For this challenge was given no description/text inside the usual Google Form, so nothing to report here. We’re only given the .apk file, for us ready to inspect.

Solution

Let’s start from the usual:

*jadx -d out goingnative.apk*

From the path “out/sources/com/mobiotsec/goingnative/MainActivity.java”, there are some interesting things to notice:

* there is a native function *checkFlag*
* there is a library which is being loaded, which is “goingnative”
* the function splitFlag, which checks if the flag in the format FLAG{XXXXXXXXX} and if it is 15-character long

The function splitFlag uses code from the dynamic library and we can use some tool like Ghidra/IDA (here, the first one will be used) to analyze the library. This can be found in the resources folder, precisely inside “out/resources/lib/x\_86\_64/libgoingnative.so”.

In the list of functions, we can notice “Java\_com\_mobiotsec\_goingnative\_MainActivity\_checkFlag” function, which contains a validation over the flag being input, giving “Correct flag” or “Invalid flag” as output. There is a loop evaluation going on to check if the input

The key part is found inside the *validate\_input* function, specifically in the check for three parts of the string: “status”, “1234” and “quo”. This undergoes the observation of the *strtok* function, which puts a delimiter between all chars. We still don’t know what such delimiter can be; one chance can be putting the underscore which is the usual delimiter as char or either trying the app itself executing via the MainActivity.

Inputting it, this confirms the flag is: *FLAG{status\_1234\_quo}*