StudentID: 6A5723ED

Challenge: Two Factor Android

Date: 4/7/2025

Challenge Solution:

1. The challenge provides two links. One to an Android .apk file, and another to a 2FA portal that looks like it will be more useful after we understand the app.
2. Dropped the twoFactorAndroid.apk file into Kali
3. Ran "sudo apt install jadx"; JADx is a tool for pulling apart and analyzing .apk files, which is exactly what is needed here.
4. "./bin/jadx-gui" opens up a nice GUI for us to analyze the apk with; much easier to understand than a command line in a case like this.
5. Opened *twoFactorAndriod.apk* using JADx GUI.
6. It seems like the majority of the interesting Java code behind this app is in *Source Code/com/example.twofactorandroid/MainActivity* directory. This is where we find the below:
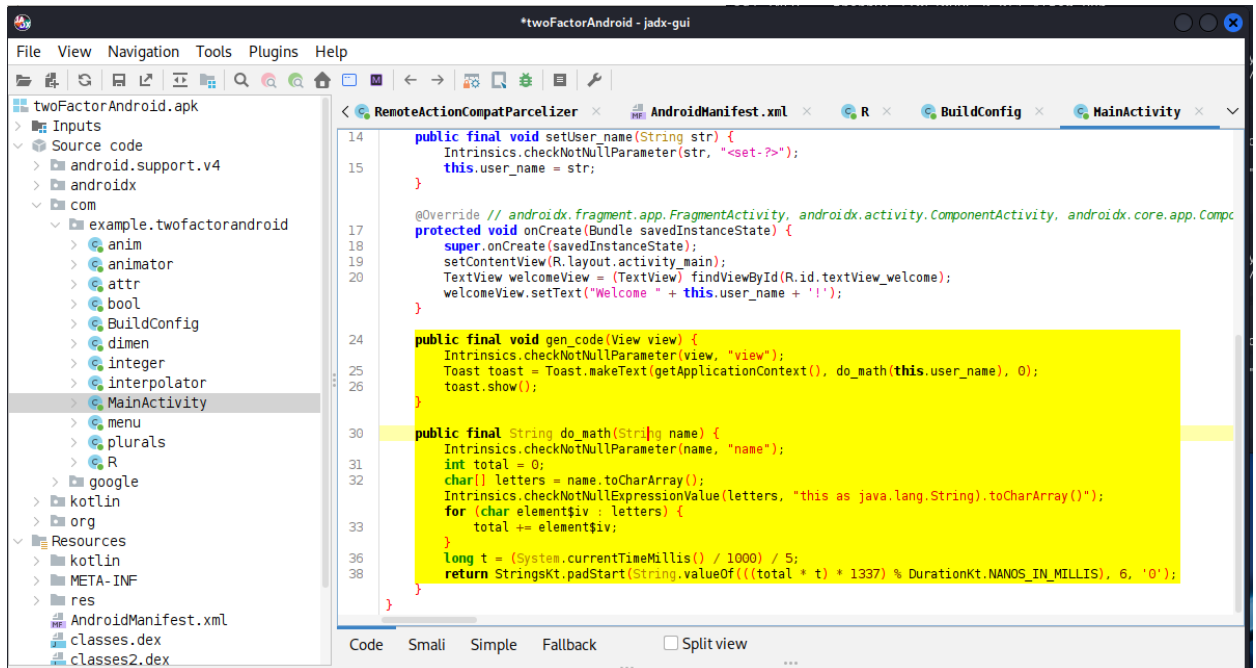

7. So we have a hardcoded variable, *user_name*, set equal to *john*
8. Further down, we have some logic that appears to be generating the 2FA keys, but it looks like it relies on the username to do some math, which creates a chance for us

to pivot from this version, which gives us *John*'s 2FA codes to the *Admin* account.

9. If we grab this logic, translate it to python, and stick it into a lightweight python IDE such as thonny, we end up with a script like this.
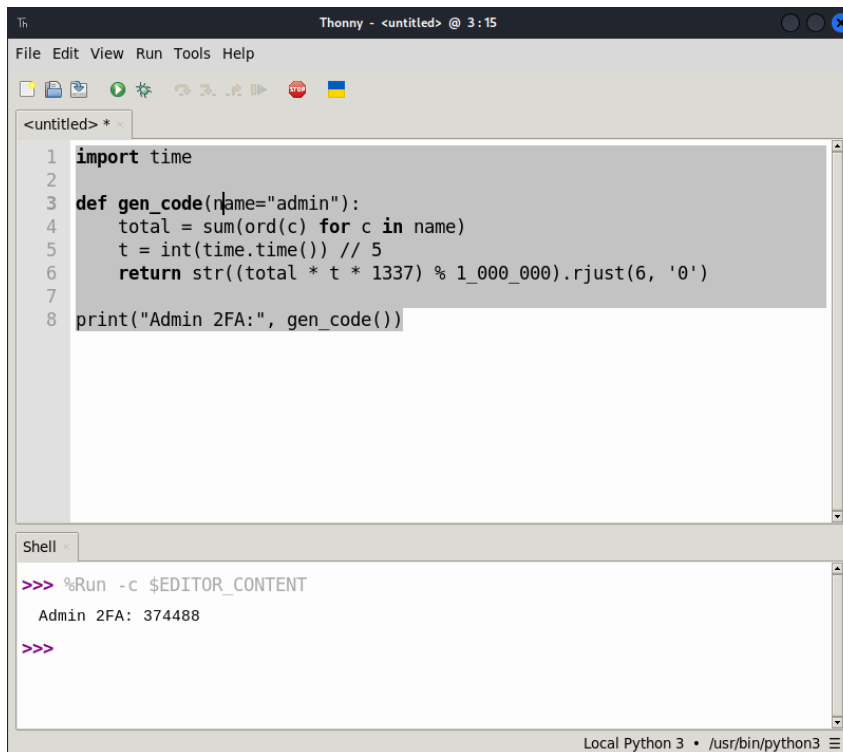
10. Running the script gets us our 2FA code.

11. If we go to that login portal we were given at the beginning, fill out the Capcha, and put in our 2FA code from Thonny (be quick, it resets every five seconds), we get the flag.

Login Portal

Login successful.
MetaCTF{andro1ds_want_m@ny_fact0rs_too}

Verification Code

??????

Logging in as admin. Please check your twoFactorAndroid app for the code. The code expires every 3 minutes.

Submit