CANDY: haCking infotAiNment AnDroid sYstems

https://youtu.be/aw0d-IoGD7E

Gianpiero Costantino

Ilaria Matteucci

Introduction

Vehicles are Cyber-Physical System (CPS):

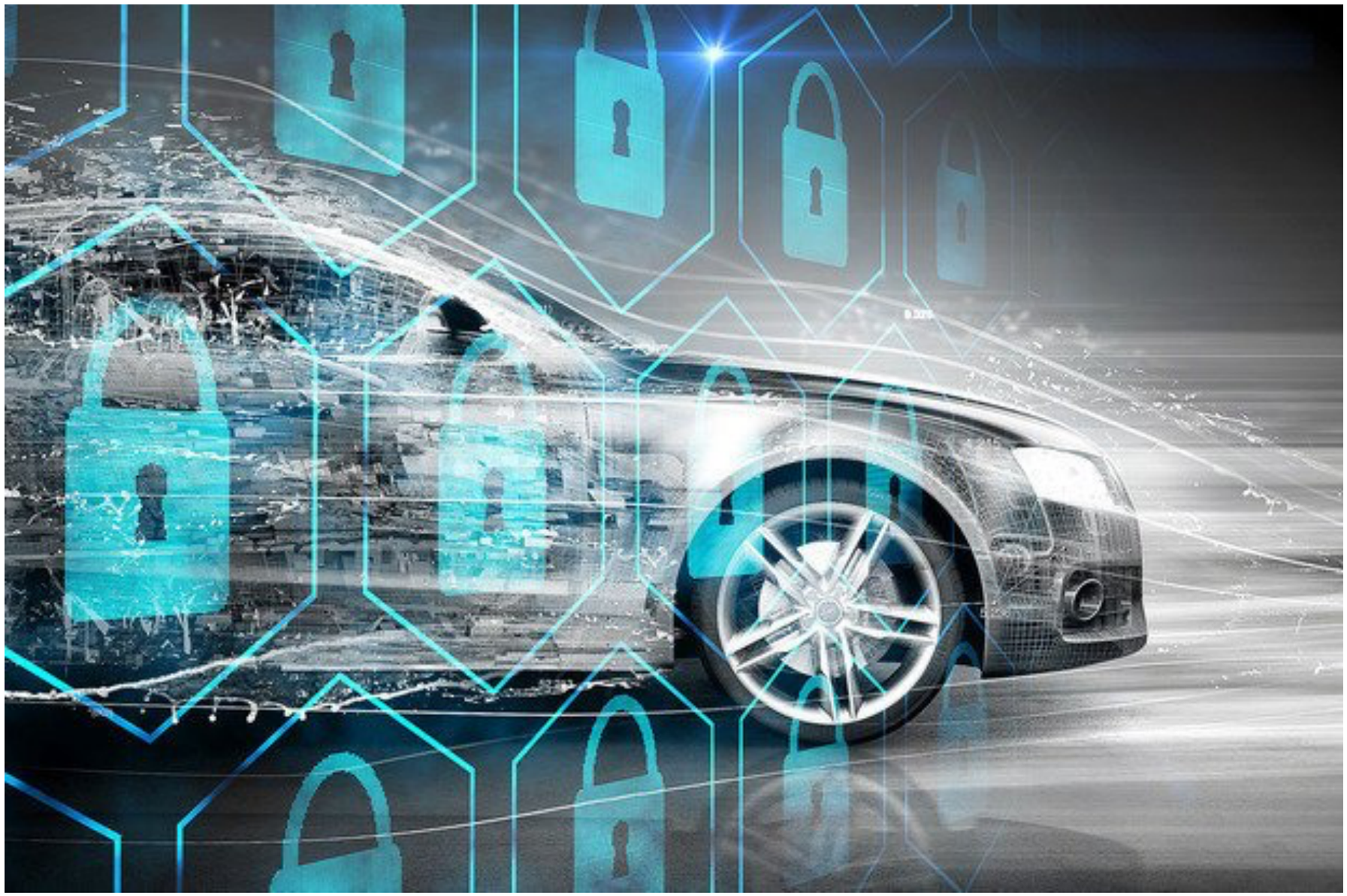
- ➡ Parking sensors
- ➡ Infotainment system
- ➡ Wireless connectivity
- ➡ Lane assistant

Safety-critical system are being exposed to security issues:

