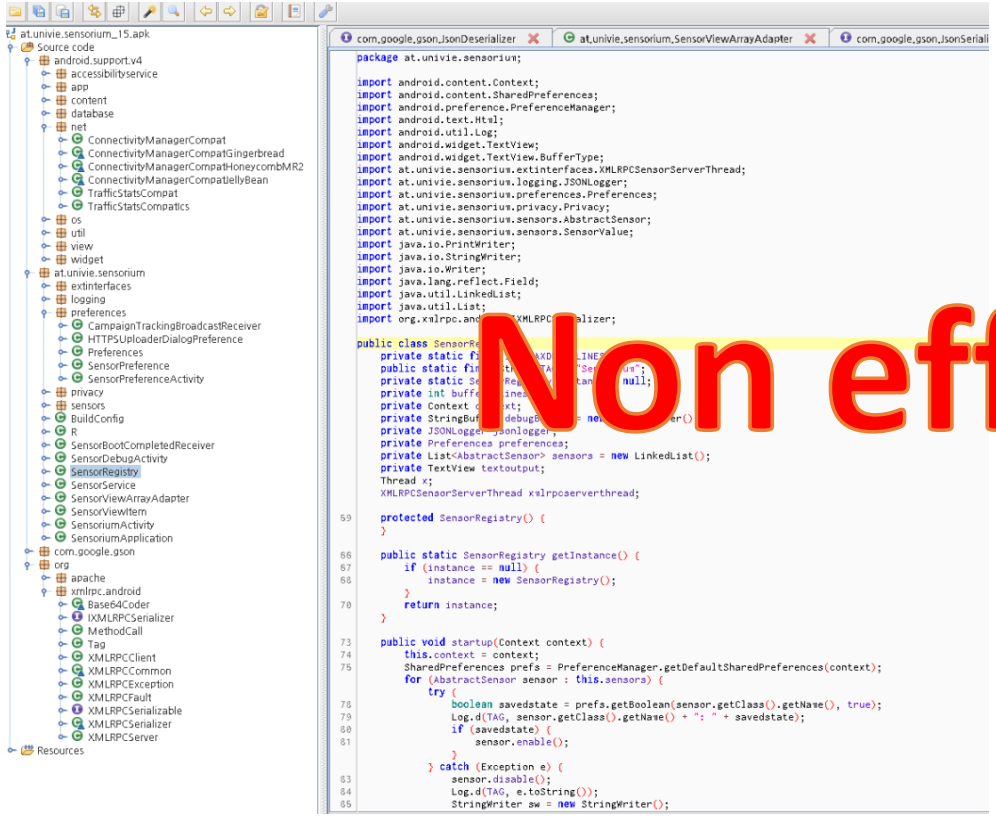
Manual inspection



How to know what to look for first?

{Ctrl+F, Strings + Grep, Androguard, Smalisca, etc.} = TEXTUAL, VERBOSE, SLOW, TEDIOUS, BORING

Manual inspection



How to know what to look for first?

Ctrl+F Strings + Grep, Androguard, Smalisca, etc.} = TEXTUAL, VERBOSE, SLOW, TEDIOUS, BORING

Non effective !!

