An Introduction to Car Hacking

Analyzing Proprietary Automotive Systems with CANalyzat0r

who?

Philipp Schmied

IT Security Consultant @ SCHUTZWERK

Thesis: Car Hacking | Bug Bounty

❤️ RE, Exploit Development

@CaptnBanana

github.com/ps1337

(Likes slide effects)

[1]

What Is This?

My journey in car hacking (so far)

Not just CAN hacking

OK But Why?

About Car Hacking

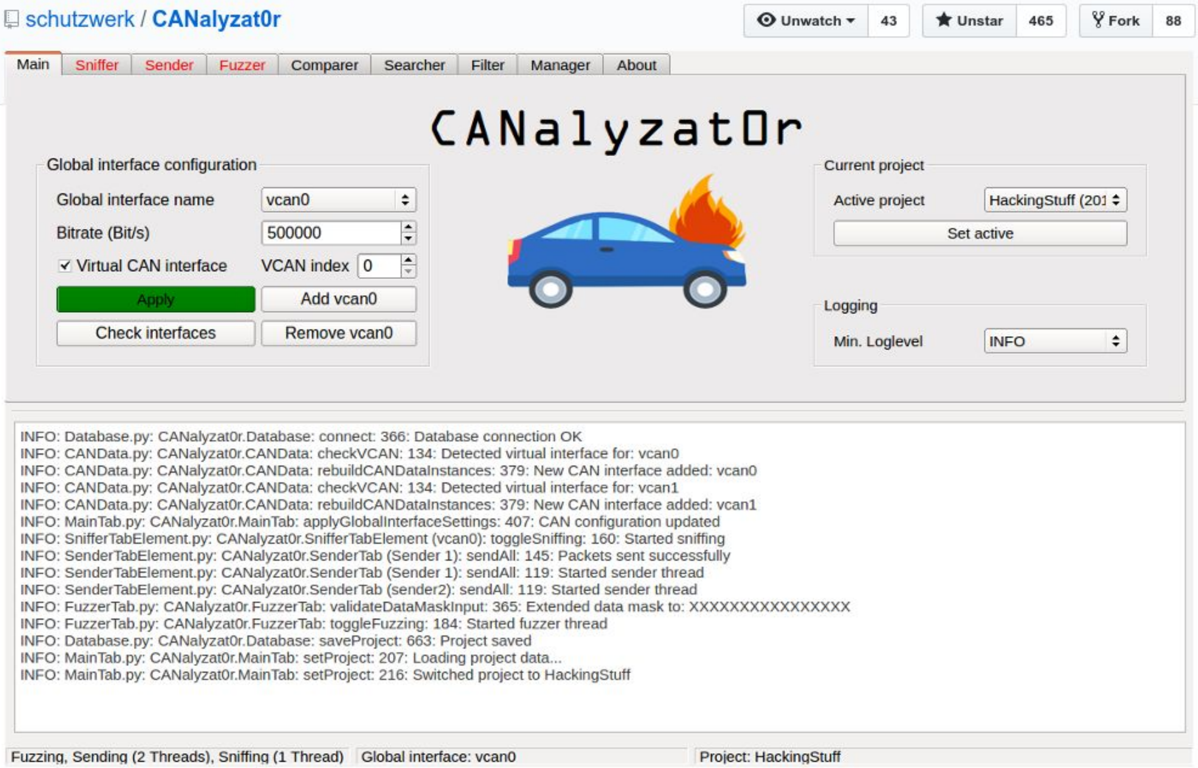
- Car ~ small corporate network
- Interconnected ECUs and sensors
- Proprietary software and services
  - provided by 3rd parties
  - trusted? reviewed? tested?
  - privileges? vulnerabilities?
  - same for ECUs

About Car Hacking

- Variety of interfaces
  - CAN, OBD
  - USB (audio, video, images, vCard, …)
  - GPS, BT, WiFi, …

