

# Report for the CPS Mid-Course Project

Andrea Auletta

`andrea.auletta@studenti.unipd.it`

Niccolò Zenaro

`niccolo.zenaro@studenti.unipd.it`

December 24, 2024

# Contents

<b>1</b>	<b>Introduction and objectives</b>	<b>3</b>
<b>2</b>	<b>System setup</b>	<b>3</b>
<b>3</b>	<b>Experiments</b>	<b>3</b>
3.1	Fabbrication attack . . . . .	3
3.2	Suspension attack . . . . .	4
3.3	Masquerade attack . . . . .	4
3.4	CIDS algorithm . . . . .	4
<b>4</b>	<b>Results and Discussion</b>	<b>4</b>

# 1 Introduction and objectives

In this report we will describe the work done for the mid-course project of the course Cyber-Physical Systems and IoT security. The paper to which we refer is [1] “**Fingerprinting Electronic Control Units for Vehicle Intrusion Detection**” where is explained the design of an algorithm: the CIDS (Clock-based Intrusion Detection System), that is able to detect different kinds of attack. In this paper is shown the functioning of three kinds of attack: the ”Fabbriation attack”, the ”Suspension attack” and the ”Masquerade attack”. What we tried to do was to implement the three attacks and the CIDS algorithm in a simulation environment using **ICSim**, instead of physically as shown in the paper.

## 2 System setup

We worked on a virtual machine with **LinuxMint22** as operating system. The simulator used is **ICSim**. All the code is written in **Python** and can be found at the following link: . We used the **can library** to create messages and to simulate the communication of the attackers with the ECUs.

## 3 Experiments

For each attack we are assuming that the attacker is already in the system and is able to send messages to several ECUs. To emulate a periodic ECU so that you can at least see something in the simulator we created the script *fakeECU.py* where are sent messages every 2 seconds to open and close a certain number of doors of the car.

### 3.1 Fabbriation attack

In the Fabbriation attack we are trying to inject messages into the CAN bus to make the system behave in a different way. Essentially, we created a message with the ID of the doors in a way that in the middle of those two seconds gap between legitimate messages we sent other commands to open and close the doors and causing it to function differently from normal behavior.

### 3.2 Suspension attack

In the Suspension attack we are trying to stop the receiving of messages sent by a legitimate ECU. We made a DoS like attack: we sent a big quantity of messages with the ID of the left arrow into the CAN bus in order to lose the messages sent by it. Here we put a very small time sleep, precisely because the aim is to clog the network and not receive packets from legitimate ECUs. The result was that if we were trying to turn on the left arrow, it was not working.

### 3.3 Masquerade attack

In the Masquerade attack we are trying to send messages with the ID of the legitimate ECUs. The idea here is that we have to shield the fact that an ECU is compromised. After running the weak ecu script, the attacker will listen the CAN bus for 20 seconds, in this way it can calculate the period of the messages sent by the weak ECU and then it will send messages with the same ID and the same period. We decided to send a message to open different doors with respect to the ones opened by the weak ECU, and this can be seen at simulation time.

### 3.4 CIDS algorithm

## 4 Results and Discussion

## References

- [1] Kyong-Tak Cho and Kang G. Shin. “Fingerprinting Electronic Control Units for Vehicle Intrusion Detection”. In: *Proceedings of the 25th USENIX Conference on Security Symposium* (2016), pp. 911–927.