- ➡ Connectivity is the key enabler

Attack surface

Local Vs **Remote**

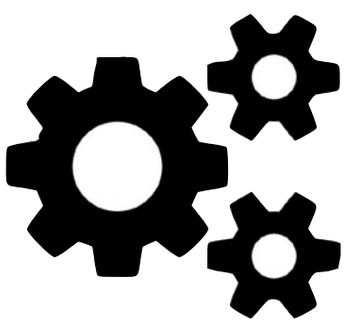


Attack on Jeep Cherokee

Remote Exploitation of an Unaltered Passenger Vehicle.
C.Miller and C. Valasek, BlackHat 2015

CANDY

Hacking CAN bus vehicle communications by remotely injecting a Trojan-horse on the Android In-Vehicle Infotainment system

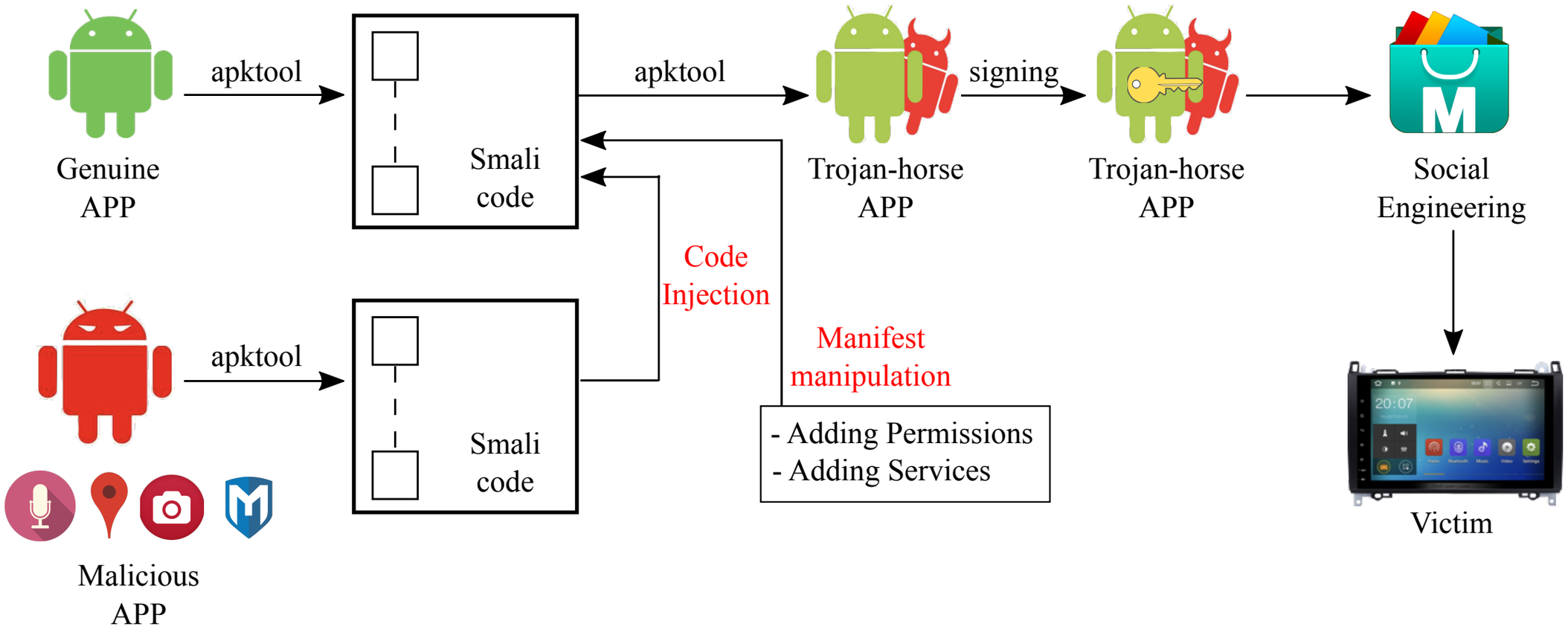


Running the attack: ***the target device***

- Bosion Android Radio with Android 4.4 KitKat
- Installed on a Volkswagen Golf 1.6 TDI
- Connected to the CAN bus network through a CAN bus-decoder
- The radio is connected to the Internet through a 3G dongle

