# **Security for Flutter Applications by Adam Josevski**

**Code Obfuscation**

Code obfuscation happens when you alter the app’s binary using encryption. Code Obfuscation hides sensitive information shown in the code and means the attacker cannot reverse engineer the mobile application. If the attacker is able to reverse engineer redback operations mobile apps, the attacker can see sensitive information such as API keys, classes, function names, and all strings values [1]. Code Obfuscation can be used on Android, iOS, and MacOS and to secure flutter mobile apps using obfuscation. Implementing code obfuscation can be found within Flutter documentation [2].

**Rooting or Jailbreaking protection**

Rooting android devices and jailbreaking or iOS devices is a serious security issue mobile developers must be aware of and must detect this issue to act accordingly. Rooting and jailbreaking devices will allow the attacker to implement malware in the application to export client’s critical data [1].

An extensive guide on how to implement rooting and jailbreaking protection in mobile development applications can be followed here: <https://pub.dev/packages/flutter_jailbreak_detection>. This guide also states that there is a plugin called RootBeer that detects vulnerable mobile applications that can be rooted by attacker. In addition, another plugin called DTTJailbreakDetection is used to check whether iOS devices can be jail broken.

**How to Secure flutter application code:**

We also want to restrict network traffic to make sure redback operations domain is not communicating with any other domains. From the GitHub links provided by redback operation I can see the developers have not implemented ways to restrict network traffic for iOS and android. Therefore, I recommend following <https://www.codeplayon.com/2021/12/how-to-secure-flutter-application-code/> . This will make the flutter applications secure and restrict malicious activity on the network. I will also highly recommend using SSL or TLS encryption, this will protect data transferred between the mobile application and the relevant server [1].

**Note**

Please note, that I will create a presentation to elaborate further, and I will recommend further changes. This document is to show the mobile developers of Redback Operations on how to implement critical security measures in flutter applications.

**References**

[1] AB. Satyaprakash (2022, February, 2). *5 steps to secure your next Flutter* app [Website]. Available: <https://medium.com/nerd-for-tech/5-steps-to-secure-your-next-flutter-app-549def2428b3>

[2] Flutter (n.d). *Obfuscating Dart code* [Website]. Available: <https://docs.flutter.dev/deployment/obfuscate>

[3] appmire.be (n.d). *flutter\_jailbreak\_detection 1.8.0* [Website]. Available: flutter\_jailbreak\_detection: <https://pub.dev/packages/flutter_jailbreak_detection>

[4] Codeplayon (2021, December. 2). *How to secure flutter application code* [Website]. Available: <https://www.codeplayon.com/2021/12/how-to-secure-flutter-application-code/>