Report:

Analyzing IMO packets:

1. For the last few days, we are working to analyze the packets of IMO captured by Wireshark. Firstly, we have taken some capture and working to analyze the packets that are captured while making a call via IMO.
2. We are comparing two captures of this type.
3. Firstly, we want to figure out the signaling packets.
4. In the first capture, we enter IMO at about 13 second (according to the Wireshark capture time) and at about 14.403 second, the receiver’s IP in found.
5. So, we are assuming that within these packets (no. 55-78), we have the information about the receiver.
6. According to the same way, for another call we may come to a decision that within packet no 140-160, the information is present.
7. We are assuming that the receiver’s phone no will be present in these data, may be in an encrypted form.
8. So, we are trying to find out if we can get the number.
9. Within this serial no, the packets with IMO’s ip is mainly the QUIC type. So, assuming that the QUIC packets are our suspected packets and analyzing those.
10. We are writing some programs to check if we can get the phone number from the data(payload) of the packets.
11. Besides, comparing the two captures, we have got some similarities.
12. Firstly, there is a packet(QUIC) of length 116 both in the capture.
13. Besides, we have also got two these type of similar packets of length 71 and 159.
14. As, there are some similarities, these packets may contain some information we think.
15. Besides, there are some malformed QUIC packets in both the capture.

Apk Analysis Update:

1. The first problem we face is that most of the classes are named ‘a’, ‘b’, ‘c’. So this is tough where to start.
2. After looking some random packages, found some Meaningful Class Name. So, our first target is to understand those classes. So, here are some details about the findings.
   1. We found 3 jar files from **viber.apk**. each of them are linked. We found a package called **‘http’** from all of the jar. The classes from these packages use some Java Build in **Encryption** class and methods. Like **X509TrustManager, X509 Certificate**. These classes are used mainly for certificate verification.
   2. Another class PubKeyManager contains some random string. This class is used in another class called **OkURLConnectionHttpRequest** but didn’t found how the random strings are used.
   3. Most the classes import some build in javax.net class. Like SSLContext, SSLSession
   4. In the 2nd jar file of Viber, all supported CipherSuite name and TLS, SSL version are written.
3. After randomly checked the jar, next we decompile imo and there are only 1 jar file. But the main problem is most of the classes are in a,b,c format.
   1. But in **com.imo.android.imoim.p,** found something about Base64 encoding and decoding in class **g** and **k**.
   2. On the other hand, we suspect some imo QUIC packet in wireshark as signaling packet. Thought that Base64 algorithm can be used to decode the encrypted signaling data. But the problem is we can clearly see that the captured data is not encrypted using Base64 algorithm. Data isn’t directly encrypted using Base64.
   3. No other Encryption related java function is found on IMO till now.
4. Main fact is we can’t run the jar file to see its output.
5. So now our target is to go through every class to see any important function can be found.