Università
della
Svizzera
italiana

Security ML **2023**

Student: Filippo Casari

Report for Assignment 2

1. Malware Detection

1.1. Understanding the Data

First, I would say that a lot of features are important to detect a malware in an Android application since certain permissions or using some APIs could trigger these features. It seems to me that the dataset is well structured.
Due to the fact that malicious apps frequently make unusual or suspicious API calls, the API call signatures are crucial characteristics for identifying Android malware. Similar to this, an application's requests for authorization can be a key sign of its potential for malicious behavior.
I am not an expert in Android applications, but ServiceConnector seems a very interesting feature that a malware can exploit. A ServiceConnection object is created when a component binds to a service using the bindService method. This object gets callbacks from the Android system when the connection to the service is made, when it is severed, or when an error occurs. Then, you can interact with the service by calling methods on its interface using the ServiceConnection object.
As I mentioned, since I do not have expertise in this field I cannot suggest other features. As I will explain in the following sections, my models work very well; no extra features are necessary to me indeed. Due to its nature, malicious apps may use this service to interact with a hidden or malicious service.

1.2. Evaluating Models

In this scenario, I would say that it is more impactful a false-negative since if we got a false-positive should not be a problem for our system. In fact, in the worst case scanario, our benign application will be killed by the detector. In contrast, if our detector is not able to detect properly a malicious app, we will allow the program to run anyway without control of its behavior.

1.3. Training Your Models

1.4. Quality of the Dataset

1.5. Bypassing your Model

1.6. Improving your Model

2. Hardware Trojan Detection