How many of you would like a tool…

Comprehensive

Visual

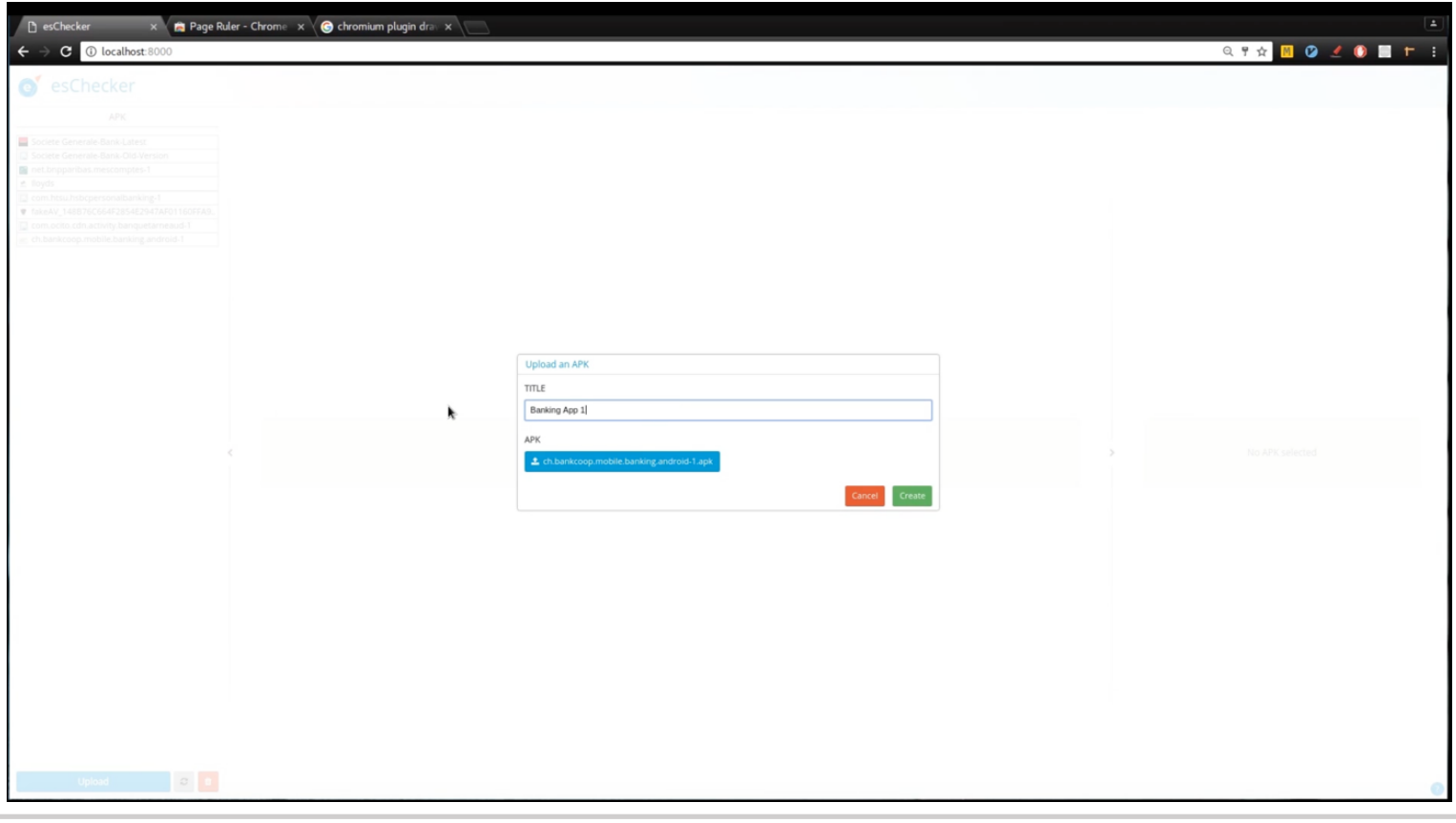
Customisable

Easy to use

No black & white

Demo esChecker

- *Download your APK files*
- *The binary is disassembled and decompiled. Each piece of code is analysed against a set of heuristics*