About Car Hacking

- Car Hacking: Obscure sector
- But also: variety
  - Findings
  - Methodology



Global interface configuration

Global interface name: vcan0

Bitrate (Bit/s): 500000

☑ Virtual CAN interface

VCAN index: 0

Current project

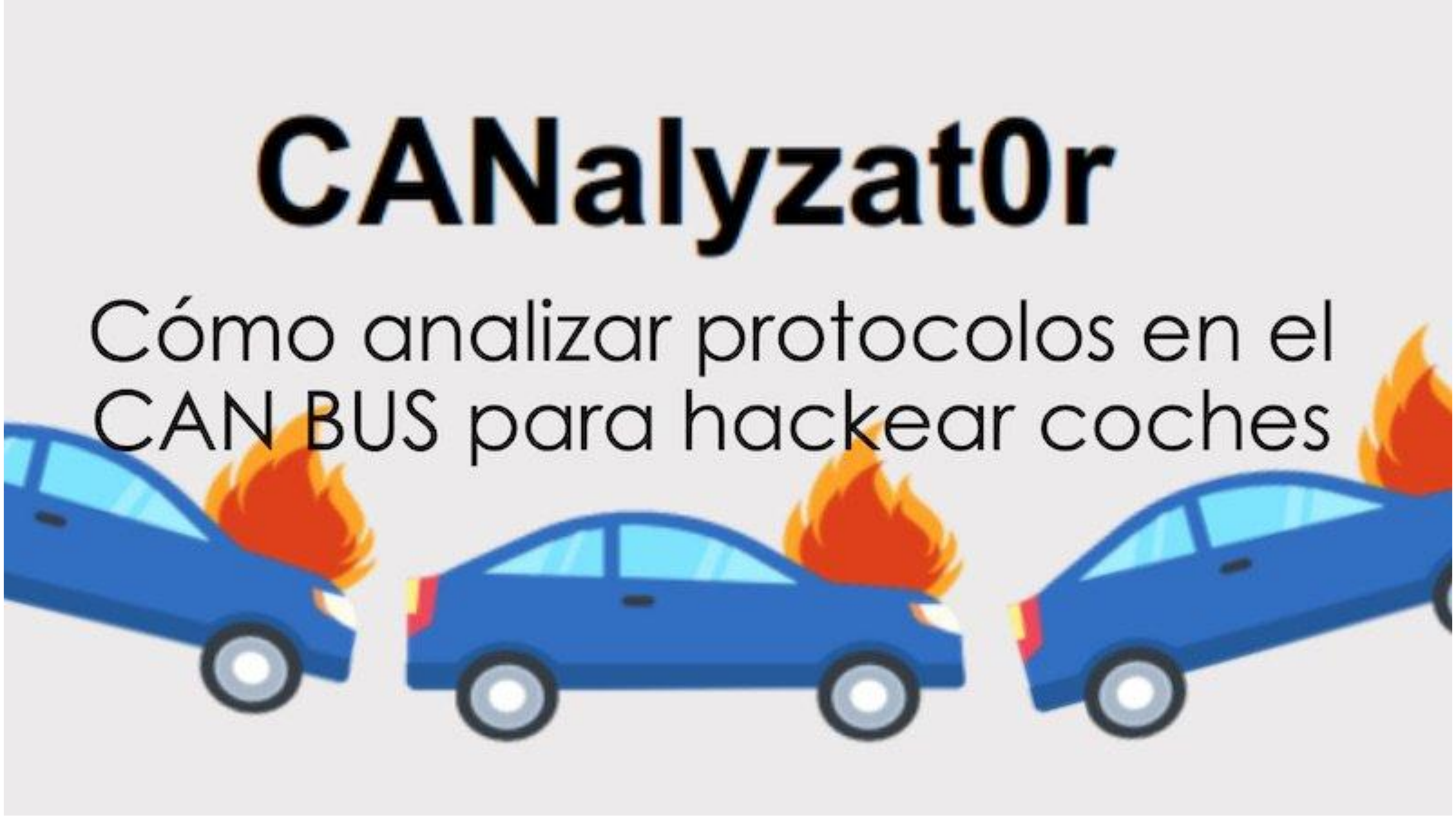
Active project: HackingStuff (201

Logging

Min. Loglevel: INFO

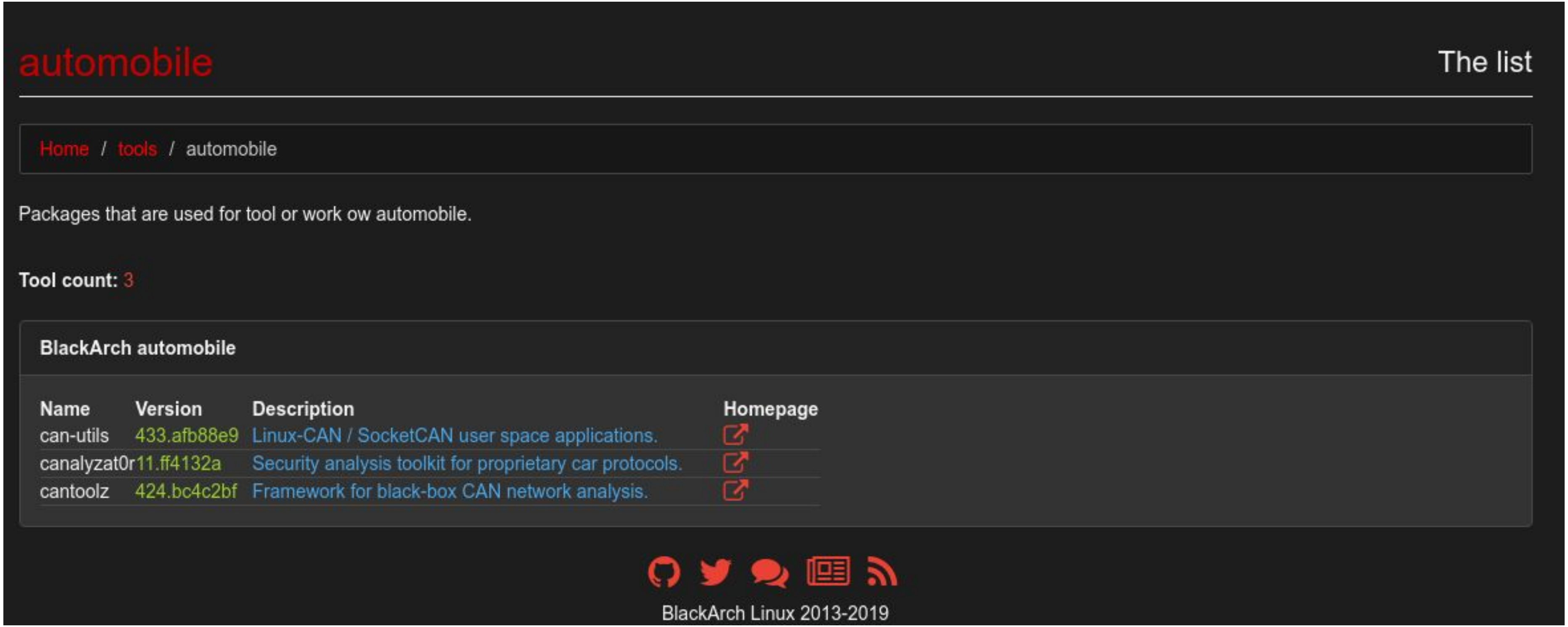
```
INFO: Database.py: CANalyzat0r.Database: connect: 366: Database connection OK
INFO: CANData.py: CANalyzat0r.CANData: checkVCAN: 134: Detected virtual interface for: vcan0
INFO: CANData.py: CANalyzat0r.CANData: rebuildCANDataInstances: 379: New CAN interface added: vcan0
INFO: CANData.py: CANalyzat0r.CANData: checkVCAN: 134: Detected virtual interface for: vcan1
INFO: CANData.py: CANalyzat0r.CANData: rebuildCANDataInstances: 379: New CAN interface added: vcan1
INFO: MainTab.py: CANalyzat0r.MainTab: applyGlobalInterfaceSettings: 407: CAN configuration updated
INFO: SnifferTabElement.py: CANalyzat0r.SnifferTabElement (vcan0): toggleSniffing: 160: Started sniffing
INFO: SenderTabElement.py: CANalyzat0r.SenderTab (Sender 1): sendAll: 145: Packets sent successfully
INFO: SenderTabElement.py: CANalyzat0r.SenderTab (Sender 1): sendAll: 119: Started sender thread
INFO: SenderTabElement.py: CANalyzat0r.SenderTab (sender2): sendAll: 119: Started sender thread
INFO: FuzzerTab.py: CANalyzat0r.FuzzerTab: validateDataMaskInput: 365: Extended data mask to: XXXXXXXXXXXXXXXX
INFO: FuzzerTab.py: CANalyzat0r.FuzzerTab: toggleFuzzing: 184: Started fuzzer thread
INFO: Database.py: CANalyzat0r.Database: saveProject: 663: Project saved
INFO: MainTab.py: CANalyzat0r.MainTab: setProject: 207: Loading project data...
INFO: MainTab.py: CANalyzat0r.MainTab: setProject: 216: Switched project to HackingStuff
```

Fuzzing, Sending (2 Threads), Sniffing (1 Thread) | Global interface: vcan0 | Project: HackingStuff



CANalyzat0r

Cómo analizar protocolos en el CAN BUS para hackear coches



automobile

The list

Home / tools / automobile

Packages that are used for tool or work ow automobile.

Tool count: 3

BlackArch automobile

Name	Version	Description	Homepage
can-utils	433.afb88e9	Linux-CAN / SocketCAN user space applications.	↗
canalyzat0r	11.ff4132a	Security analysis toolkit for proprietary car protocols.	↗
cantoolz	424.bc4c2bf	Framework for black-box CAN network analysis.	↗

