# Анализ вредоносного программного обеспечения

## Лабораторный практикум

## Требования

* ОС Linux с установленными: unzip, apktool (<https://ibotpeaches.github.io/Apktool/documentation/>), openssl , backsmali(<https://github.com/jesusfreke/smali>), radare2 (<https://github.com/radare/radare2>);
* Предварительно загруженный файл https://github.com/oreilly-mlsec/book-resources/blob/master/chapter4/datasets/infected.apk
* Доступ в Интернет
* Предварительно установленные библиотеки для Python:
  + Pandas, numpy и matplotlib для работы с данными;
  + Scikit-learn, catboost, xgboost или подобные для работы с моделями машинного обучения.

## Постановка задачи

Требуется исследовать apk-файл на наличие подозрительной активности.

## Рекомендации по использованию инструментов для решения задачи

1. Распаковать apk файл с помощью unzip. Посмотреть AndroidManifest. Отметить подозрительные разрешения.
2. Убедиться, что большинство файлов бинарные и распаковать с помощью apktool decode.
3. Исследовать сертификат приложения командой

**openssl pkcs7 -in original/META-INF/CERT.RSA -inform DER -print**

1. Отметить информацию об авторах приложения.
2. Изучить ассемблер с помощью radare2.
3. Исследовать сетевую активность приложения с помощью mitmproxy и запуска эмулятора с помощью

**cd <ANDROID-SDK-LOCATION>/tools**

**emulator -avd pwned -wipe-data -http-proxy http://localhost:8080**

**adb install infected.apk**

1. Отметить подозрительную сетевую активность.

## Ожидаемый результат

Сформирован список признаков вредоносного ПО, сформированы предложение о том, как перевести признаки в вид, пригодный для подачи на вход модели машинного обучения.

## Рекомендованная литература

1. Чио К., Фримэн Д. Машинное обучение и безопасность / trans. Снастина О.В. М.: ДМК  Пресс, 2020. 388 с.