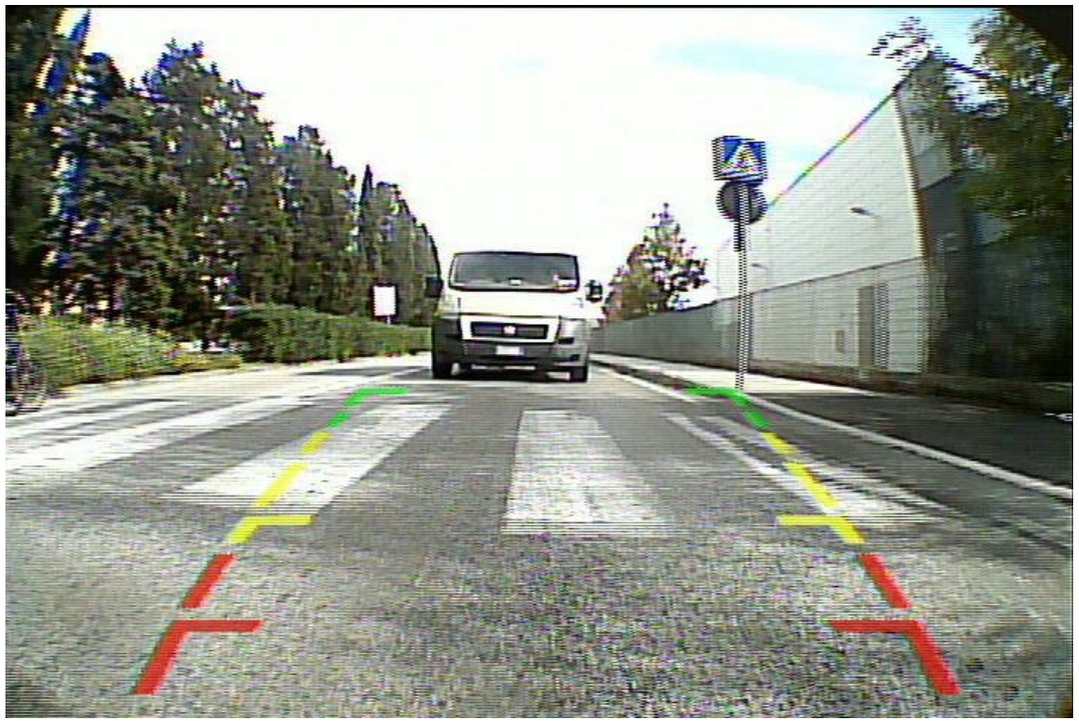
Attack Work-flow

In collaboration with Antonio La Marra (IIT-CNR)



I. ***Remotely accessing*** the In-Vehicle Infotainment system
II. ***Recording*** driver's voice
III. ***Taking*** photos and ***grabbing*** vehicle's trajectories
IV. ***Collecting*** information spread on the CAN bus

