MotorIST project, SIRS

Elie Bruno,
Flavien Valea,
Tanguy Vésy

Owner/s

Mechanic

Manufacturer

Car

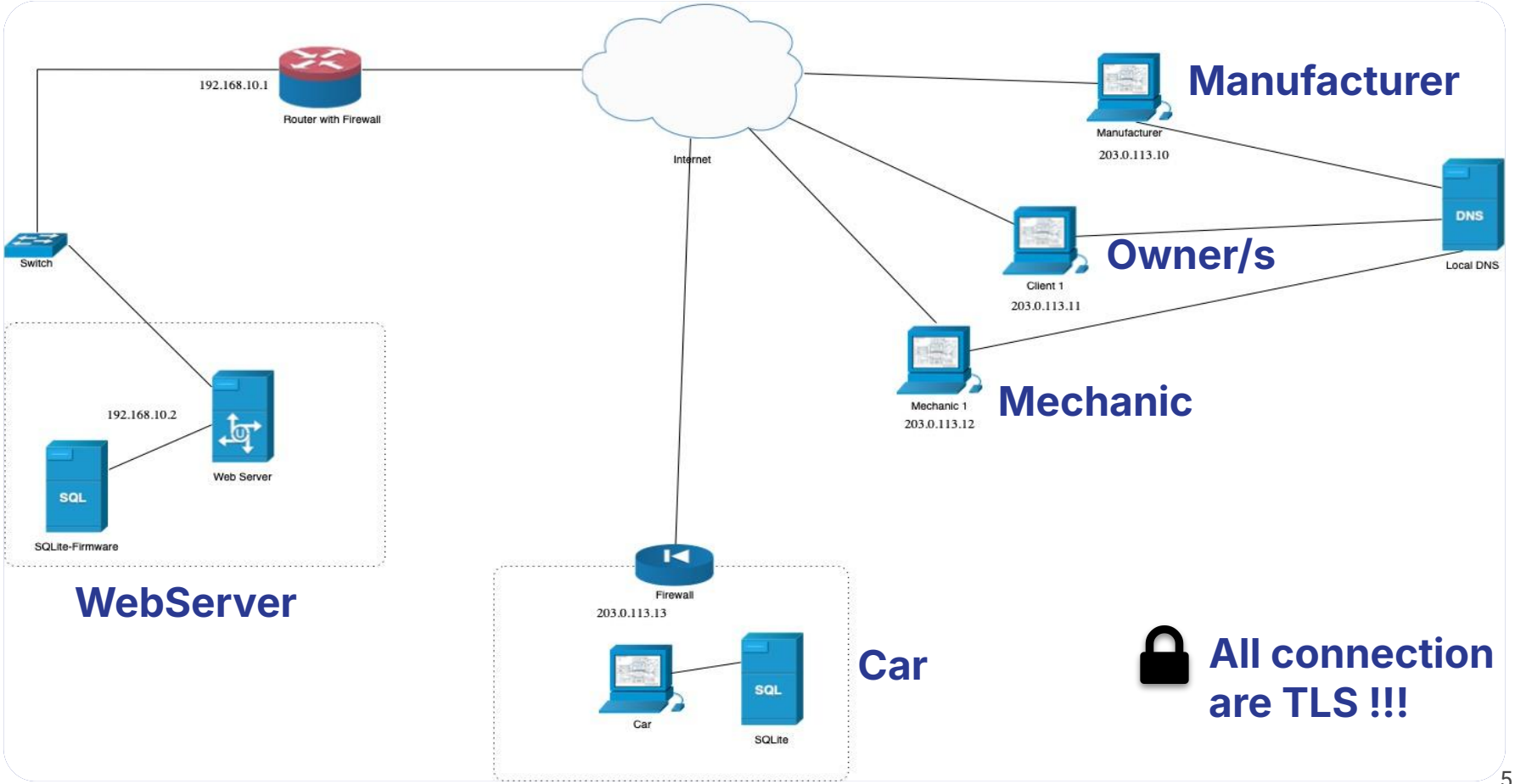
Goals

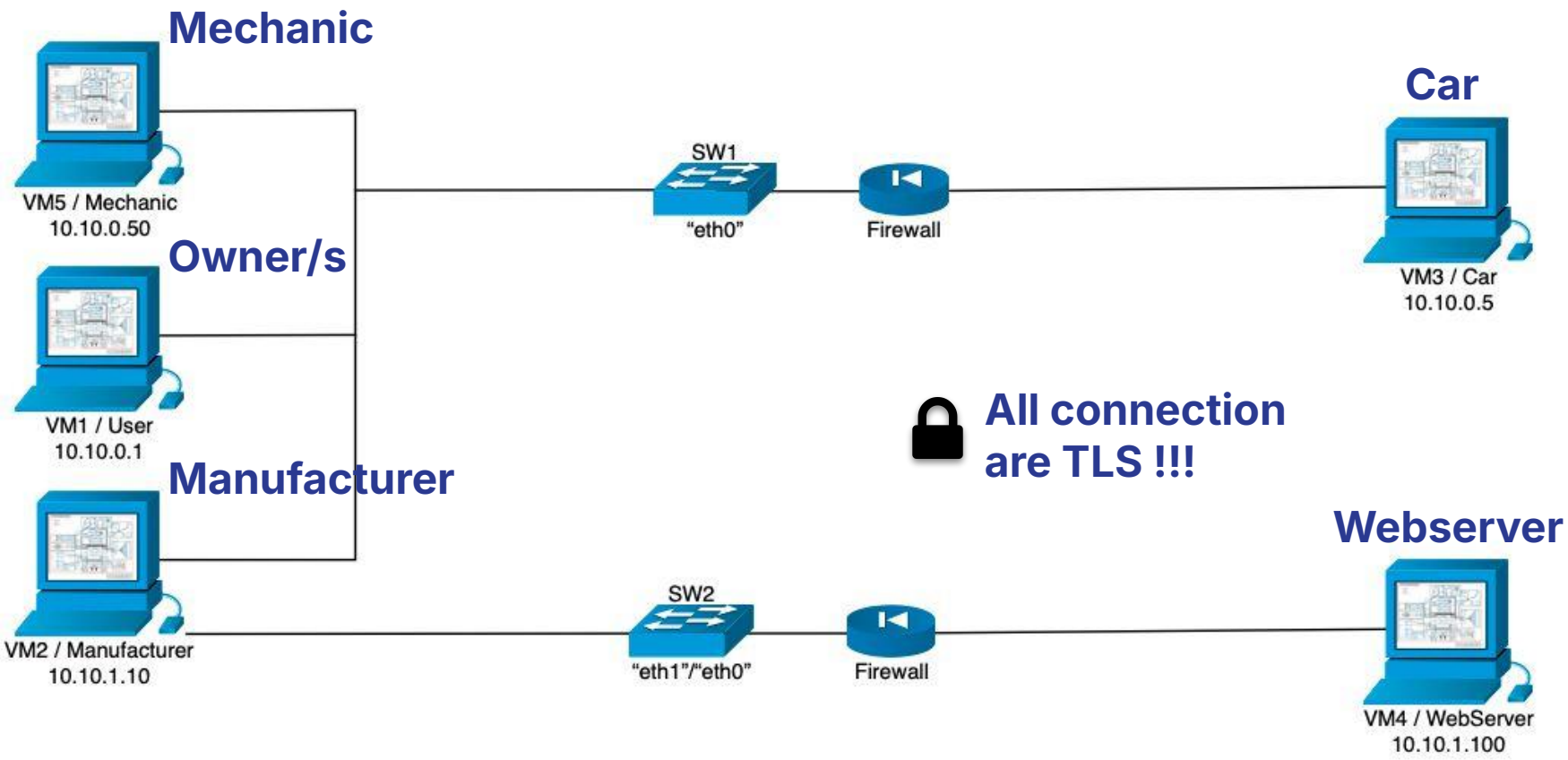
- [SR1: Confidentiality] The car configurations can only be seen by the car owner.
- [SR2: Integrity 1] The car can only accept configurations sent by the car owner.
- [SR3: Integrity 2] The car firmware updates can only be sent by the car manufacturer.
- [SR4: Authentication] The car manufacture cannot deny having sent firmware updates.

SRA-Multiple users-Audit

Security challenge

- [SRA1: data privacy] One user cannot know the configuration of the other user, but may know some current information of the car.
- [SRA2: authorization] An unauthorized user cannot change the configuration of the other user.
- [SRA3: authenticity] It must be possible to audit the car and verify which configuration actions were performed by which users.


192.168.10.1
Router with Firewall
Internet
Manufacturer
Manufacturer
203.0.113.10
DNS
Local DNS
Client 1
203.0.113.11
Owner/s
Switch
Mechanic 1
203.0.113.12
Mechanic
192.168.10.2
Web Server
SQL
SQLite-Firmware
WebServer
Firewall
203.0.113.13
Car
SQL
SQLite
Car
All connection are TLS !!!


Mechanic
VM5 / Mechanic
10.10.0.50
Owner/s
VM1 / User
10.10.0.1
Manufacturer
VM2 / Manufacturer
10.10.1.10
SW1
"eth0"
Firewall
Car
VM3 / Car
10.10.0.5
All connection
are TLS !!!
SW2
"eth1"/"eth0"
Firewall
Webserver
VM4 / WebServer
10.10.1.100

Hybrid cryptographic scheme

- **AES-256/GCM** for data encryption (fast, symmetric),
- **RSA** to securely encrypt the AES key (asymmetric),
- **ECDSA + SHA3-256** for integrity and authenticity (digital signatures).
- adds **timestamp and nonce** for freshness
- It uses **BouncyCastle** to provide certain algorithms and maintain compatibility.