BlackArch Linux 2013-2019

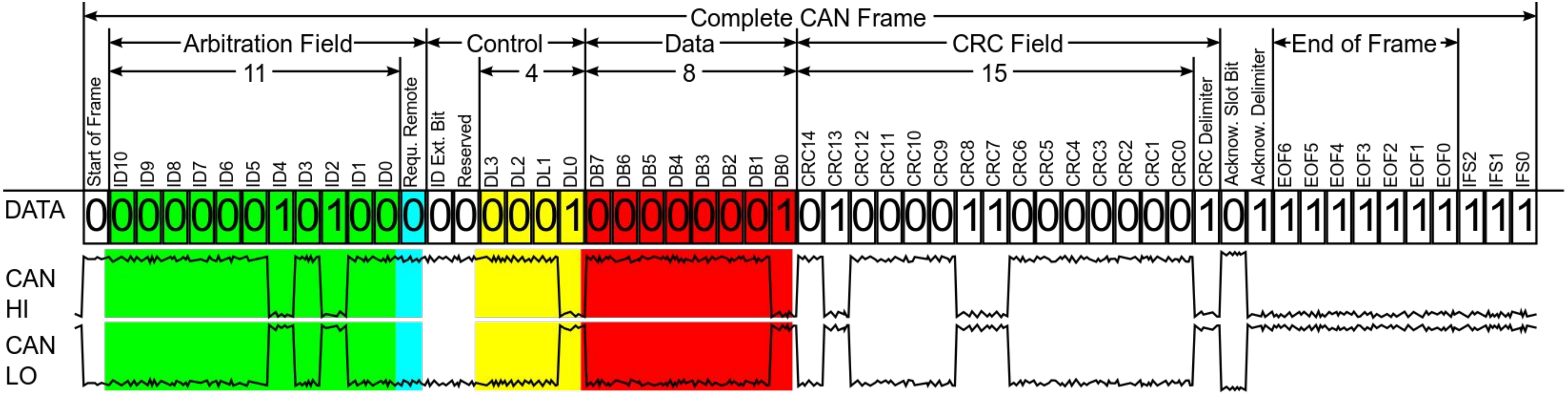
Goal

Share details regarding *CANalyzat0r*

Share knowledge & methodology

Present analysis results

Automotive Networks



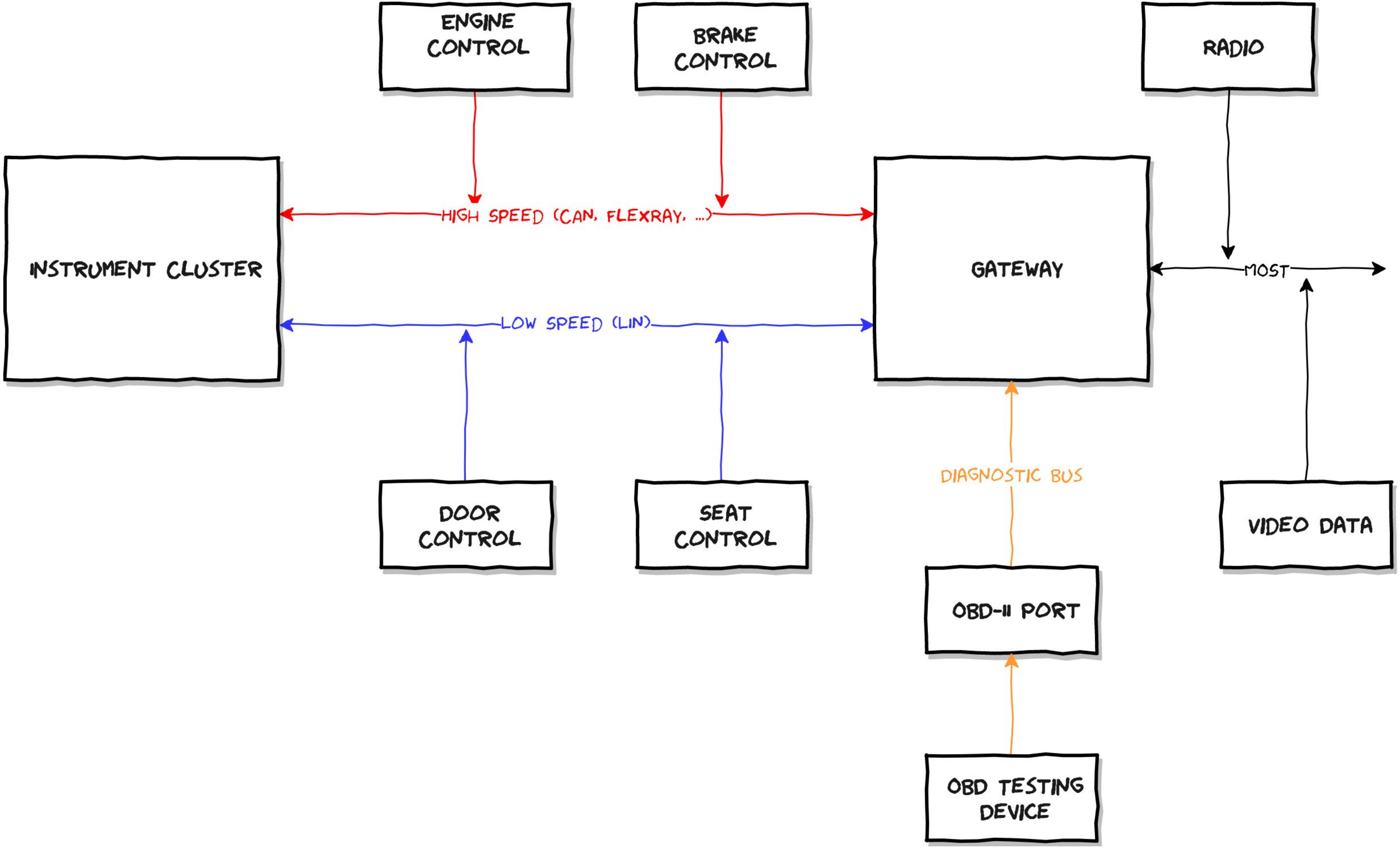
	Start of Frame	ID10	ID9	ID8	ID7	ID6	ID5	ID4	ID3	ID2	ID1	ID0	Requ. Remote	ID Ext. Bit	Reserved
DATA	0	0	0	0	0	0	0	1	0	1	0	0	0	0	0

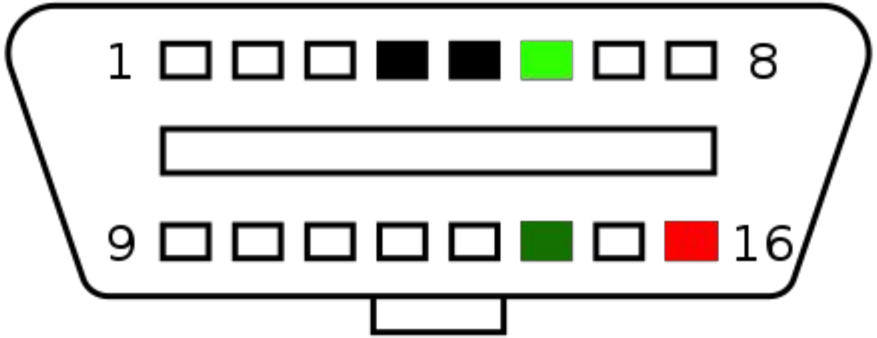
	DL3	DL2	DL1	DL0	DB7	DB6	DB5	DB4	DB3	DB2	DB1	DB0
DATA	0	0	0	1	0	0	0	0	0	0	0	1

	CRC14	CRC13	CRC12	CRC11	CRC10	CRC9	CRC8	CRC7	CRC6	CRC5	CRC4	CRC3	CRC2	CRC1	CRC0
DATA	0	1	0	0	0	0	1	1	0	0	0	0	0	0	0

	CRC Delimiter	Acknow. Slot Bit	Acknow. Delimiter	EOF6	EOF5	EOF4	EOF3	EOF2	EOF1	EOF0	IFS2	IFS1	IFS0
DATA	1	0	1	1	1	1	1	1	1	1	1	1	1

[3]


ENGINE CONTROL
BRAKE CONTROL
RADIO
INSTRUMENT CLUSTER
HIGH SPEED (CAN, FLEXRAY, ...)
GATEWAY
MOST
LOW SPEED (LIN)
DIAGNOSTIC BUS
DOOR CONTROL
SEAT CONTROL
VIDEO DATA
OBD-II PORT
OBD TESTING DEVICE


1
8
9
16

[6]

Automotive Networking

- Gateway interconnects various busses
- IPv6, VLANs
- CAN <-> SecOC
  - Secure Onboard Communication
    - CANFD -> AUTOSAR
      - [...] “aims for resource-efficient and practicable authentication mechanisms” [8]
- Plain CAN:
  - Sniff
  - Replay
    - → Fake messages
  - Inject

Threats in Automotive Networking

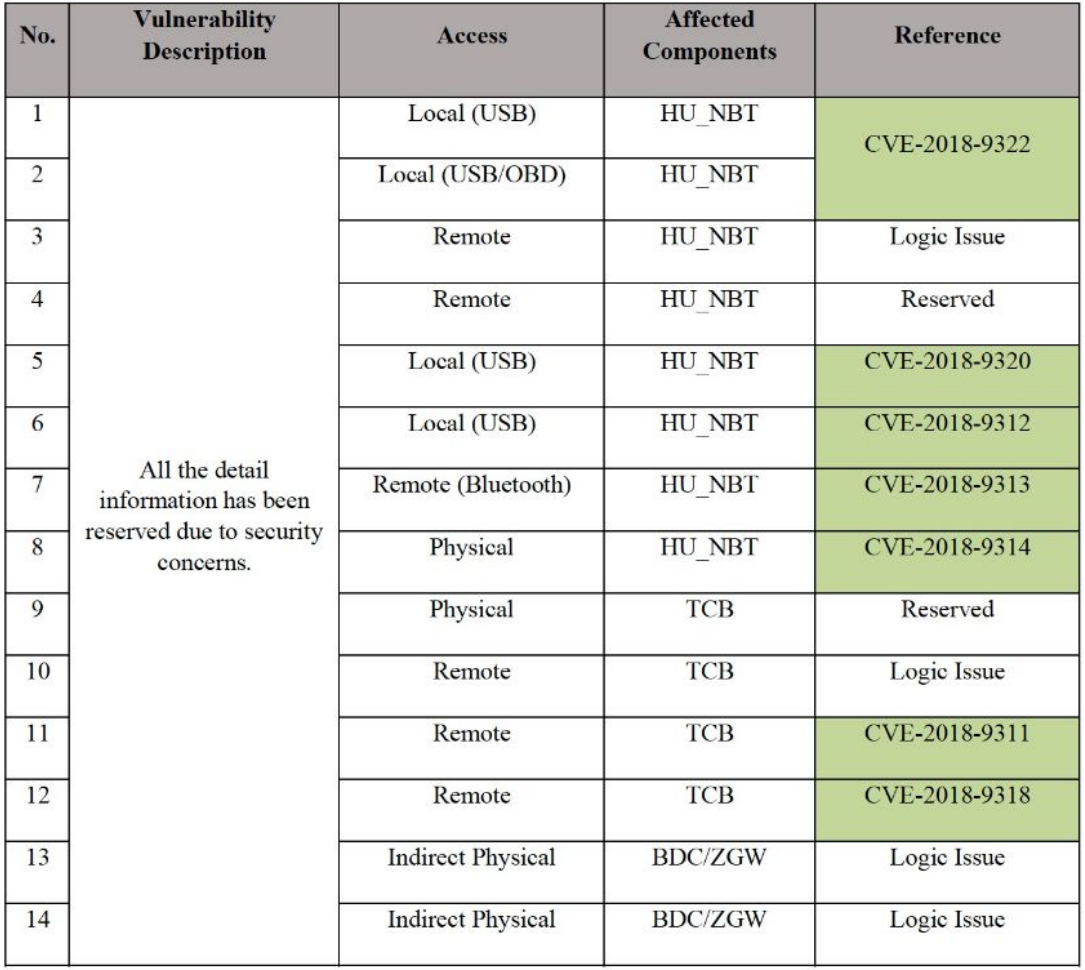
- Critical:
  - Attacker controls ECU
    - Is able to send arbitrary CAN messages
      - KeenLab BMW Analysis
  - Plain CAN: No authentication, encryption whatsoever

- MQTT brokers and services
  - Remote access to car network?

