

UNIVERSITE CATHOLIQUE DE LOUVAIN

ECOLE POLYTECHNIQUE DE LOUVAIN



Leak sources analysis System

Setup

Supervisor: Dr. Ramin SADRE

by Mattieu DETAILLE
& Mohammad Muntasib Ali SYED

Contents

1 Setup guide **3**

2 requirements **3**

3 System overview **3**

4 Building your leak source detection system **4**

5 Running your leak source detection system **5**

1 Setup guide

This guide will help you to build a live automated leak source analysis system. This system will allow you to generate a leak source automated report while testing your smart device. You will only have to connect the smart device you want to test to a Wi-Fi you will have generated.

If you only want to generate a leak source automated report based on *pcap* (basic capture) and/or *txt* (*Man in the Middle attack* capture) files and thus not to perform any traffic capture automatically, it is possible! Just have a look in the *Manual.pdf* file. If you want only to perform the basic capture and thus not the *Man in the Middle attack*, you don't need to perform all the steps below (just the steps 5, 6 and 7 of the section 4).

The basic capture means that you will only capture some network traffic without performing a *Man in the Middle attack*. A *Man in the middle attack*, if it succeeds, will help you to have a look to your encrypted network traffic.

2 requirements

For building your leak source detection system, you will require:

- A smart device (smart phone, smart TV, etc) to test
- A USB stick able to generate a Wi-Fi
- An ODROID server running on Ubuntu.
- A hard drive of several hundreds of GBs.
- An ethernet cable
- A physically accessible modem able to access the internet.

3 System overview

On the figure 1, you can observe a summary of the method for capturing our network traffic. On this figure, the *black box* receives the network traffic of the smart device, saves it to a hard drive and outputs it to the internet. This *black box* can also receive some network traffic from the internet and send it to the smart device.

In order to explain this model as simply as possible, we describe the path followed by the network traffic from the smart device to the back-end servers of the service tested (to the internet).

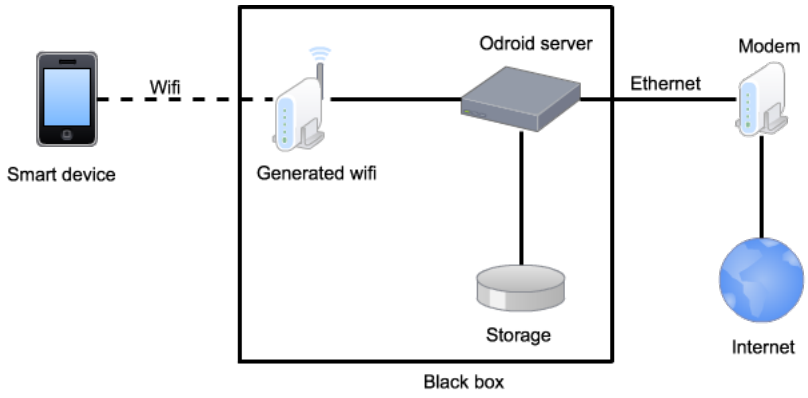


Figure 1: Network traffic capture model

First, the network traffic generated by the smart device is sent to a generated Wi-Fi. This generated Wi-Fi is maintained by a Wi-Fi USB stick[1]. We have indeed to configure the ODROID server such that it generates a Wi-Fi called *thesis*. The ODROID server is itself connected to a modem via an ethernet cable. It has thus access to the internet thanks to this modem and this ethernet cable.

Secondly, the network traffic of the smart device is received on the ODROID server. It is then captured either by *tshark* software for the *basic capture* or by the *Mitmproxy* software for the *Man in the Middle attack capture*. After having been captured, the network traffic is saved on a hard drive (*storage* on the figure 1) connected via USB to the ODROID server. It is finally transferred to the internet. This thus means that the network traffic of the smart device is finally transferred by the modem to the back-end servers of the service tested.

The network traffic that comes from the internet to the smart device tested follows the exact reverse path.

4 Building your leak source detection system

For building this system, you will have to follow these steps:

1. Start you ODROID (near your internet modem).
2. Copy the directory containing all our source code in your ODROID.
3. Plug your hard drive into your ODROID.
4. Modify the variable *MEDIA* at the beginning of the script *run.sh* to be your hard drive path (where you want to store the capture)

[1]We took the Model D-Link N300 for our experiment.

5. Fill the necessary information (= global parameters of the system) in the file *to_fill.py*. This file explains you precisely what to fill and why. It is really important to complete this file as precisely as possible as it will be used for finding your personal leak sources.

6. Run the script *init.sh*: *bash init.sh*

7. Update the *Safebrowsing database*[2] (it can take several hours) and put it in the *database directory*. If you want, you can put the database directory into your hard drive in order to gain some space on your ODROID. For that, just modify the path at the beginning of the file *safebrowsing.py* before updating the database.

8. Plug your ethernet cable into your ODROID and into your internet modem.

9. Plug your Wi-Fi USB stick into your ODROID.

10. Configure your ODROID for using the USB stick to generate a Wi-Fi named *thesis* (read the setting-up-generated-wifi.pdf file to perform this step)[3].

11. Connect your smart device to your generated Wi-Fi.

5 Running your leak source detection system

For running this system, read the *Manual.pdf* file.

[2]to update the database, run : python covert_channel.py -u ; the script is in the folder *Covert Channel/*

[3]Even if this tutorial is made for Raspberry Pi, it perfectly works for ODROID server!