Project Assignment

Projects can be done by groups with no more than 3 persons (3 or 2 persons is recommended).

Each group should choose a project assignment (different group can choose the same).

Each group will implement the project with the related tool and produce a small report (up to 5 pages) and present the work with a PowerPoint presentation (15 minutes).

During the presentation every member will discuss a part of the project and every member will run the project on its computer. Every member should be able to implement small modifications on the fly during the presentation.

Steps for the project:

1. Choice of the project topic and communication of the student names to the teacher (additional material will be given for the chosen project)

2. Schedule meetings to discuss about design choices and work done in the project

3. After approximately 2/3 of the project, review with the teachers of the current state and initial results

4. Submission of the project:
   - The project must be documented including the design choices
   - The code and the documentation must be submitted as a single zip file.

The project must be submitted to teachers at least two days before the date of the exam.

Project evaluation: unsatisfactory, satisfactory, good.

To attend the oral test, it is necessary to have a positive evaluation of the project.
The evaluation is a bonus for the final mark.

Project 1

Create an ADVISE model for different Adversaries trying to attack the back-end servers related to autonomous vehicles, including the threats suggested in the next table.

High level and sub-level descriptions of vulnerability/ threat			Example of vulnerability or attack method	
Threats regarding back-end servers related to vehicles in the field	1	Back-end servers used as a means to attack a vehicle or extract data	1.1	Abuse of privileges by staff (**insider attack**)
			1.2	**Unauthorized internet access** to the server (enabled for example by backdoors, unpatched system software vulnerabilities, SQL attacks or other means)
			1.3	**Unauthorized physical access** to the server (conducted by for example USB sticks or other media connecting to the server)
	2	Services from back-end server being disrupted, affecting the operation of a vehicle	2.1	**Attack on back-end server stops it functioning**, for example it prevents it from interacting with vehicles and providing services they rely on
	3	Vehicle related data held on back-end servers being lost or compromised ("data breach")	3.1	Abuse of privileges by staff (**insider attack**)
			3.2	**Loss of information in the cloud**. Sensitive data may be lost due to attacks or accidents when data is stored by third-party cloud service providers
			3.3	**Unauthorized internet access to the server** (enabled for example by backdoors, unpatched system software vulnerabilities, SQL attacks or other means)
			3.4	**Unauthorized physical access to the server** (conducted for example by USB sticks or other media connecting to the server)
			3.5	**Information breach** by unintended sharing of data (e.g. admin errors)