Photos from parking-camera

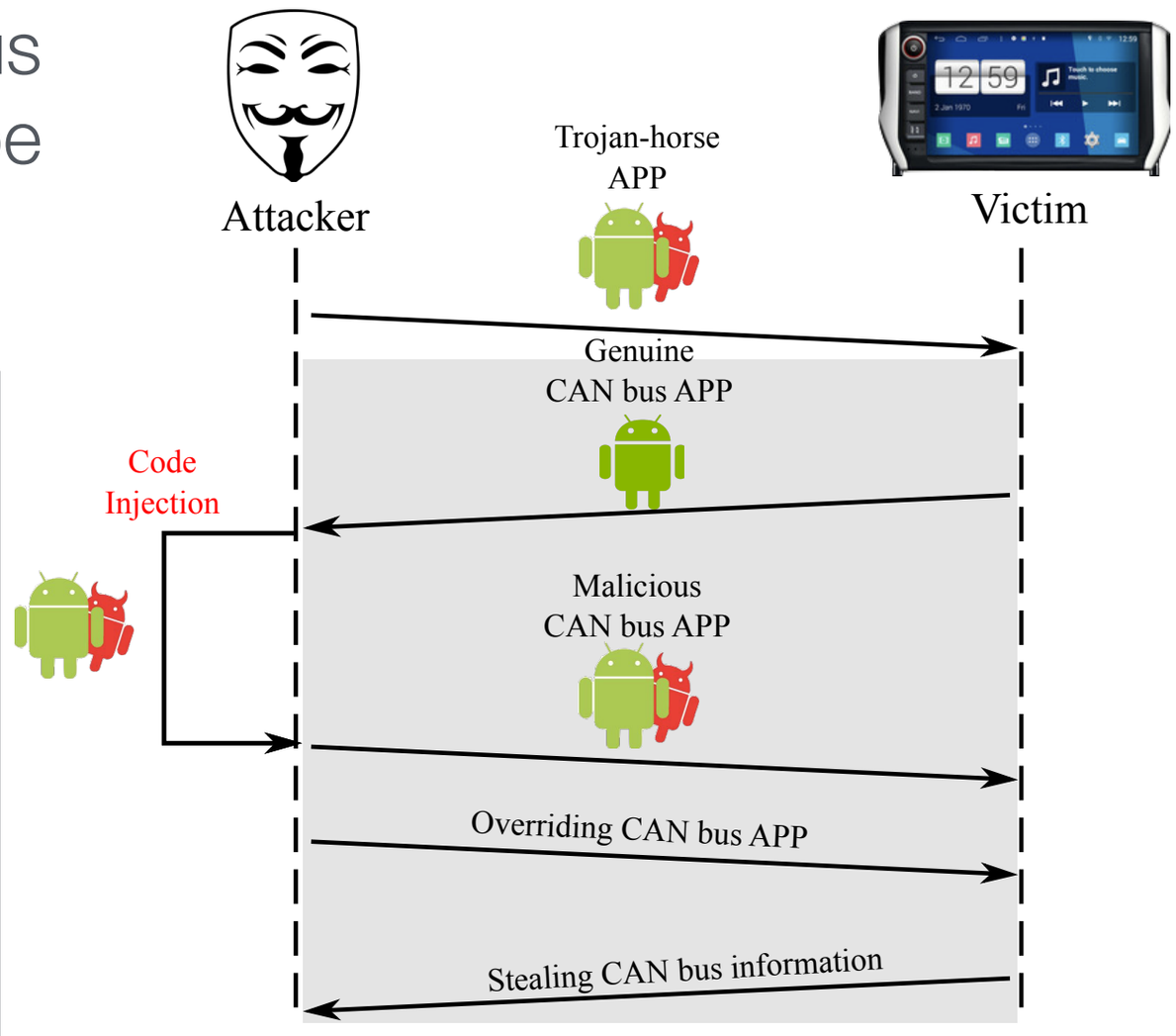


Vehicle’s trajectories

Stealing CAN bus information

The attacker downloads and modifies the original APP to store the CAN bus information on files that later on can be downloaded

Water temperature
Seat belt attached or not
Handbrake pulled or not
Car doors status
Remaining fuel
Voltage of the battery
Engine rpm
Car speed
Air conditioning system status
Distance from an obstacle



CAN bus data


modifiedCarInfo
21:02
Carburante
0L
Cintura
OK
GiriMotore
0RPM
Tronco
Off
Out temp
21.0℃
Freno a mano
Off
VelocitàN
0.0km/h
Batteria
12.1V
acqua
OK
Chilometraggio
126421km
Avvertimento !
Sbloccato !

Our research directions

Studying vulnerabilities:

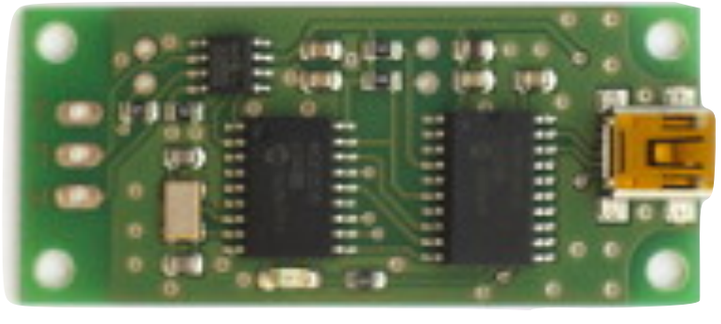
- (*Can level*) Analyzing and learning CAN messages
- (*Firmware level*) Studying the firmware's code