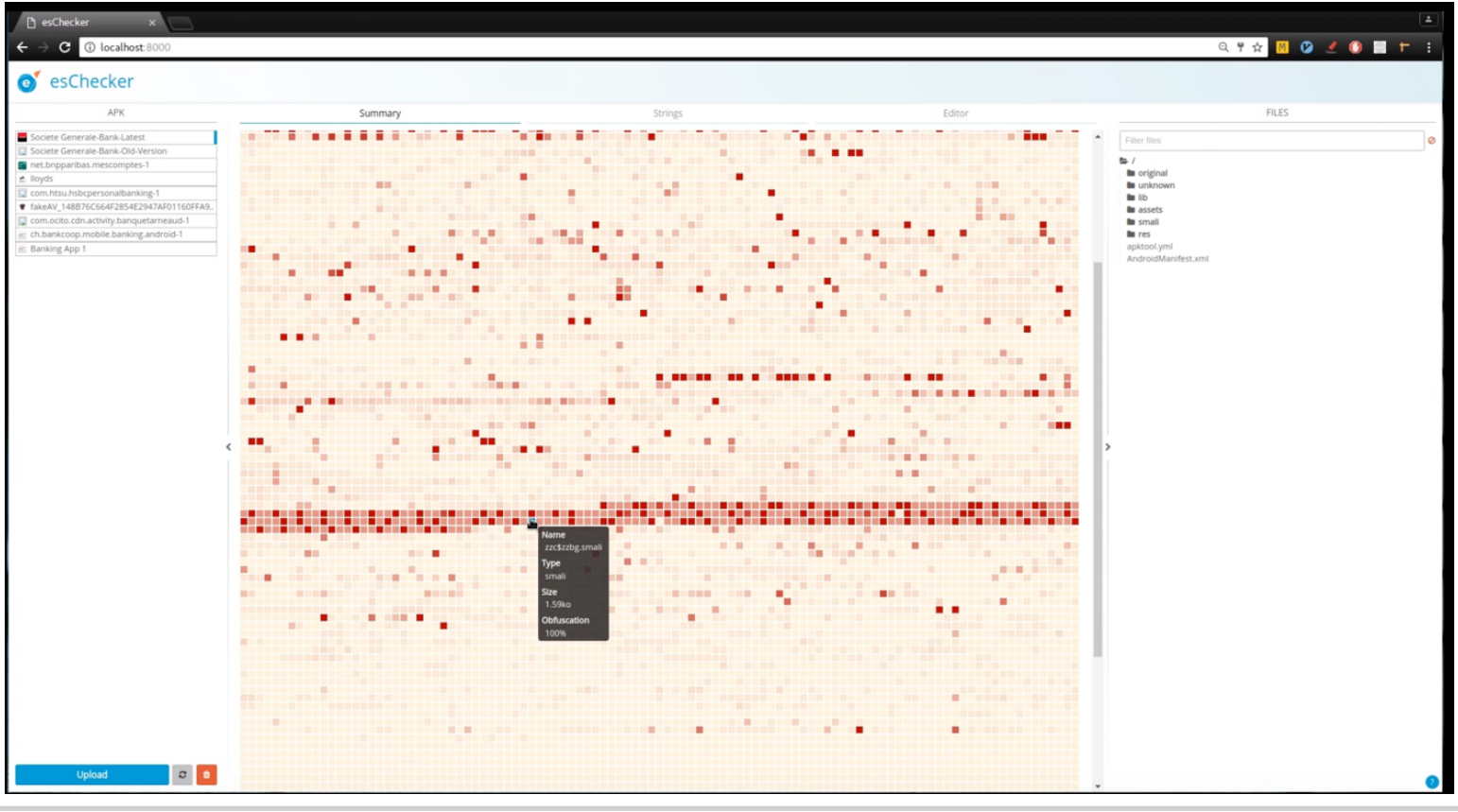


Demo esChecker

- *As outcome of the analysis, a tile view is given.*
- *User has the opportunity to choose the APK*
- *User can browse into the file mapping view*
- *User can browse within the file tree*