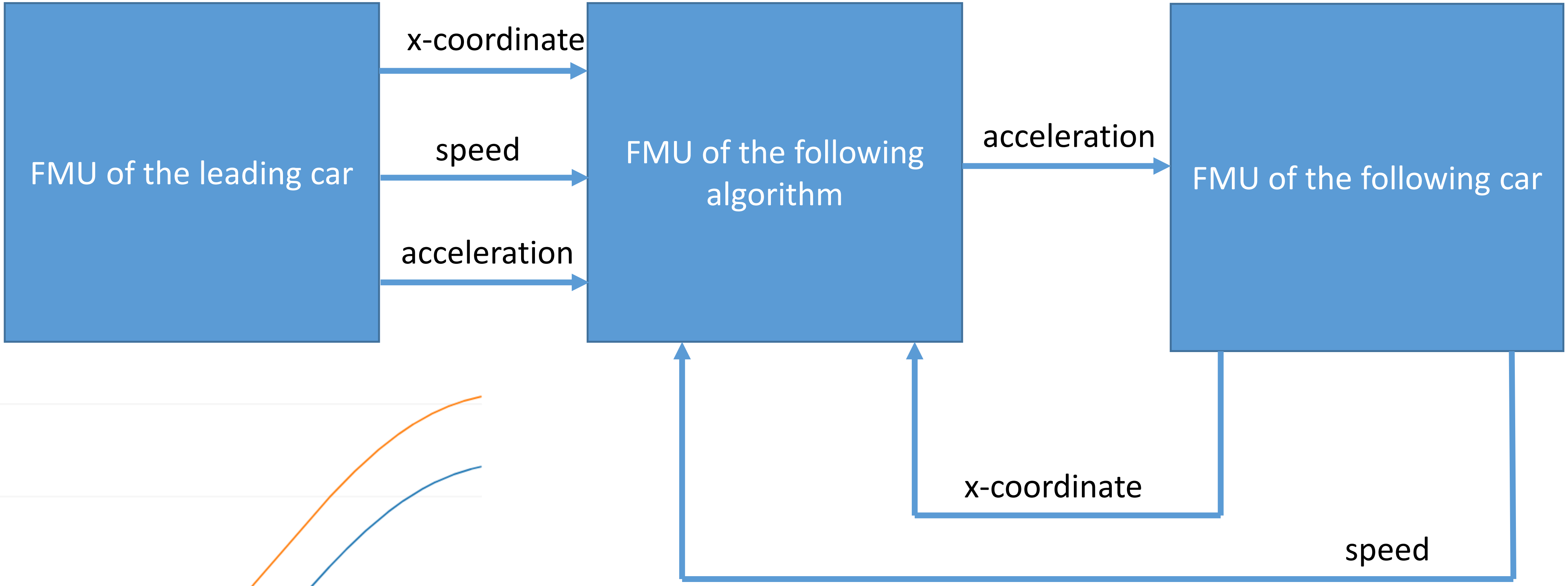
Project Development

1. Create an attack tree with roughly 20 elements (knowledge, access, skill, attack step and goal).
2. Create 3 different adversary profiles
    1. Insider
    2. Hacker
    3. Physical Intruder
3. Evaluate the probability of each adversary to achieve the goals.
4. Analyze the results.

Project 2

Co-simulate a scenario with an autonomous car following a leader car with a desired distance of 15 meters.

Analyse reasonable data alteration attacks that can lead to a crash between the two car.



FMU of the leading car

x-coordinate

speed

acceleration

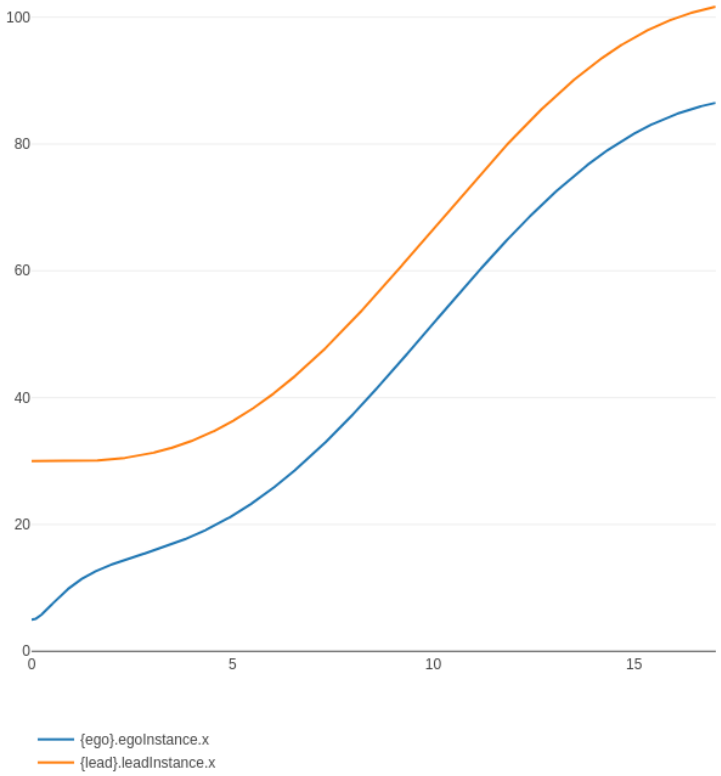
FMU of the following algorithm

acceleration

FMU of the following car

x-coordinate

speed



The three fmus will be provided

Students can use the attack fmu introduced during the course to implement the data alteration attack

Project Development

1. Create the INTO-CPS project and co-simulate the vanilla scenario
2. Consider two scenarios: 1) an attack altering the value of x-coordinate between the Ego car and the Following car and 2) an attack alterning the value of accellerazione della following car
3. Analyze the resulting behavior in both cases
4. An attack can last from a few steps to many steps, and it can recur many times. Show an attack together with a case that does not lead to a crash between the two cars, and a case in which there is a car crash.

Project 3

Apply model checking for malware analysis in a subset of java bytecode.

Build a state machine for the abstract execution of the bytecode and describe the malware pattern as a CTL formula.

Project Development

1. Consider the following subset of java bytecode instructions {pop, push, load, ldc, store, if, new, invoke, return} and the malware pattern: “*new android/content/IntentFilter* ; *…; ldc ……;".*

2. Define an abstraction of the bytecode, by removing information not needed for the analysis.

3. Model the transition system of the execution of the abstract bytecode in NUSMV and check the presence of the patter using a CTL formula.

4. Assume code obfuscation, resulting in the pattern split into two distinct methods.

5. Check the presence of the malware in the modified bytecode.