Threats in Automotive Networking

- Exposed services via USB ports
  - Attach network interface, set static IP
  - Scan/exploit/read/write
  - See: KeenLab BMW Paper

- Both remote and local attack surfaces
  - privilege escalation
  - hopping on other network nodes



No.	Vulnerability Description	Access	Affected Components	Reference
1	All the detail information has been reserved due to security concerns.	Local (USB)	HU_NBT	CVE-2018-9322
2		Local (USB/OBD)	HU_NBT	
3		Remote	HU_NBT	Logic Issue
4		Remote	HU_NBT	Reserved
5		Local (USB)	HU_NBT	CVE-2018-9320
6		Local (USB)	HU_NBT	CVE-2018-9312
7		Remote (Bluetooth)	HU_NBT	CVE-2018-9313
8		Physical	HU_NBT	CVE-2018-9314
9		Physical	TCB	Reserved
10		Remote	TCB	Logic Issue
11		Remote	TCB	CVE-2018-9311
12		Remote	TCB	CVE-2018-9318
13		Indirect Physical	BDC/ZGW	Logic Issue
14		Indirect Physical	BDC/ZGW	Logic Issue

[2 KeenLab]

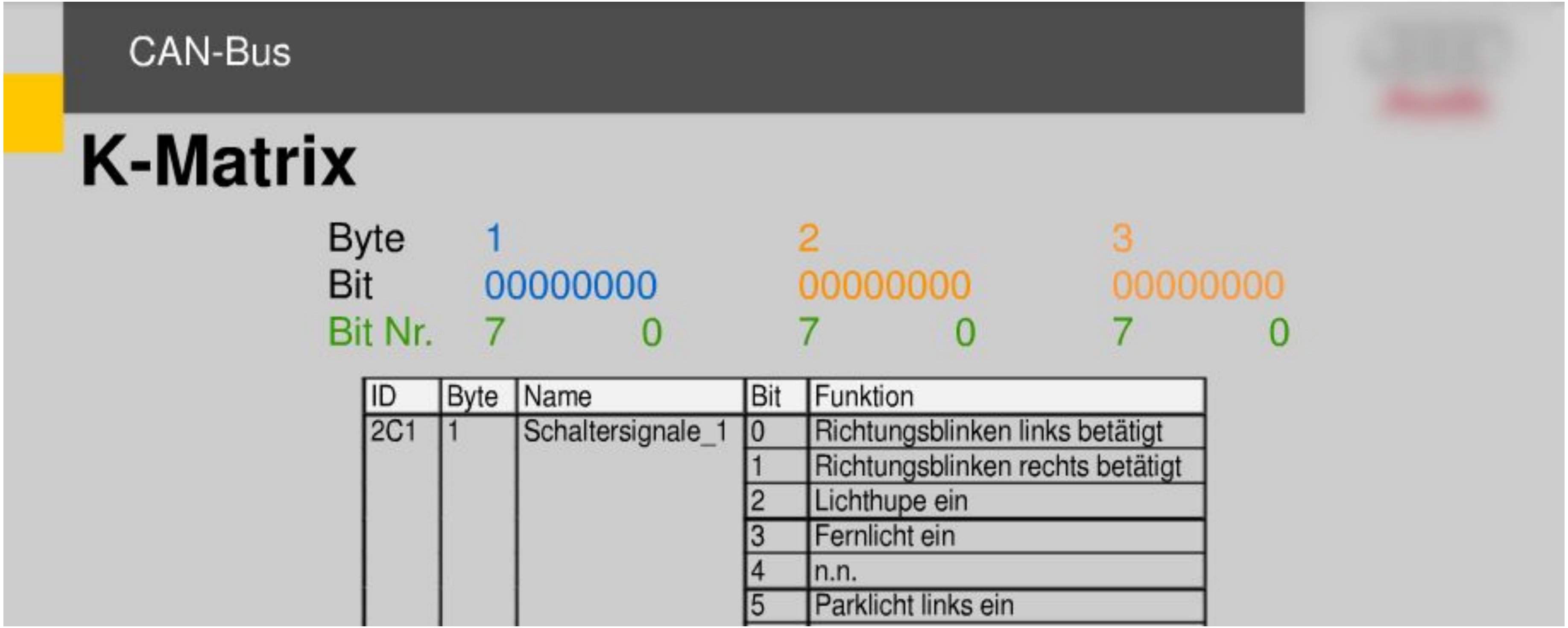
Table: Vulnerabilities and CVEs in Our Research Confirmed by BMW

Bus Analysis

- Connect to Bus
  - Twisted pair
  - Tap Wires
    - Reachable from exterior?
      - New fancy rear mirrors?
  - MiTM Devices
    - CANBadger: Remote access to car network

- Get K-Matrix / CAN Matrix
- Do fun stuff with the car
  - Control steering while driving
  - Kill services while driving and see what happens
  - Disable brakes

K-Matrix Example



CAN-Bus

K-Matrix

Byte	1		2		3	
Bit	00000000		00000000		00000000	
Bit Nr.	7	0	7	0	7	0

ID	Byte	Name	Bit	Funktion
2C1	1	Schaltersignale_1	0	Richtungsblinken links betätigt
			1	Richtungsblinken rechts betätigt
			2	Lichthupe ein
			3	Fernlicht ein
			4	n.n.
			5	Parklicht links ein

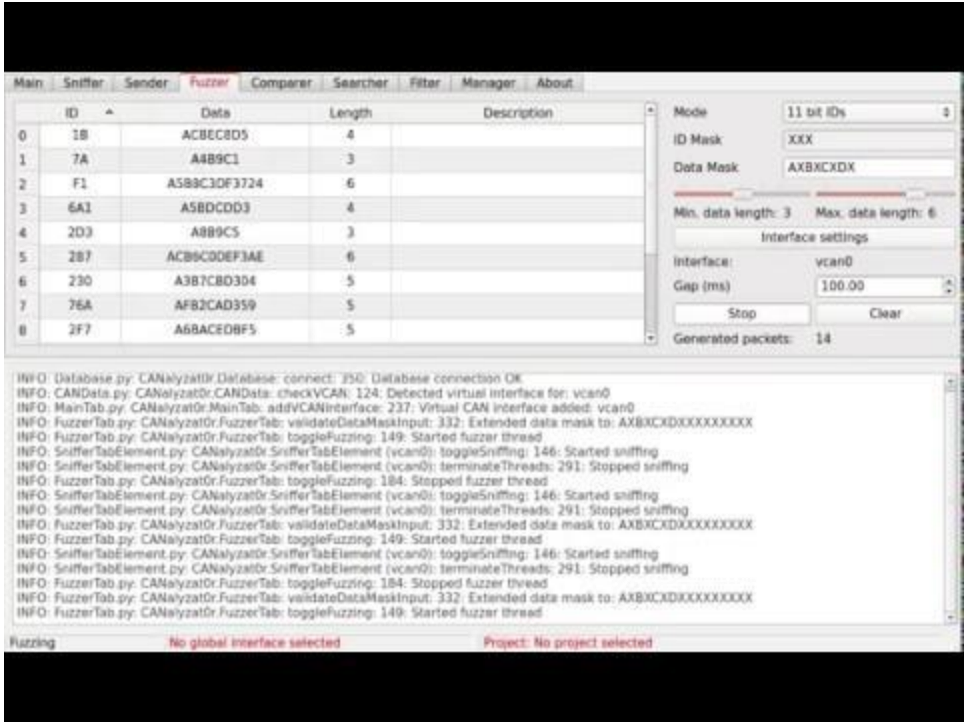
[4]

Analyzing CAN with the CANalyzatOr

Why CANalyzat0r ?

- Needed something as practical part :D
- I wanted to code
- After using various tools: Had new ideas
  - GUI (BOO!!!1!!elf)
  - Simplify common analysis tasks
  - Manage dumps, packets, findings and notes: SQLite/JSON
    - export -> Git
  - Multi interface support
  - Use in combination with can-utils
- sudo make run
- Surprisingly various people needed it too

Sniffing and Fuzzing



	ID	Data	Length	Description
0	1B	ACBEC8D5	4	
1	7A	A4B9C1	3	
2	F1	A5B3C3DF3724	6	
3	6A1	A5BDCDD3	4	
4	2D3	A8B9C5	3	
5	2B7	ACB9CDDEF3AE	6	
6	230	A3B7CBD304	5	
7	76A	AFB2CAD359	5	
8	2F7	A6BACEDBF5	5	

