

CASE STUDY

A popular news app, that's keen about optimizing retention and performance, used SafeDK solution to learn about an SDK that crashed it over 20K times a day, and instantly shut it off!

Background

After suffering ongoing app performance faults traced back to 3rd party SDKs, a major news app implemented SafeDK's solution. Barely 24 hours have passed since it released a new version to the store, and the app team received alerts by the SafeDK system about an ad-network SDK that has crashed the app **over 20,000** times! The team had validated the alerts, found them to be accurate, and instantly turned off the "naughty SDK" (by activating the 'kill switch' in the SafeDK dashboard).

The effect was immediate and boosted the app's retention. SafeDK's solution found the problem quickly, prevented the need to investigate about the cause or to release a version update in order to solve the issue.

The fast identification of performance faults, in this case an ad-network SDK that causes crashes, and the immediate resolution, saved money in ads revenue and boosted user satisfaction.

This success story is about a top tier news app that serves millions of users daily.

It offers breaking news, world-class reporting and in-depth commentary.

The company also offers multiple news magazines, both online and in print.

"SafeDK's solution changed the game for us. We now feel confident in testing various SDKs and are able to manage issues like the one mentioned here with ease. We are not worried about SDKs issues anymore. It's easily manageable."

Nitzan, Mobile Project Manager

The Challenge

Being such a popular news outlet, the app has implemented multiple SDKs that enable it with all kinds of capabilities, such as analytics, communication, ad-based monetization and more. However, that also entails an SDK management overhead.

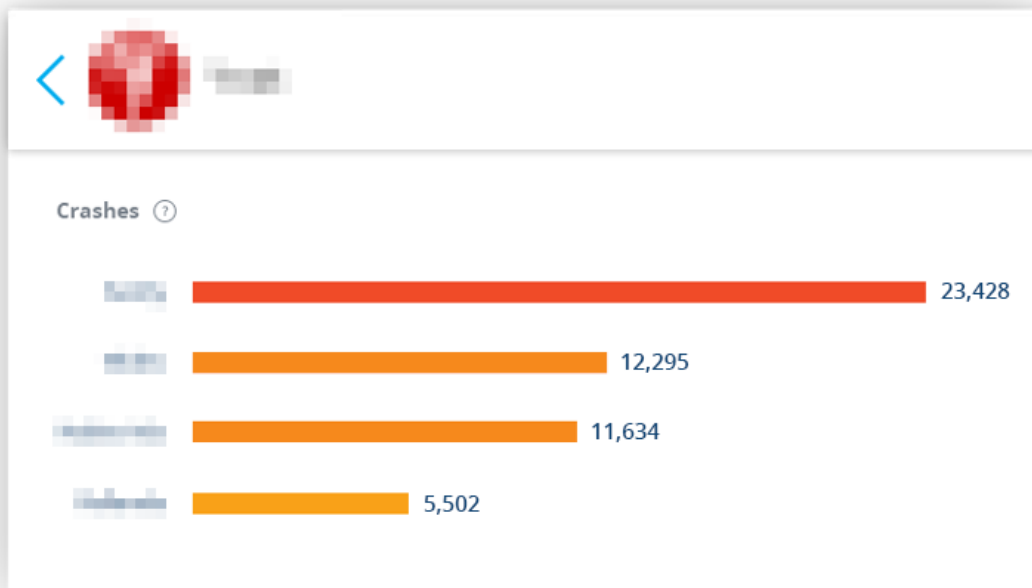
Implementing a substantial amount of 3rd party SDKs is a routine in the lives of popular apps that have reached a certain level of maturity.

Naturally, some of these SDKs cause (knowingly or not) performance issues, privacy breaches or violations. The harmful results reflect on user retention, churn and revenues.

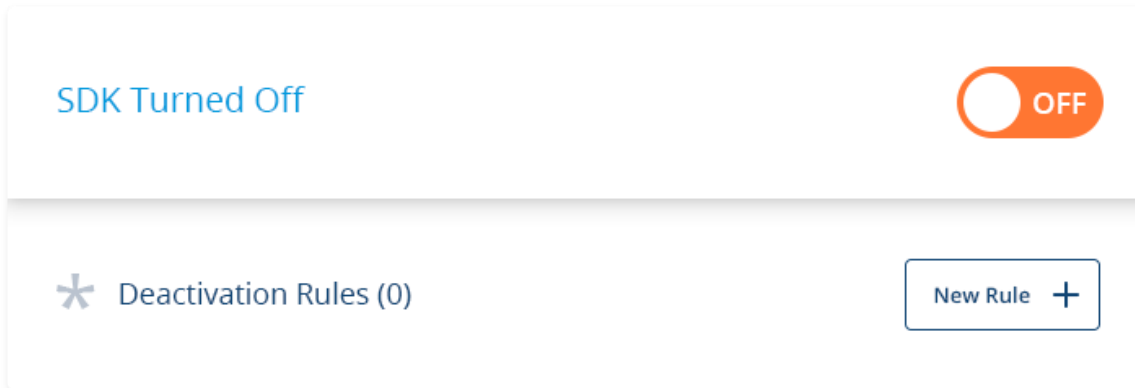
After suffering multiple performance issues, and receiving negative reviews in the App Store and Google Play, the app developers took every action to ensure they properly address SDKs-related issues. They have decided to implement SafeDK.

The Solution

One day into the release of an app update (this time SafeDK inside), it was clear that there's a stability issue. A quick glance at the SDKs crashes information in the SafeDK dashboard, or at the emails alerting about crashes sent by SafeDK, were enough to find the source. One of the newly integrated ad-network SDKs crashed the app **more than 20,000 times a day!**



Faced with the possibility of losing countless users and revenue due to these crashes, the app developers immediately switched off the SDK remotely, using SafeDK's 'kill switch' for SDKs. The recovery was instant, and there was no need to release another version update.



The Result

Being able to instantly turn off the crashing SDK had saved the app money in ads revenue and stopped the stream of negative reviews and uninstalls.

The story doesn't end here. The following day, SafeDK was approached by the ad-network SDK team, asking for more details, as they weren't even aware of the crashes they caused, and they wanted to take actions as soon as possible. SafeDK team had run an analysis and sent the SDK team logs with explanations on the causes of each crash. The SDK then subsequently attended to these issues and resolved them.

Here's what the daily news app mobile project manager had to say:

"SafeDK's solution changed the game for us. We now feel confident in testing various SDKs and are able to manage issues like the one mentioned here with ease. We are not worried about SDKs issues anymore. It's easily manageable."