TaintDroid disallows application developers from including their own native-code libraries in Android applications. Suppose instead that TaintDroid allowed apps to use native-code libraries, but was otherwise unchanged.

Describe how an app designer could trivially leak sensitive data to the network without TaintDroid’s detecting the leak in this scenario. Justify your answer with relevant details of how TaintDroid propagates taint in native code. (Assume TaintDroid uses the same techniques for an app’s native-code libraries as it does for system-provided native-code libraries.) Why did TaintDroid’s designers take such a false-negative-prone approach?