Mode: 11 bit IDs

ID Mask: XXX

Data Mask: AXBXCXDX

Min. data length: 3 Max. data length: 6

Interface: vcan0

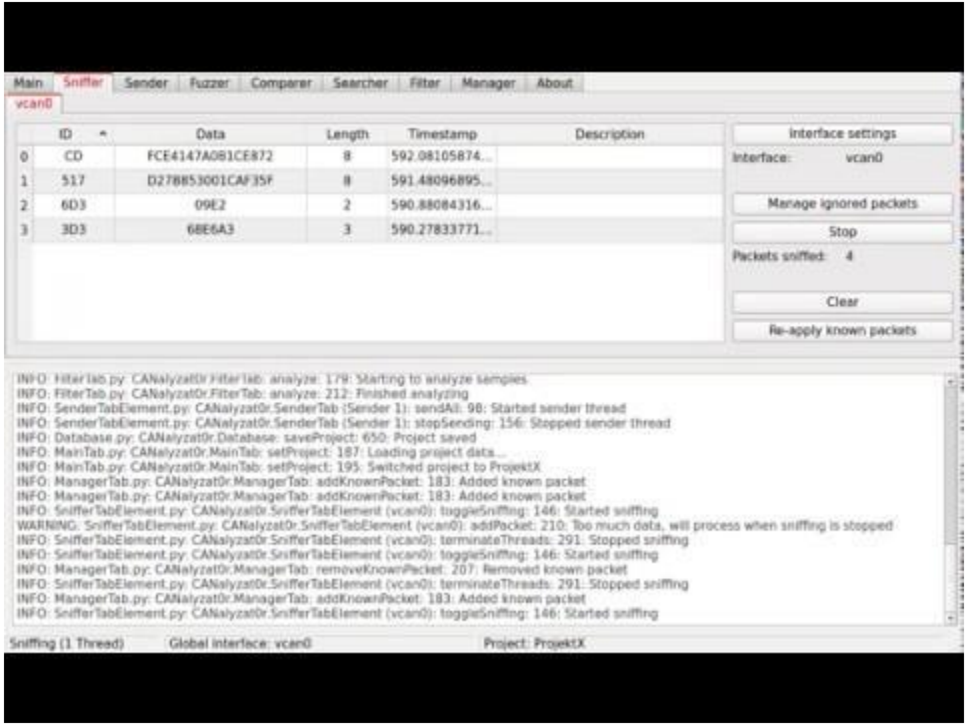
Gap (ms): 100.00

Generated packets: 14

```
INFO: Database.py: CANalyzat0r.Database: connect: 350: Database connection OK
INFO: CANData.py: CANalyzat0r.CANData: checkVCAN: 124: Detected virtual interface for: vcan0
INFO: MainTab.py: CANalyzat0r.MainTab: addVCANInterface: 237: Virtual CAN interface added: vcan0
INFO: FuzzerTab.py: CANalyzat0r.FuzzerTab: validateDataMaskInput: 332: Extended data mask to: AXBXCXDXXXXXXXXX
INFO: FuzzerTab.py: CANalyzat0r.FuzzerTab: toggleFuzzing: 149: Started fuzzer thread
INFO: SnifferTabElement.py: CANalyzat0r.SnifferTabElement (vcan0): toggleSniffing: 146: Started sniffing
INFO: SnifferTabElement.py: CANalyzat0r.SnifferTabElement (vcan0): terminateThreads: 291: Stopped sniffing
INFO: FuzzerTab.py: CANalyzat0r.FuzzerTab: toggleFuzzing: 184: Stopped fuzzer thread
INFO: SnifferTabElement.py: CANalyzat0r.SnifferTabElement (vcan0): toggleSniffing: 146: Started sniffing
INFO: SnifferTabElement.py: CANalyzat0r.SnifferTabElement (vcan0): terminateThreads: 291: Stopped sniffing
INFO: FuzzerTab.py: CANalyzat0r.FuzzerTab: validateDataMaskInput: 332: Extended data mask to: AXBXCXDXXXXXXXXX
INFO: FuzzerTab.py: CANalyzat0r.FuzzerTab: toggleFuzzing: 149: Started fuzzer thread
INFO: SnifferTabElement.py: CANalyzat0r.SnifferTabElement (vcan0): toggleSniffing: 146: Started sniffing
INFO: SnifferTabElement.py: CANalyzat0r.SnifferTabElement (vcan0): terminateThreads: 291: Stopped sniffing
INFO: FuzzerTab.py: CANalyzat0r.FuzzerTab: toggleFuzzing: 184: Stopped fuzzer thread
INFO: FuzzerTab.py: CANalyzat0r.FuzzerTab: validateDataMaskInput: 332: Extended data mask to: AXBXCXDXXXXXXXXX
INFO: FuzzerTab.py: CANalyzat0r.FuzzerTab: toggleFuzzing: 149: Started fuzzer thread
```

Fuzzing No global interface selected Project: No project selected

Managing and Recognizing Known Packets



	ID	Data	Length	Timestamp	Description
0	CD	FCE4147A0B1CE872	8	592.08105874...	
1	517	D278853001CAF35F	8	591.48096895...	
2	6D3	09E2	2	590.88084316...	
3	3D3	68E6A3	3	590.27833771...	

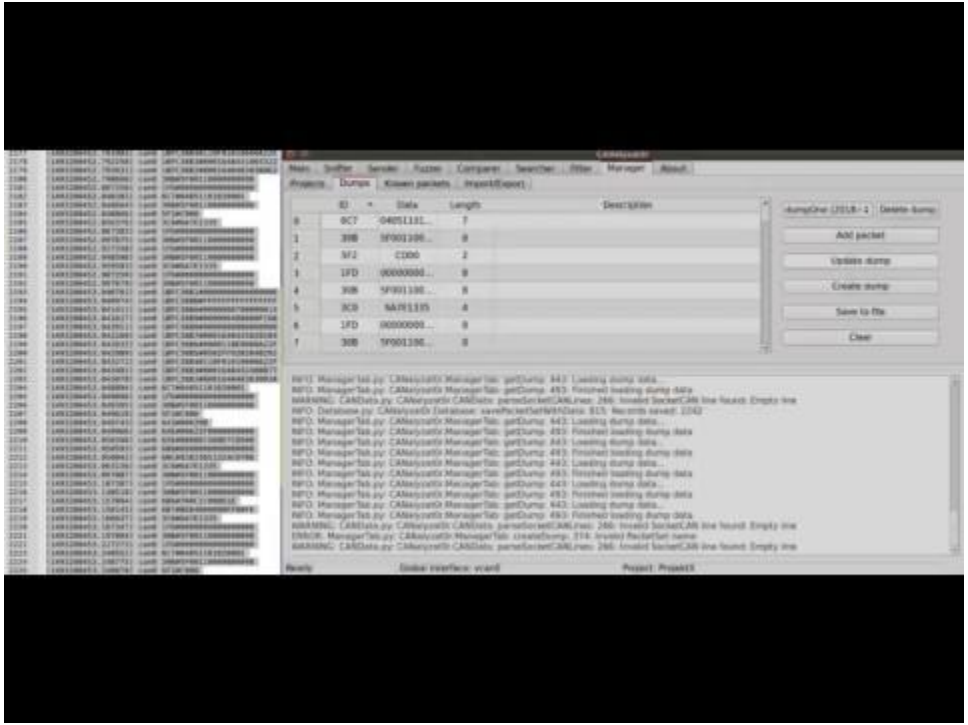
Interface: vcan0

Packets sniffed: 4

```
INFO: FilterTab.py: CANalyzat0r.FilterTab: analyze: 179: Starting to analyze samples
INFO: FilterTab.py: CANalyzat0r.FilterTab: analyze: 212: Finished analyzing
INFO: SenderTabElement.py: CANalyzat0r.SenderTab (Sender 1): sendAll: 98: Started sender thread
INFO: SenderTabElement.py: CANalyzat0r.SenderTab (Sender 1): stopSending: 156: Stopped sender thread
INFO: Database.py: CANalyzat0r.Database: saveProject: 650: Project saved
INFO: MainTab.py: CANalyzat0r.MainTab: setProject: 187: Loading project data...
INFO: MainTab.py: CANalyzat0r.MainTab: setProject: 195: Switched project to ProjektX
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: addKnownPacket: 183: Added known packet
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: addKnownPacket: 183: Added known packet
INFO: SnifferTabElement.py: CANalyzat0r.SnifferTabElement (vcan0): toggleSniffing: 146: Started sniffing
WARNING: SnifferTabElement.py: CANalyzat0r.SnifferTabElement (vcan0): addPacket: 210: Too much data, will process when sniffing is stopped
INFO: SnifferTabElement.py: CANalyzat0r.SnifferTabElement (vcan0): terminateThreads: 291: Stopped sniffing
INFO: SnifferTabElement.py: CANalyzat0r.SnifferTabElement (vcan0): toggleSniffing: 146: Started sniffing
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: removeKnownPacket: 207: Removed known packet
INFO: SnifferTabElement.py: CANalyzat0r.SnifferTabElement (vcan0): terminateThreads: 291: Stopped sniffing
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: addKnownPacket: 183: Added known packet
INFO: SnifferTabElement.py: CANalyzat0r.SnifferTabElement (vcan0): toggleSniffing: 146: Started sniffing
```