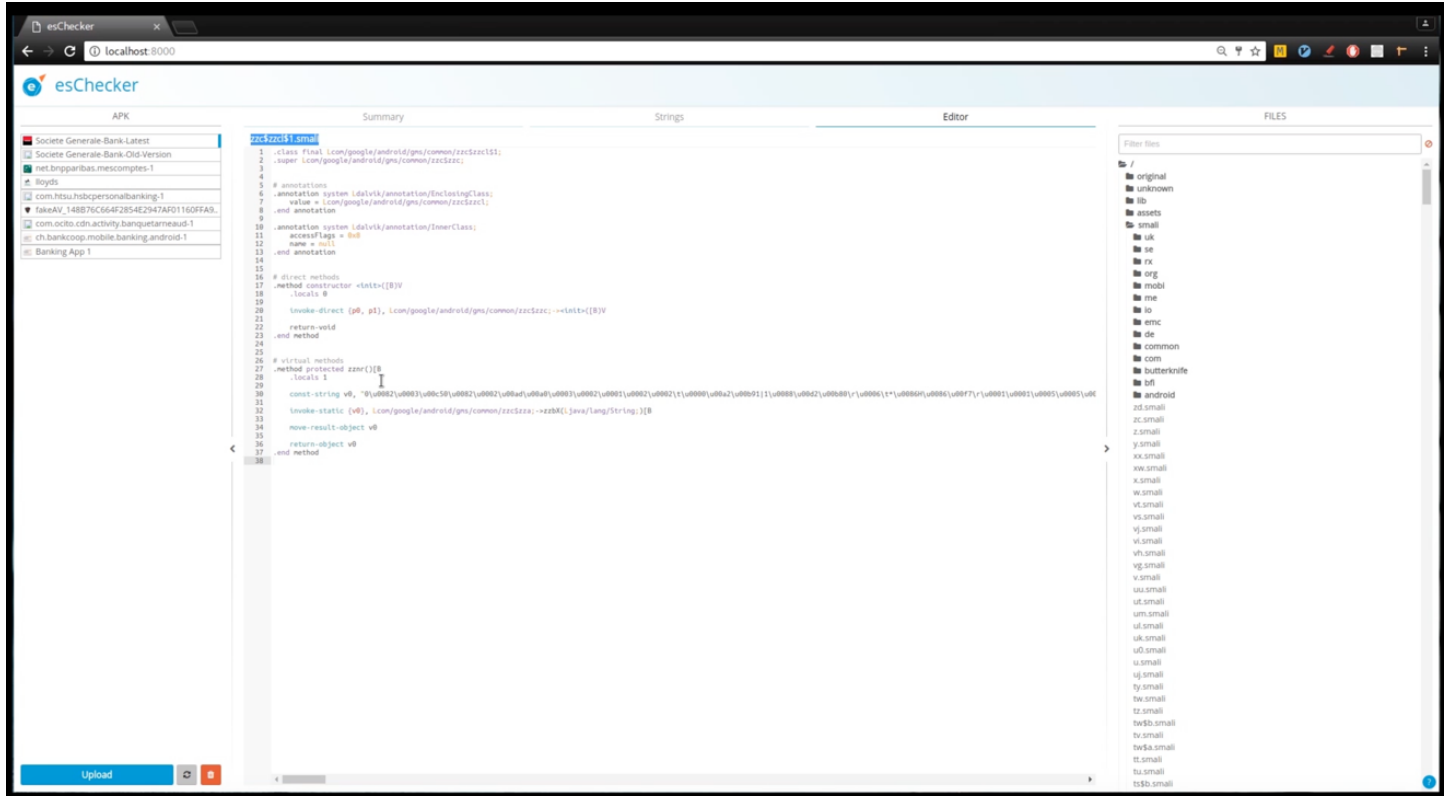
Demo esChecker

- *Each tile represents a piece of code*
- *An infobox displays information related to the file and shows how much obfuscation was found into it*
- *Different levels of colour helps to quickly see how much obfuscation was found*