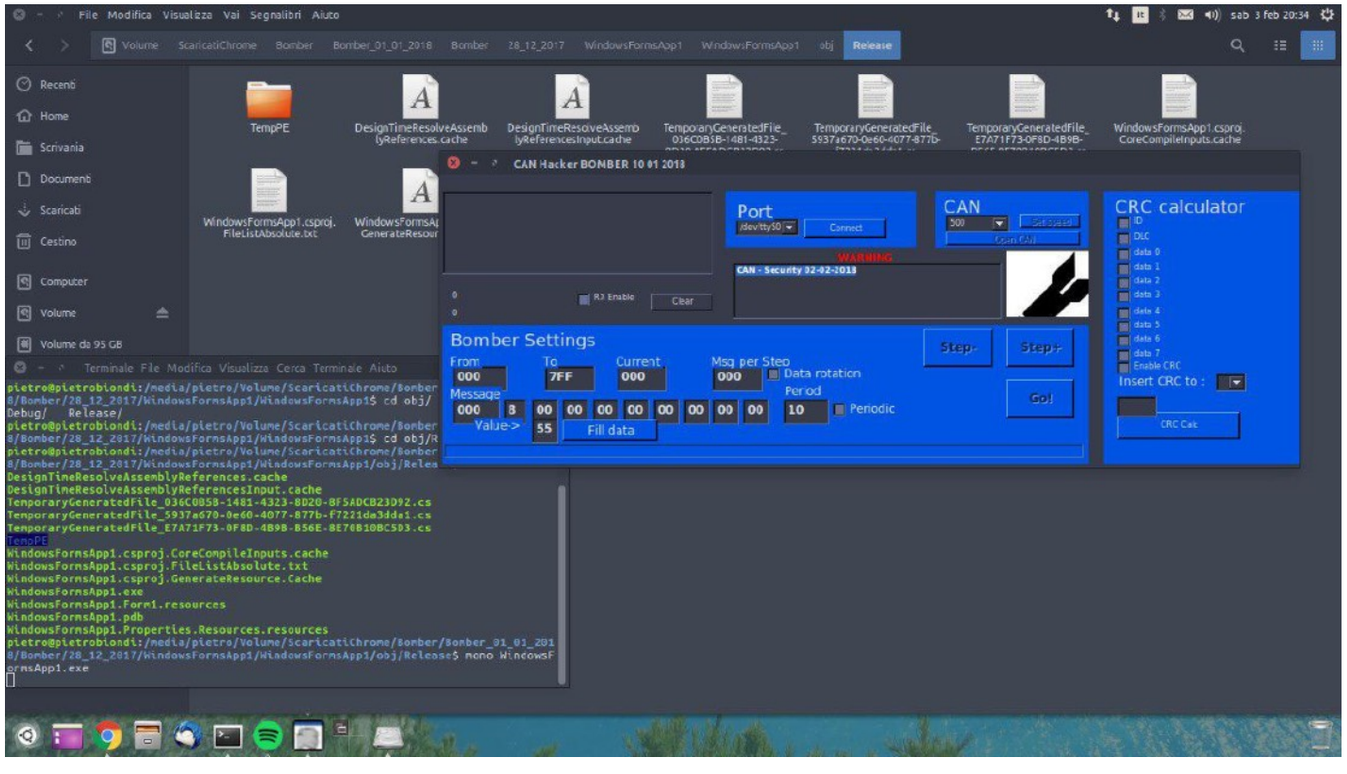
Security for vehicles:

- adding security properties to the CAN protocol
- studying drivers' attitude in V2V and V2X Infrastructure