Sniffing (1 Thread)   Global interface: vcan0   Project: ProjektX

Combination with can-utils

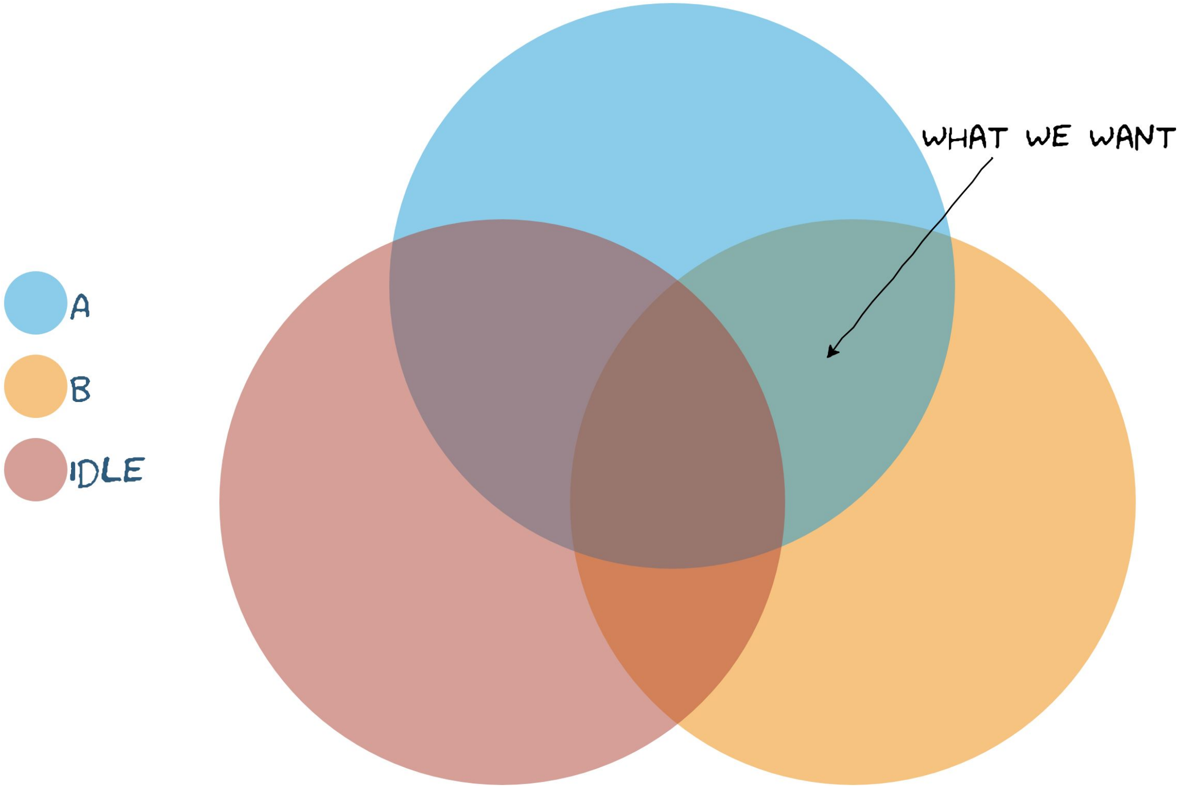


	ID	Data	Length	Description
0	0C7	04051131...	7	
1	30B	5F001100...	8	
2	5F2	C000	2	
3	1FD	00000000...	8	
4	30B	5F001100...	8	
5	3C0	9A7F1335	4	
6	1FD	00000000...	8	
7	30B	5F001100...	8	

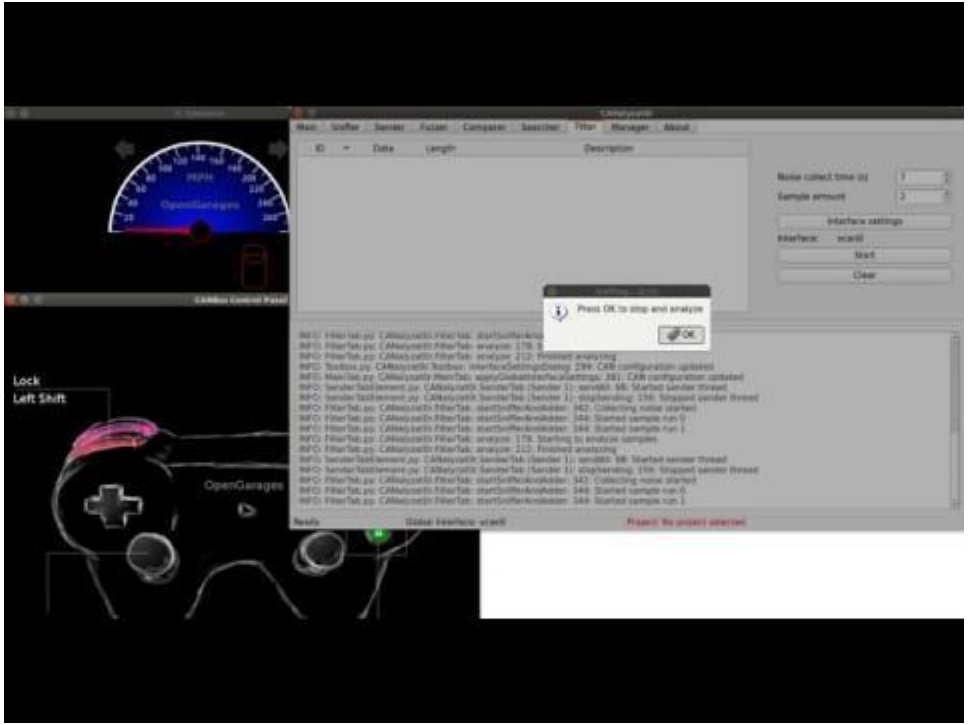
```
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: getDump: 443: Loading dump data...
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: getDump: 483: Finished loading dump data
WARNING: CANData.py: CANalyzat0r.CANData: parseSocketCANLines: 266: Invalid SocketCAN line found: Empty line
INFO: Database.py: CANalyzat0r.Database: savePacketSetWithData: 815: Records saved: 2242
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: getDump: 443: Loading dump data...
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: getDump: 483: Finished loading dump data
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: getDump: 443: Loading dump data...
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: getDump: 483: Finished loading dump data
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: getDump: 443: Loading dump data...
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: getDump: 483: Finished loading dump data
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: getDump: 443: Loading dump data...
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: getDump: 483: Finished loading dump data
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: getDump: 443: Loading dump data...
INFO: ManagerTab.py: CANalyzat0r.ManagerTab: getDump: 483: Finished loading dump data
WARNING: CANData.py: CANalyzat0r.CANData: parseSocketCANLines: 266: Invalid SocketCAN line found: Empty line
ERROR: ManagerTab.py: CANalyzat0r.ManagerTab: createDump: 374: Invalid PacketSet name
WARNING: CANData.py: CANalyzat0r.CANData: parseSocketCANLines: 266: Invalid SocketCAN line found: Empty line
```