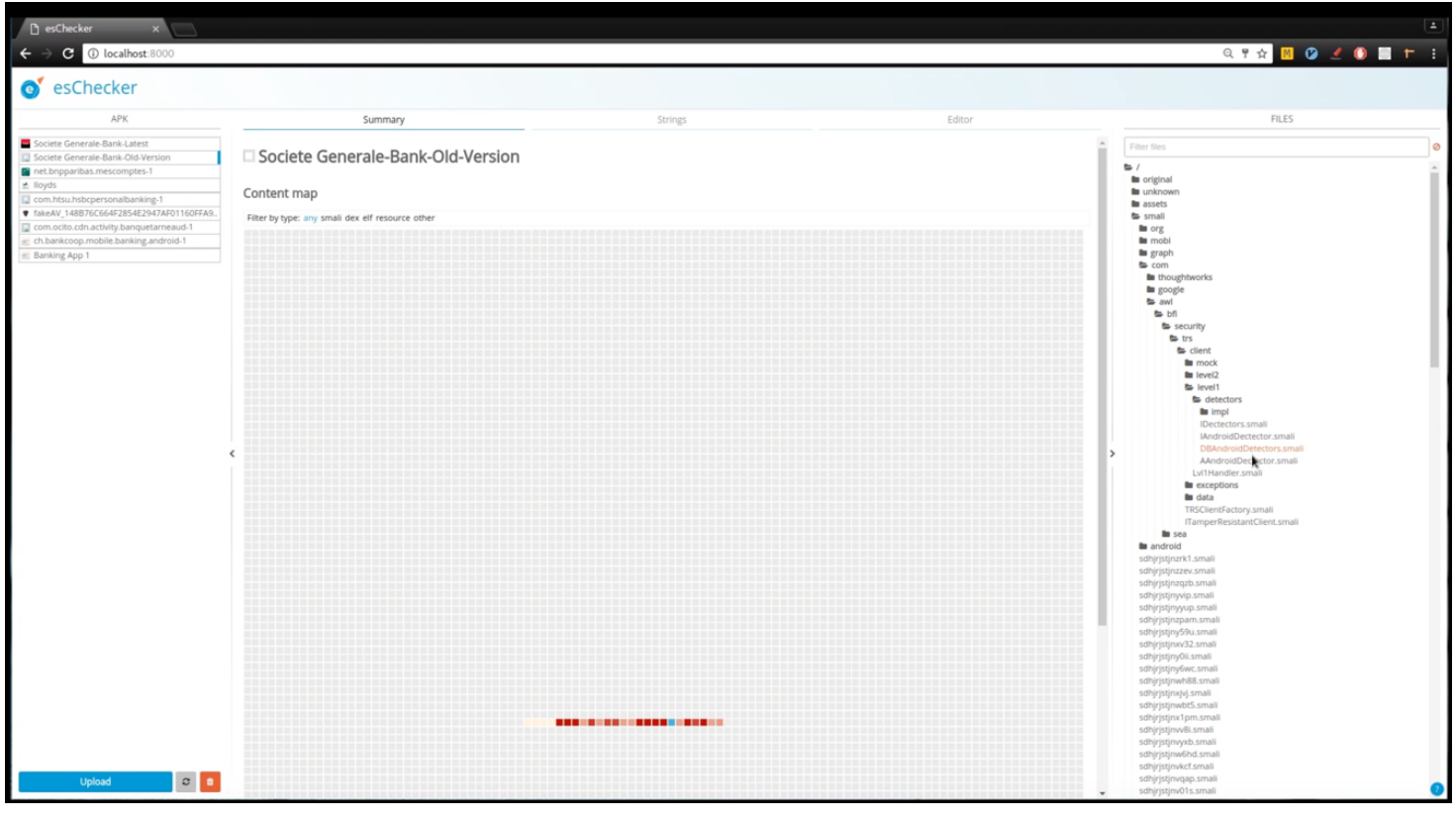
Demo esChecker

- *User can jump from the graphical view to the editor view or the other way around*
- *Corresponding obfuscation can be spotted… here, non ASCII characters and renaming*