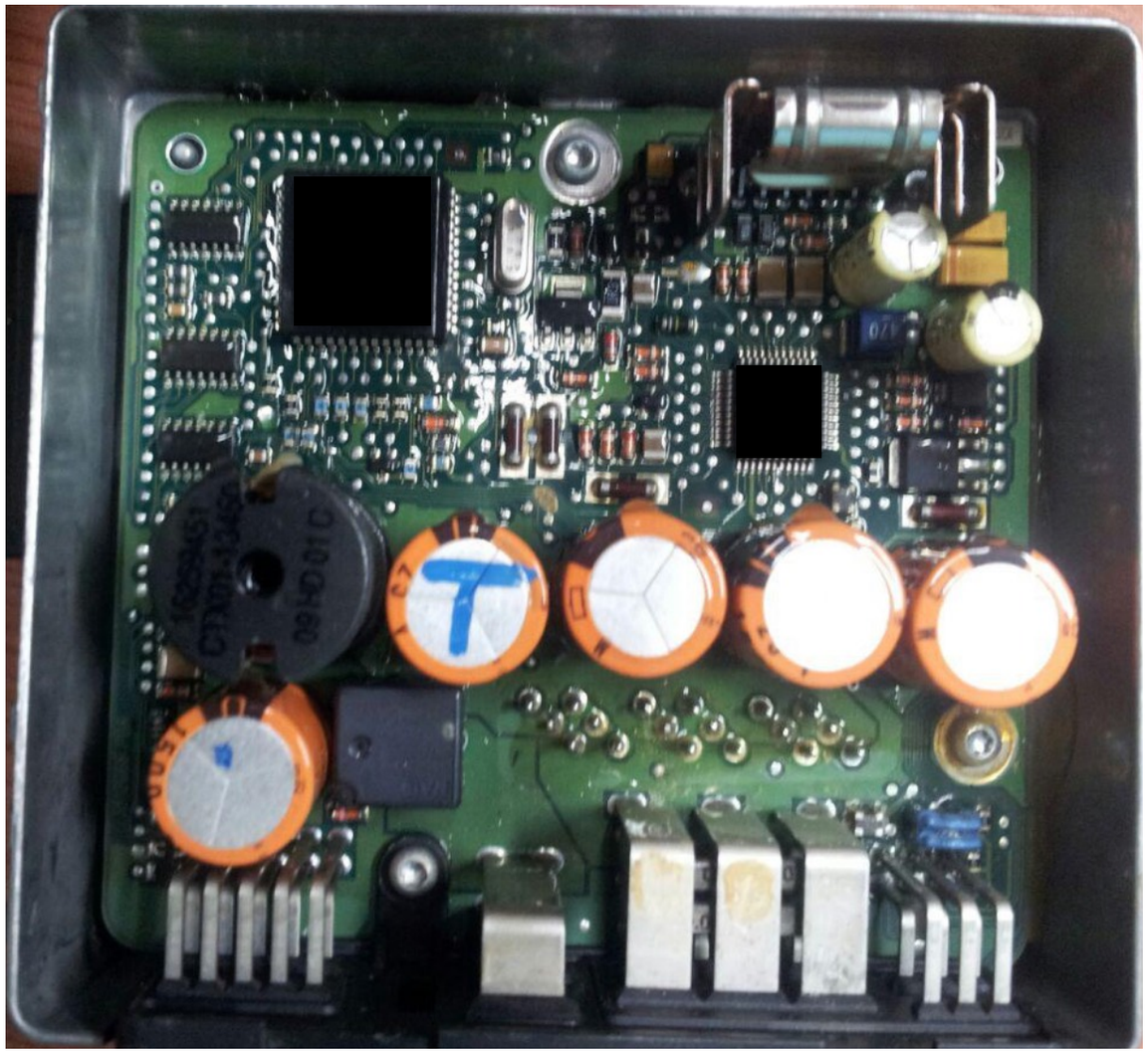
Penetration Testing **@CAN level**

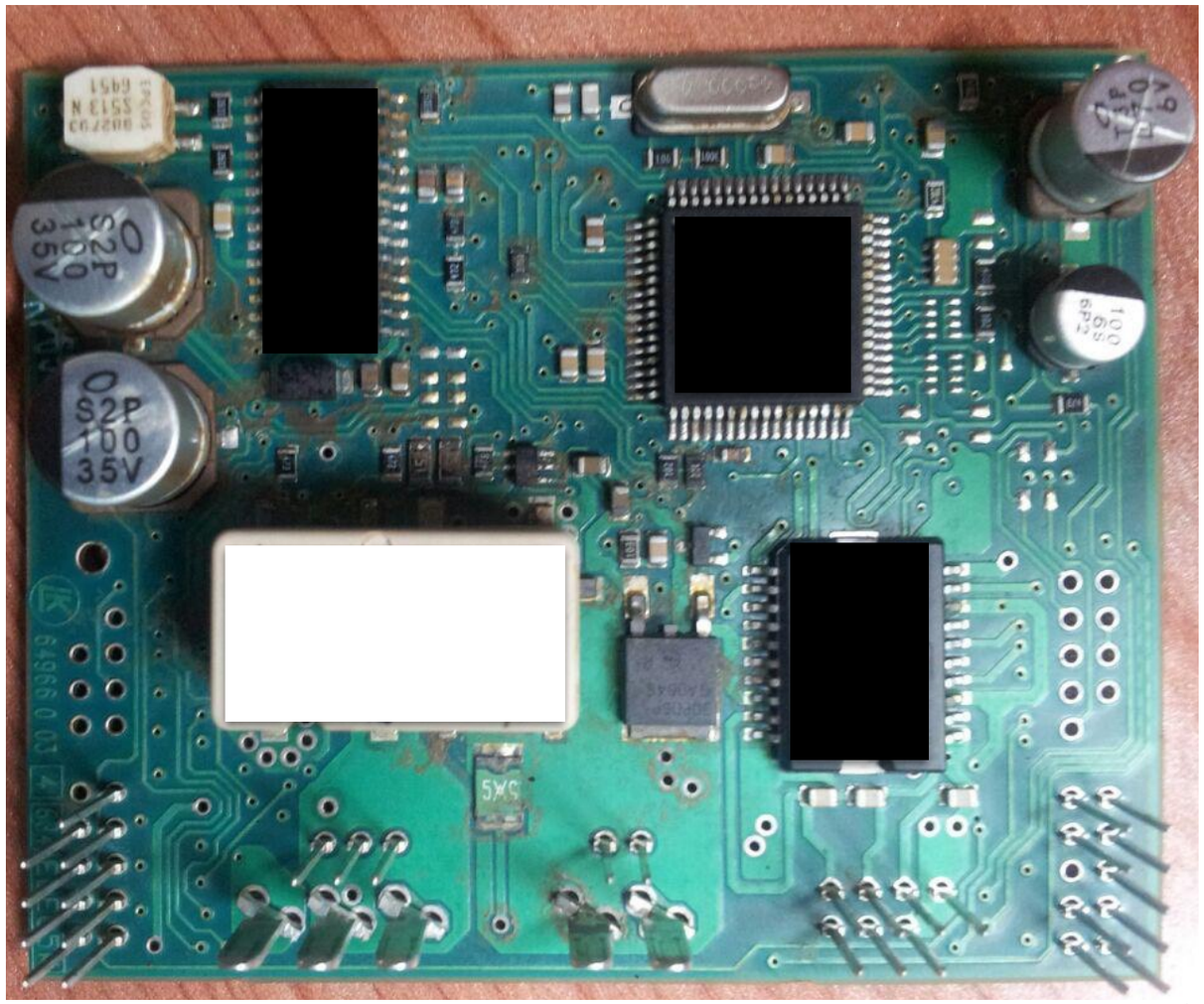
- **Receiving and analyzing CAN messages** by connecting ECUs to PCs via USBtin
- **Learning the messages' content** using reverse engineering technique (*or brute-force attack*)
- **Sending incorrect messages** to alter the behavior of the vehicle (*Man in the Middle*)