Ready   Global interface: vcan0   Project: Projekt3

Automatic Packet Filtering



Automatic Packet Filtering



Noise collect time (s) 1

Sample amount 2

Interface: vcan0

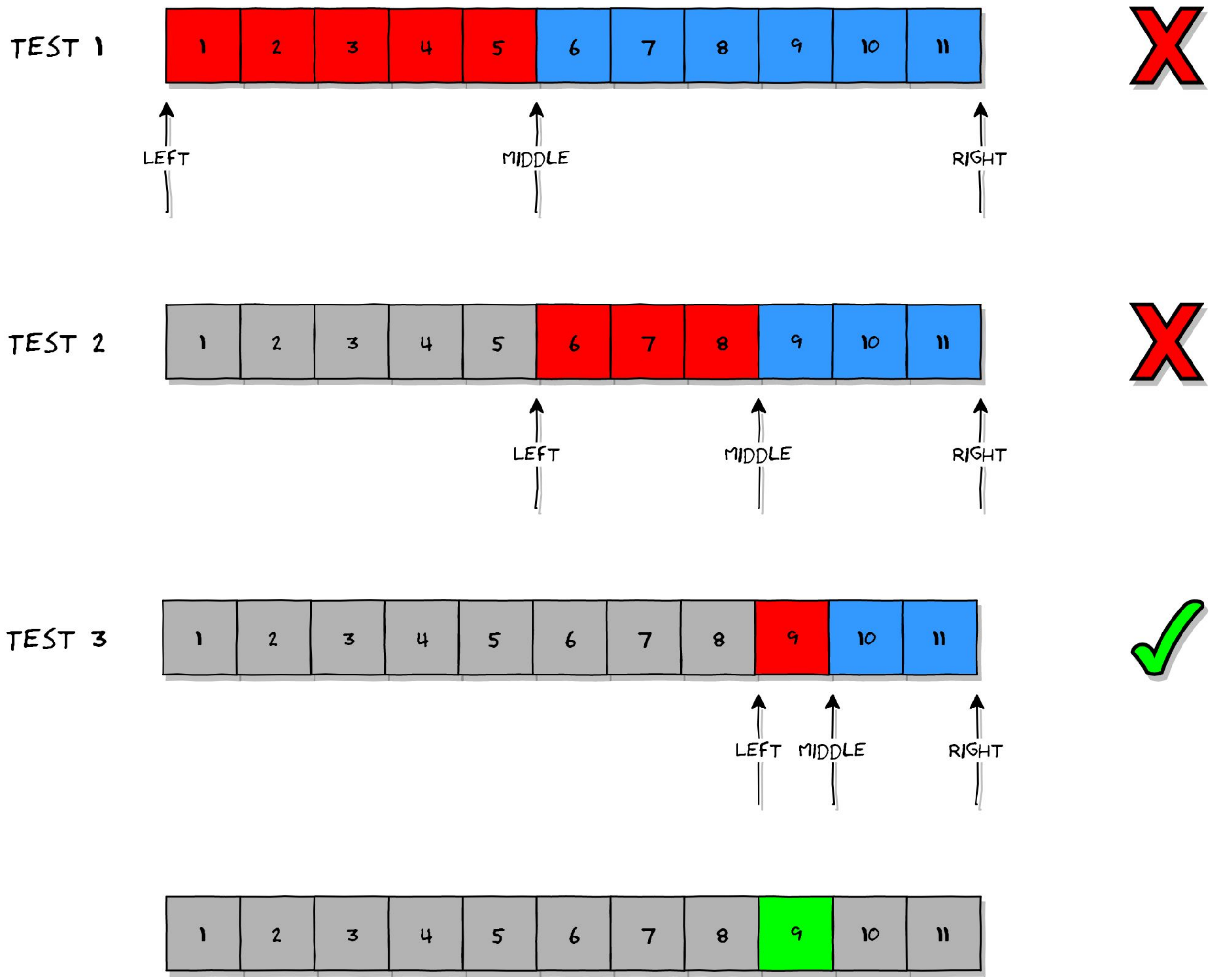
Press OK to stop and analyze

```
INFO: FilterTab.py: CANalyzat0r.FilterTab: startSnifferAnd
INFO: FilterTab.py: CANalyzat0r.FilterTab: analyze: 178: S
INFO: FilterTab.py: CANalyzat0r.FilterTab: analyze: 212: Finished analyzing
INFO: Toolbox.py: CANalyzat0r.Toolbox: interfaceSettingsDialog: 299: CAN configuration updated
INFO: MainTab.py: CANalyzat0r.MainTab: applyGlobalInterfaceSettings: 381: CAN configuration updated
INFO: SenderTabElement.py: CANalyzat0r.SenderTab (Sender 1): sendAll: 98: Started sender thread
INFO: SenderTabElement.py: CANalyzat0r.SenderTab (Sender 1): stopSending: 156: Stopped sender thread
INFO: FilterTab.py: CANalyzat0r.FilterTab: startSnifferAndAdder: 342: Collecting noise started
INFO: FilterTab.py: CANalyzat0r.FilterTab: startSnifferAndAdder: 344: Started sample run 0
INFO: FilterTab.py: CANalyzat0r.FilterTab: startSnifferAndAdder: 344: Started sample run 1
INFO: FilterTab.py: CANalyzat0r.FilterTab: analyze: 178: Starting to analyze samples
INFO: FilterTab.py: CANalyzat0r.FilterTab: analyze: 212: Finished analyzing
INFO: SenderTabElement.py: CANalyzat0r.SenderTab (Sender 1): sendAll: 98: Started sender thread
INFO: SenderTabElement.py: CANalyzat0r.SenderTab (Sender 1): stopSending: 156: Stopped sender thread
INFO: FilterTab.py: CANalyzat0r.FilterTab: startSnifferAndAdder: 342: Collecting noise started
INFO: FilterTab.py: CANalyzat0r.FilterTab: startSnifferAndAdder: 344: Started sample run 0
INFO: FilterTab.py: CANalyzat0r.FilterTab: startSnifferAndAdder: 344: Started sample run 1
```

Ready   Global interface: vcan0   Project: No project selected

Assisted Packet Filtering

- Fuzz -> minimize -> verify -> repeat
  - Answer Yes/No

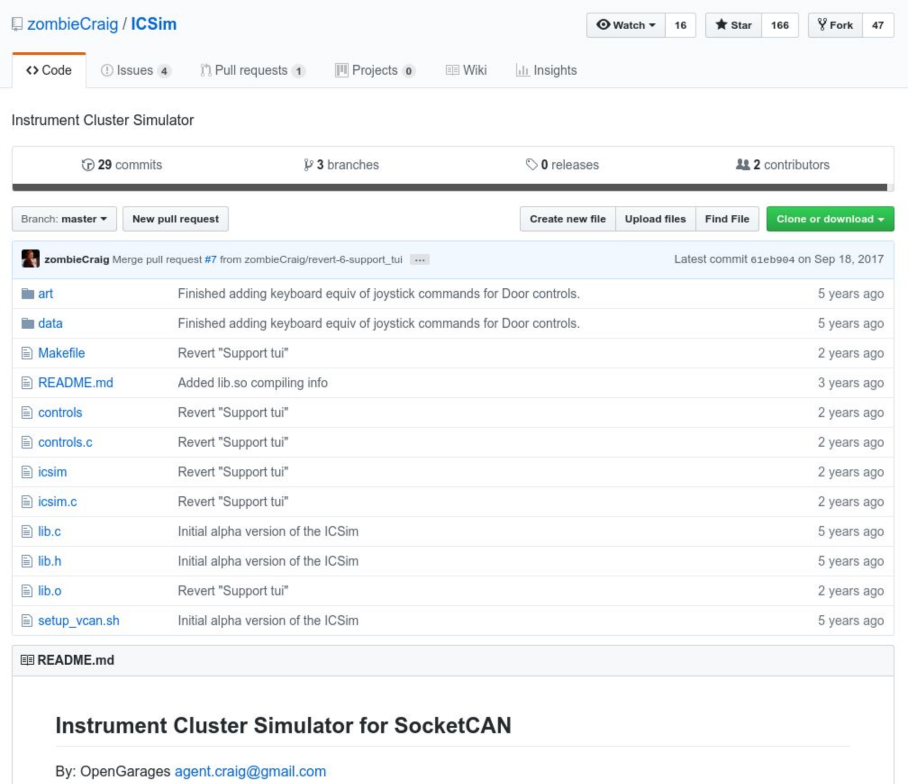

TEST 1
1 2 3 4 5 6 7 8 9 10 11
LEFT
MIDDLE
RIGHT
TEST 2
1 2 3 4 5 6 7 8 9 10 11
LEFT
MIDDLE
RIGHT
TEST 3
1 2 3 4 5 6 7 8 9 10 11
LEFT
MIDDLE
RIGHT
1 2 3 4 5 6 7 8 9 10 11

Build Your Own “Lab”

Required Steps

- Get hardware - instrument cluster (IC)
- Get wiring diagram
- Get ignition packet(s): Turn IC on
- → Do Stuff

OK Cool But I Don’t Want To Buy Stuff!!1!



Instrument Cluster Simulator

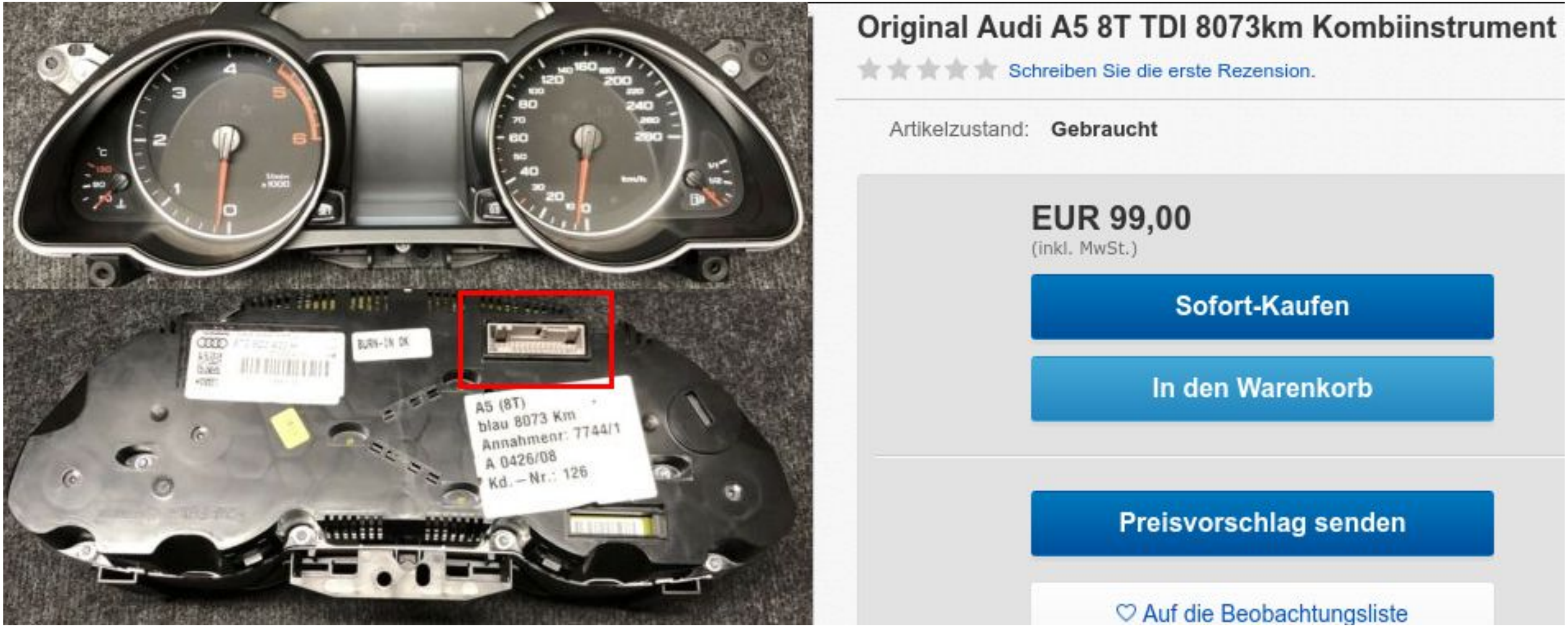
zombieCraig Merge pull request #7 from zombieCraig/revert-6-support_tui … Latest commit 61eb904 on Sep 18, 2017