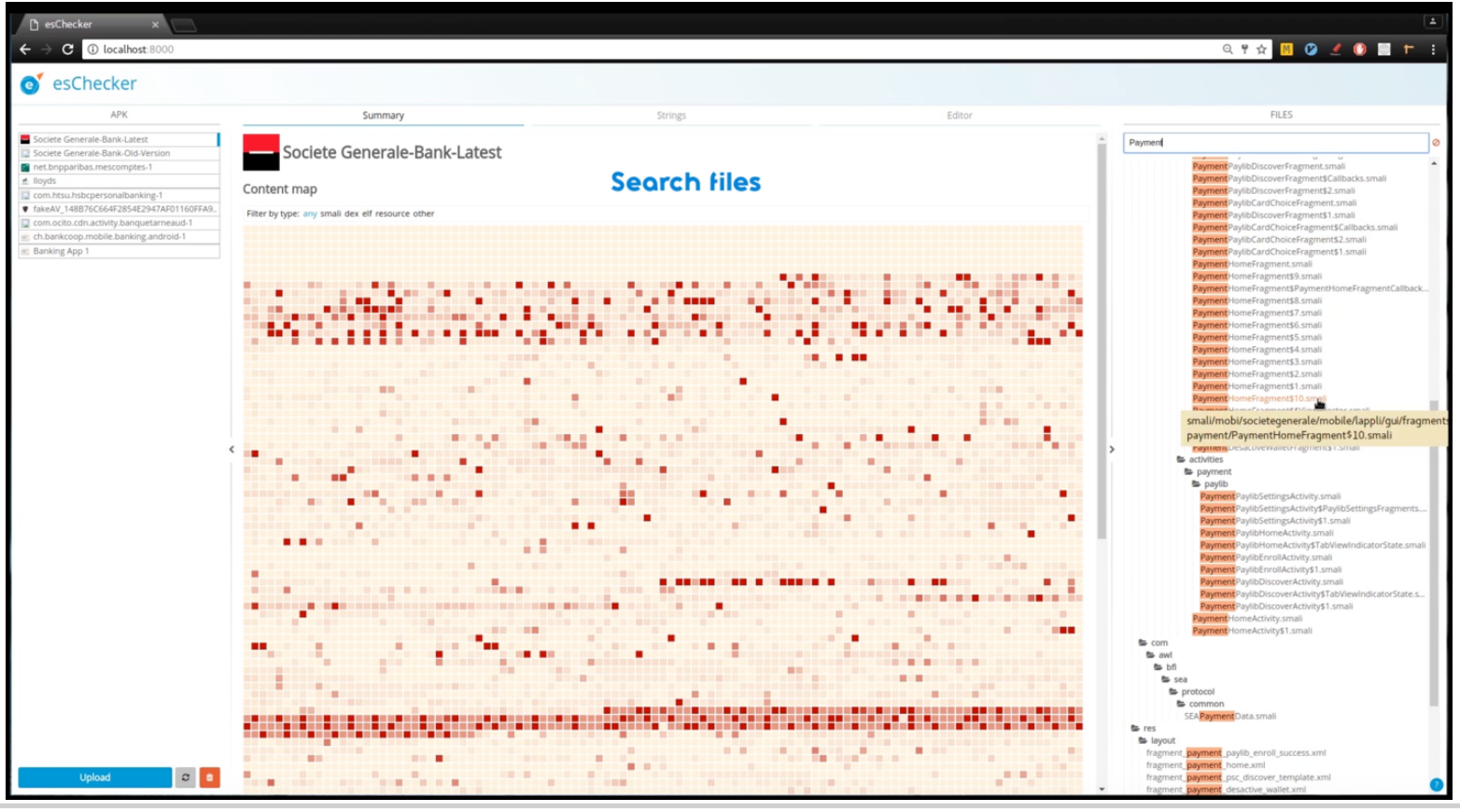
Demo esChecker

- *The view can be restricted to a given package by simply selecting the folder on the right.*
- *User can just move up and down within the folder structures.*