The secure document format

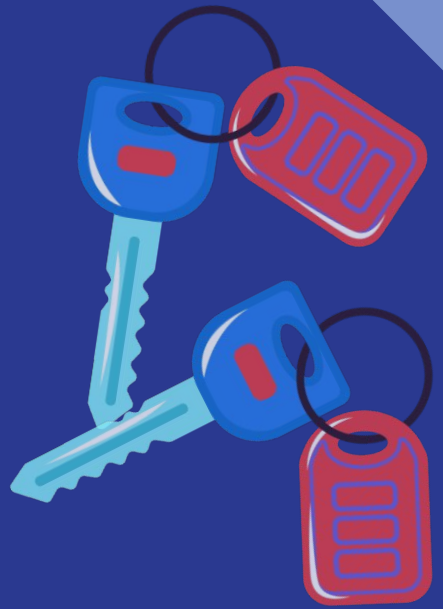
```
{
  "carID": "1234XYZ",
  "user": "user1",
  "private_configuration": {
    "ac": [
      { "out1": "789" },
      { "out2": "1011" }
    ],
    "seat": [
      { "pos1": "0" },
      { "pos3": "6" }
    ]
  },
  "public_car_info": {
    "battery_level": "75%",
    "total_mileage": "15000"
  }
}
```

```
{
   "cipherText": "R29vZCBqb2Igd2VsbCBkb25lIQ...",
   "encryptedAesKey": "QmFzZTY0RW5jb2RlZEtleQ...",
   "iv": "bXlwcm9qZWN0aXY..",
   "signature": "U2lnbmF0dXJlRm9ySW50ZWdyaXR5",
   "sender": "user123",
   "receiver": "car456"
 }
```

Key Distibution

Each entity has its own pair of private and public RSA and EC key given by the manufacturer when the car was build

In a real case we would have opted for a manual distribution

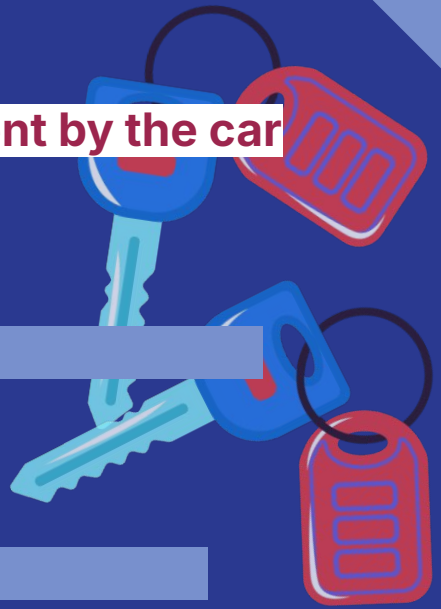


Deal with the challenges..

- [SR1: Confidentiality] The car configurations can only be seen by the car owner.

- [SR2: Integrity 1] The car can only accept configurations sent by the car owner.

  The car and user have both a pair of RSA and EC keys that guarantee the confidentiality and the integrity of the configurations sent to the car



Deal with the challenges..

The manufacturer also has it’s own private key that is trusted and the message is sent with a timestamp + nonce. The manufacturer signs the message.

- [SR3: Integrity 2] The car firmware updates can only be sent by the car manufacturer.

- [SR4: Authentication] The car manufacture cannot deny having sent firmware updates.

Security Challenge A

Security challenge

- **[SRA1: data privacy] One user cannot know the configuration of the other user, but may know some current information of the car.**

The user can not know the configuration of another user because he doesn’t have his private key. He can know public information about the car because he is a user “mock info”

Security Challenge A

Security challenge

- [SRA2: authorization] An unauthorized user cannot change the configuration of the other user.

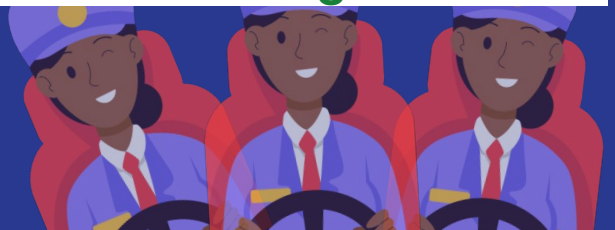
  The car knows the keys of the user and based on that accept the change or not.

Security Challenge A

Security challenge

- [SRA3: authenticity] It must be possible to audit the car and verify which configuration actions were performed by which users.

  The car database stores encrypted all the message that were received as well as the signature



DEMO