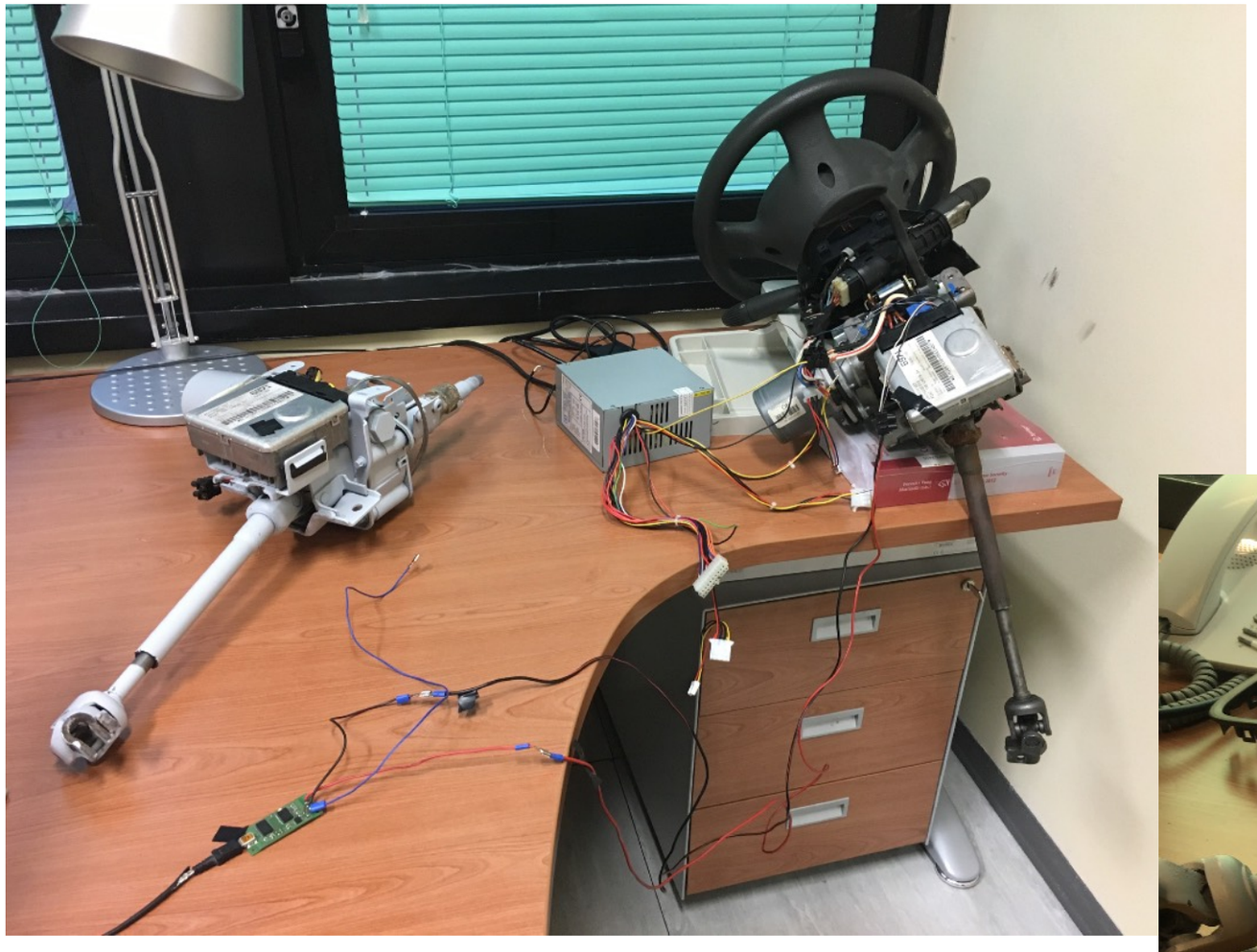


Penetration Testing **@Firmware level**





Our lab



The CAN bus as is

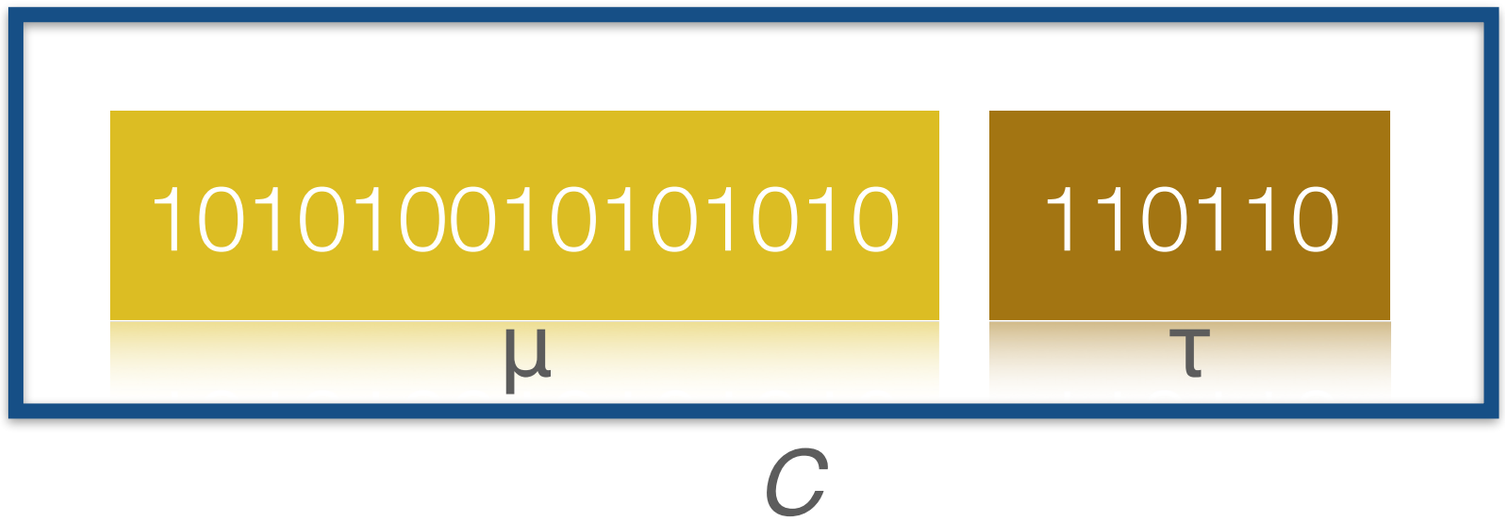
CAN bus is the communication protocol within ECUs of vehicles:

- ➡ Max data-message length is **64bit**
- ➡ **!**Authentication and **!**Integrity and **!**Confidentiality

Model based Design:

CIA solution

Turning CAN messages into ***Security by Design*** format



Confidentiality, **I**ntegrity and **A**uthentication

Future Work

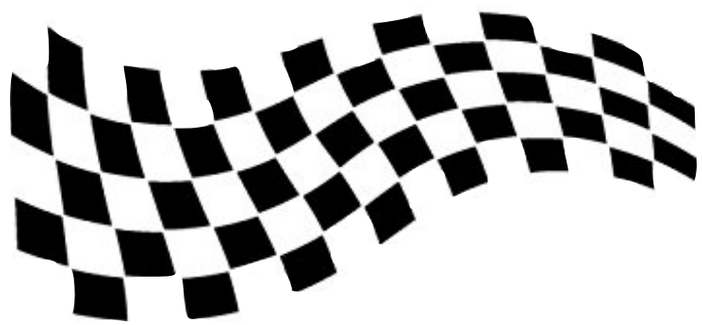
Working on a way to send messages on the CAN bus network from the IVI Android.

- *To give more impact to CANDY and to point out the vulnerabilities of the CAN protocol*

Working on a Security-by-Design framework compatible with automotive standards.

- *To the security of ICT systems in vehicles as well as optimize the trade-off between security and safety aspects in the automotive domain.*

Thank you!



CANDY: ha**C**king infot**AiN**ment An**D**roid s**Y**stems