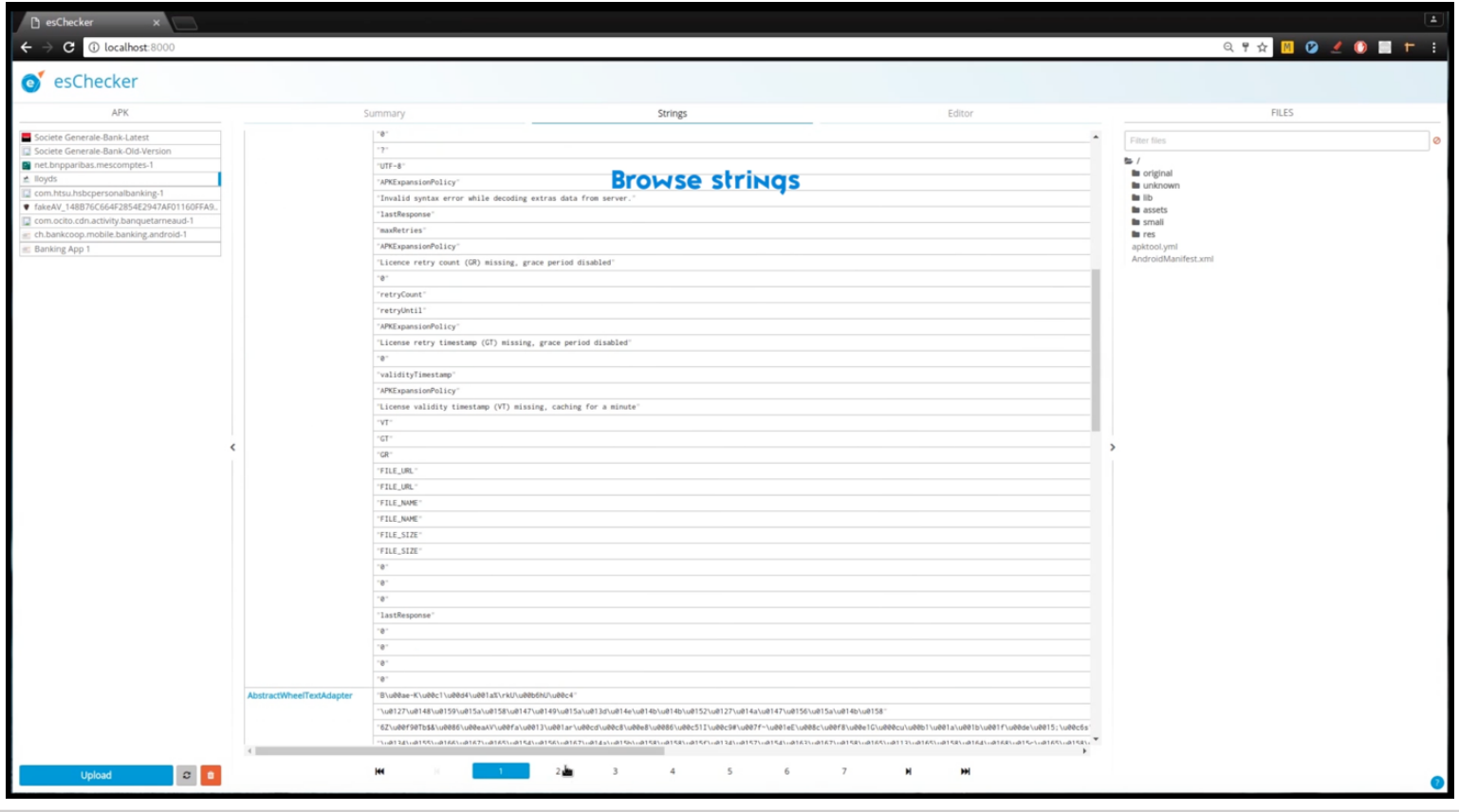
Demo esChecker

- *A search engine is available to collect all files with key words*
- *Helps to spot easily specific functions and locate them into the tile map.*



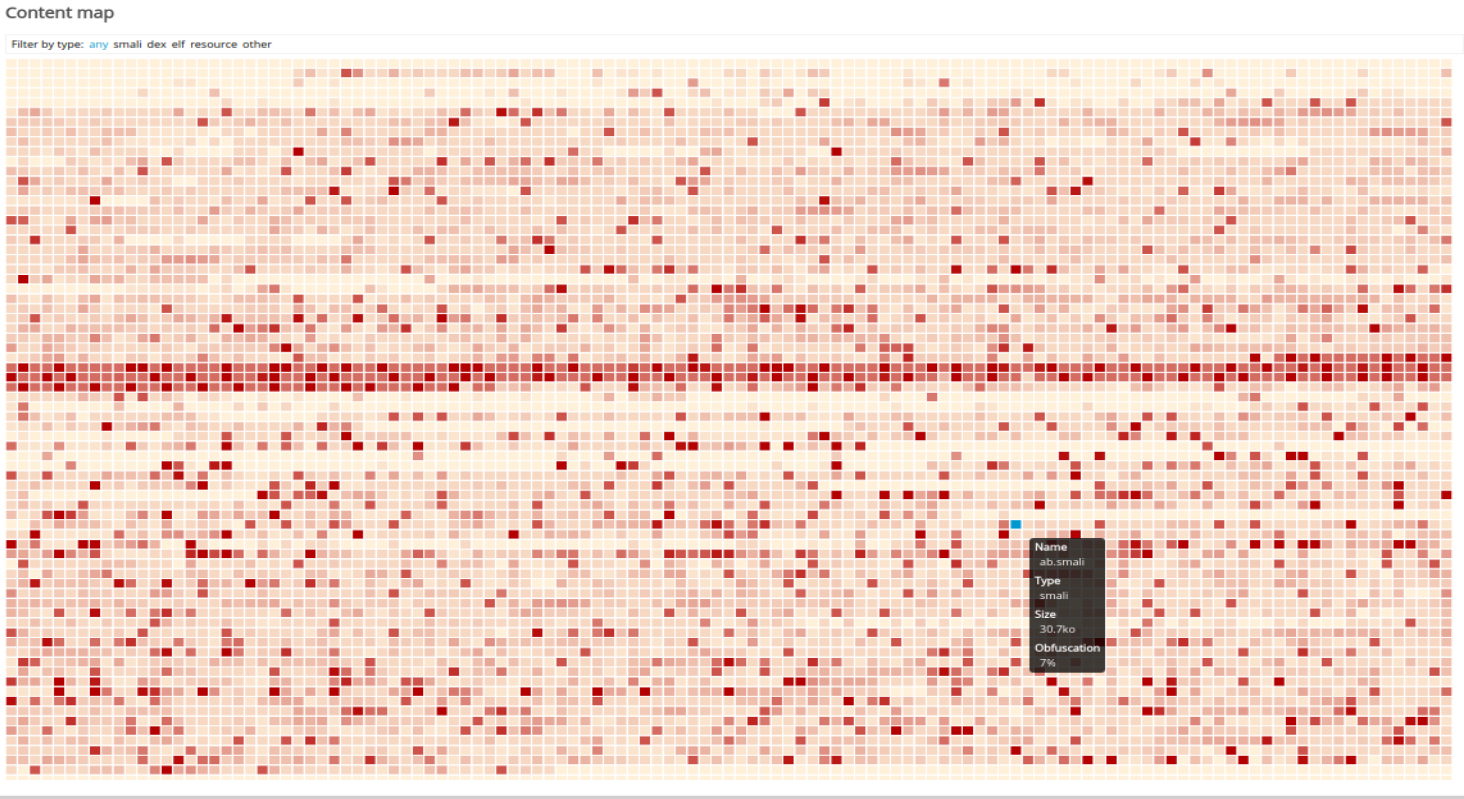
Demo esChecker

- *Strings are very useful to find his way into a binary*
- *A specific section of the tool is dedicated to strings*



