

Solving the hard problems



Travis Himes
Android Developer WBDL

Explanation

What do I mean “hard” problems?

Nobody wants them!

How I suggest approaching addressing them

Case Study

InputStream

XOR obfuscation

Bitmaps

A Primer on XOR

The Operation

$$1 \wedge 1 = 0 \quad - \quad 0 \wedge 0 = 0 \quad - \quad 1 \wedge 0 = 1 \quad - \quad 0 \wedge 1 = 1$$

Commutative

Symmetrical

Cyclic Key

password -> "passwordpasswordpasswordpasswordpassword"

APRIL 08+09

DROIDCON
BOS19



Symmetrical “Encryption”

quick brown fox



abcbabcbabcbabc



rwldmcctqxocgqa

rwldmcctqxocgqa



abcbabcbabcbabc



quick brown fox

APRIL 08+09

DROIDCON
BOS19



Position matters

rwldmcctqxocgqa



abcabcabcabcabc



quick brown fox

rwldmcctqxocgqa



cabcabcabcabcab



meaning is lost

APRIL 08+09

DROIDCON
BOS19



A Hard Problem arises...

Small Bitmaps decode fine - everything's good!

Medium and Large Bitmaps sometimes decode ok, otherwise not

Why not?

How do you fix this?

Find the code that's causing the problem

Fix the code that's causing the problem

Finding the source of the problem

Find the shortest path to repeatability

Flip (or remove) the ego:

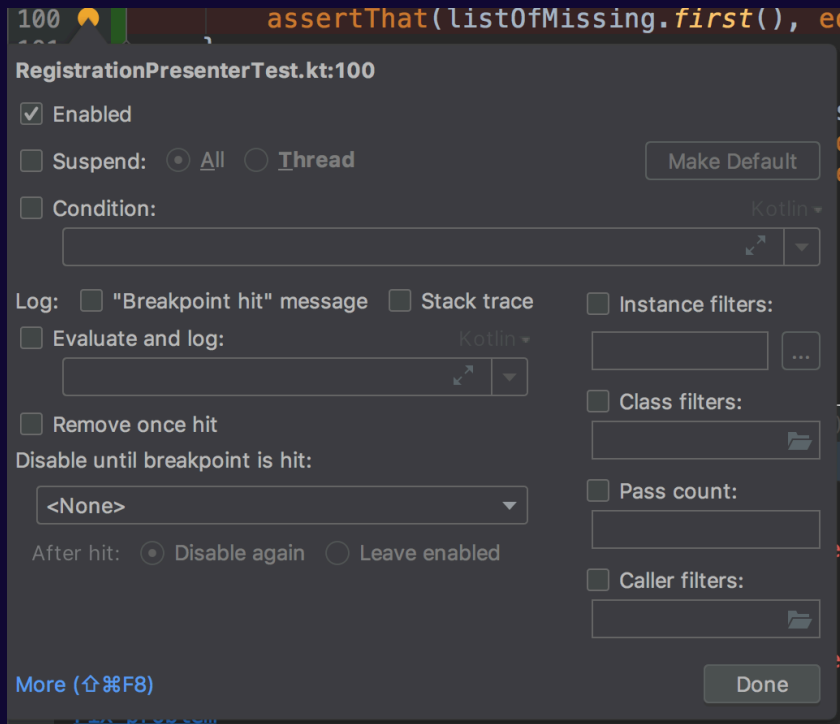
What's the simplest/silliest possible cause?

LOG EVERYTHING

Everything!

Eventually, this reveals the cause

Strategies



APRIL 08+09

DROIDCON
BOS19



Finishing the case study

BitmapFactory

quick brown fox

InputStream

↓
abcabcabcabcabc

Bitmap

↓
rwldmcctqxocgqa

Understand the cause

Learn from your mistakes

Keep a list of past offenders

- Implementation Overrides

- PendingIntents

- Date/Time/Timezone related

- val get() vs fun

Understand the cause

Thank you!

Questions?