A first example

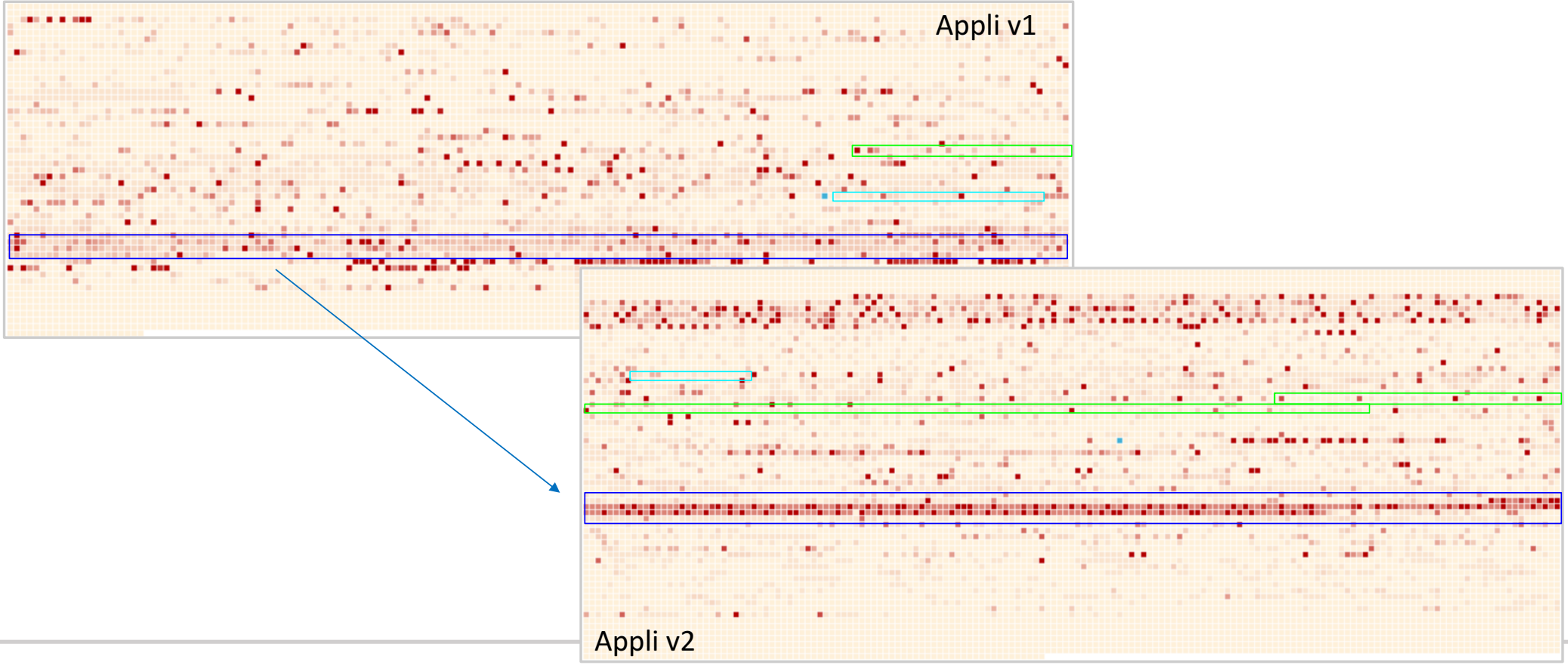
- *Weak obfuscation*
- *Strings are not obfuscated and remain way too much informative*
- *Control flow remains very easy to understand despite classes/methods/fields renaming*
- *GMS (in middle) looks more obfuscated than the rest of the app*



A second one…

- *Medium obfuscation*
- *Classes/Methods/Fields are renamed*
- *Usage of non-ascii characters for the renaming*
- *Some strings are obfuscated/encoded*
- *Control flows have been slightly obfuscated*

Delta version



Conclusion

provides an easy access to mobile app protections

Security Analysts

Save your time

Go straight to the point

Reach comprehensiveness

Everyone in the area

No need to be an expert to get a first insight

Manage the versioning and avoid regression

Use it as a benchmark

Reproducible process

Want to know more?

www.eshard.com

@eshardNews

eshard