art	Finished adding keyboard equiv of joystick commands for Door controls.	5 years ago
data	Finished adding keyboard equiv of joystick commands for Door controls.	5 years ago
Makefile	Revert "Support tui"	2 years ago
README.md	Added lib.so compiling info	3 years ago
controls	Revert "Support tui"	2 years ago
controls.c	Revert "Support tui"	2 years ago
icsim	Revert "Support tui"	2 years ago
icsim.c	Revert "Support tui"	2 years ago
lib.c	Initial alpha version of the ICSim	5 years ago
lib.h	Initial alpha version of the ICSim	5 years ago
lib.o	Revert "Support tui"	2 years ago
setup_vcan.sh	Initial alpha version of the ICSim	5 years ago

README.md

Instrument Cluster Simulator for SocketCAN

By: OpenGarages agent.craig@gmail.com



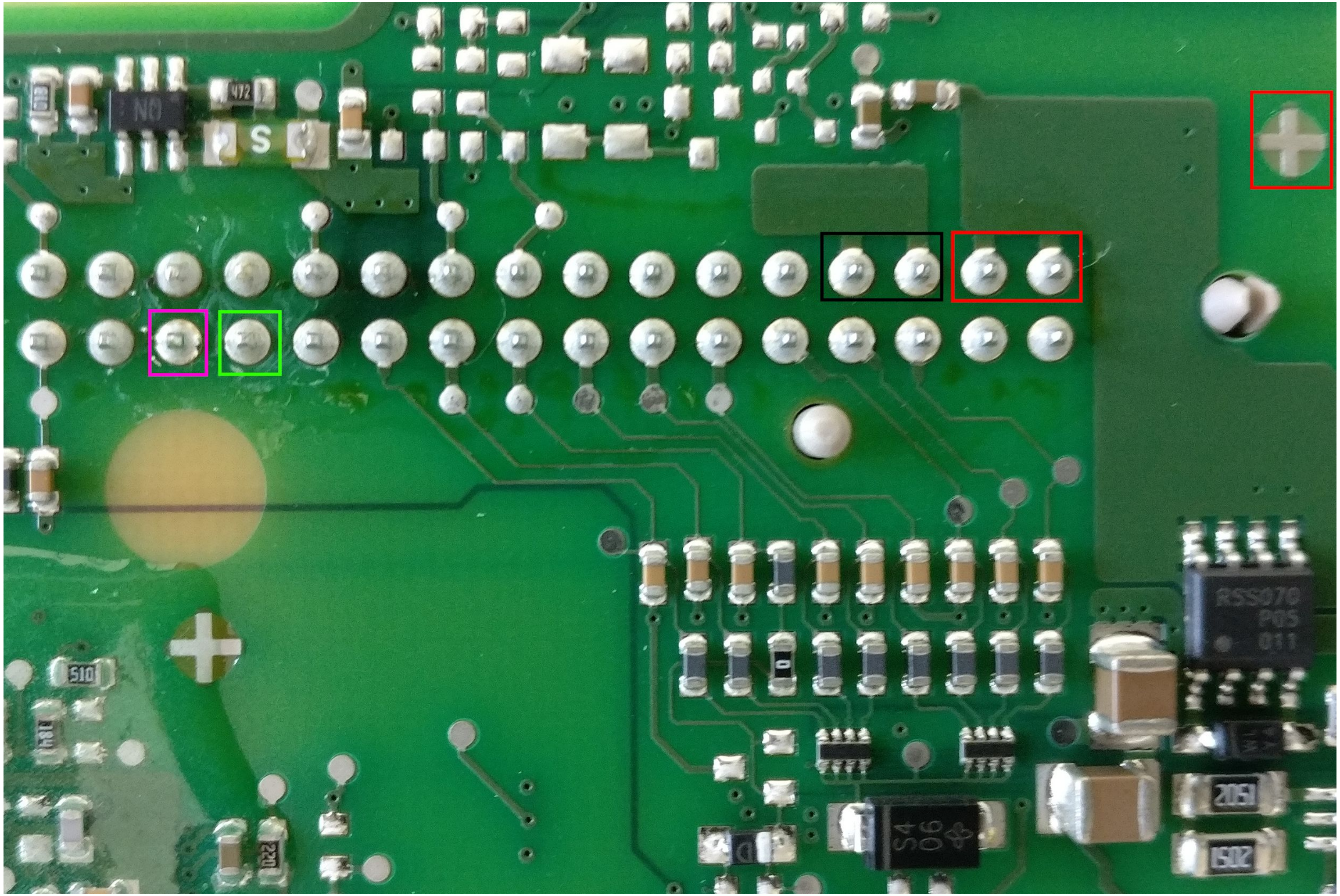
Original Audi A5 8T TDI 8073km Kombiinstrument

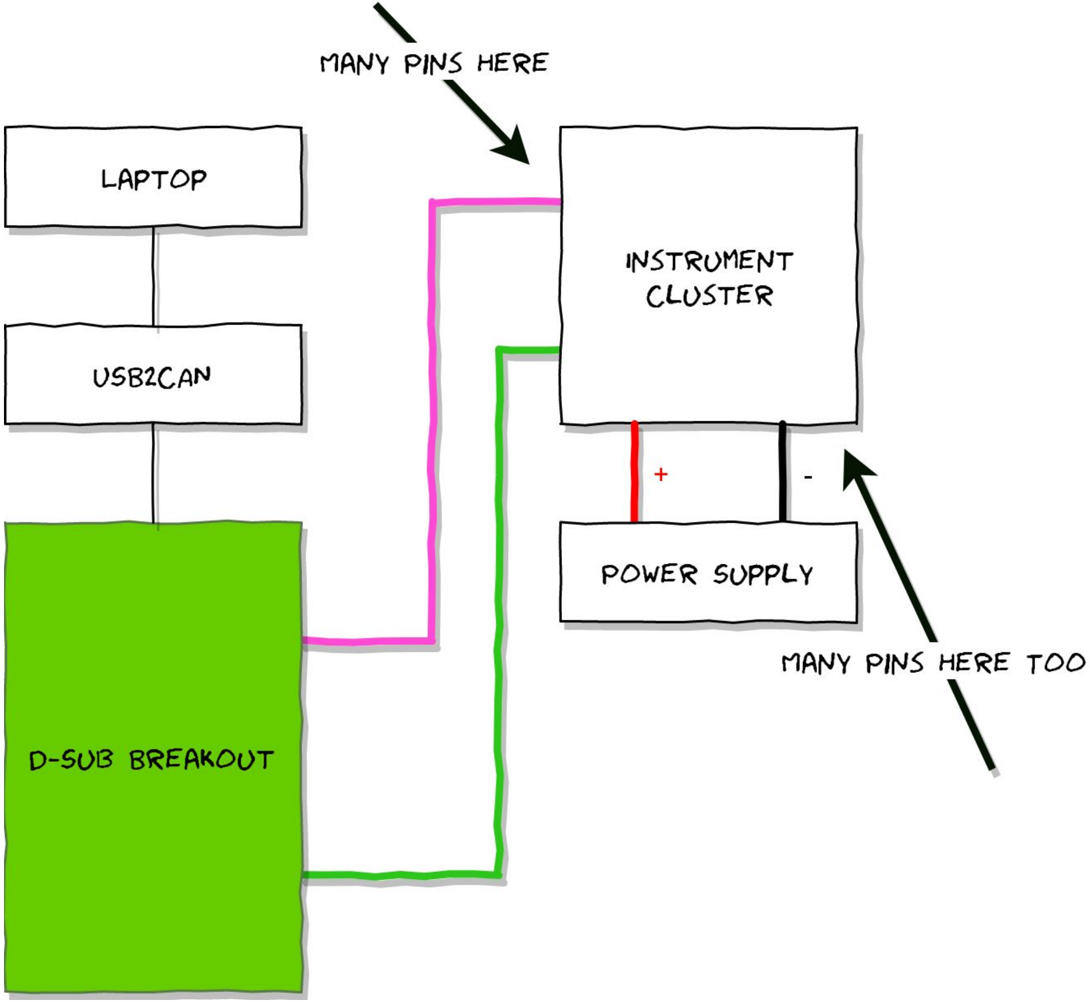
Schreiben Sie die erste Rezension.

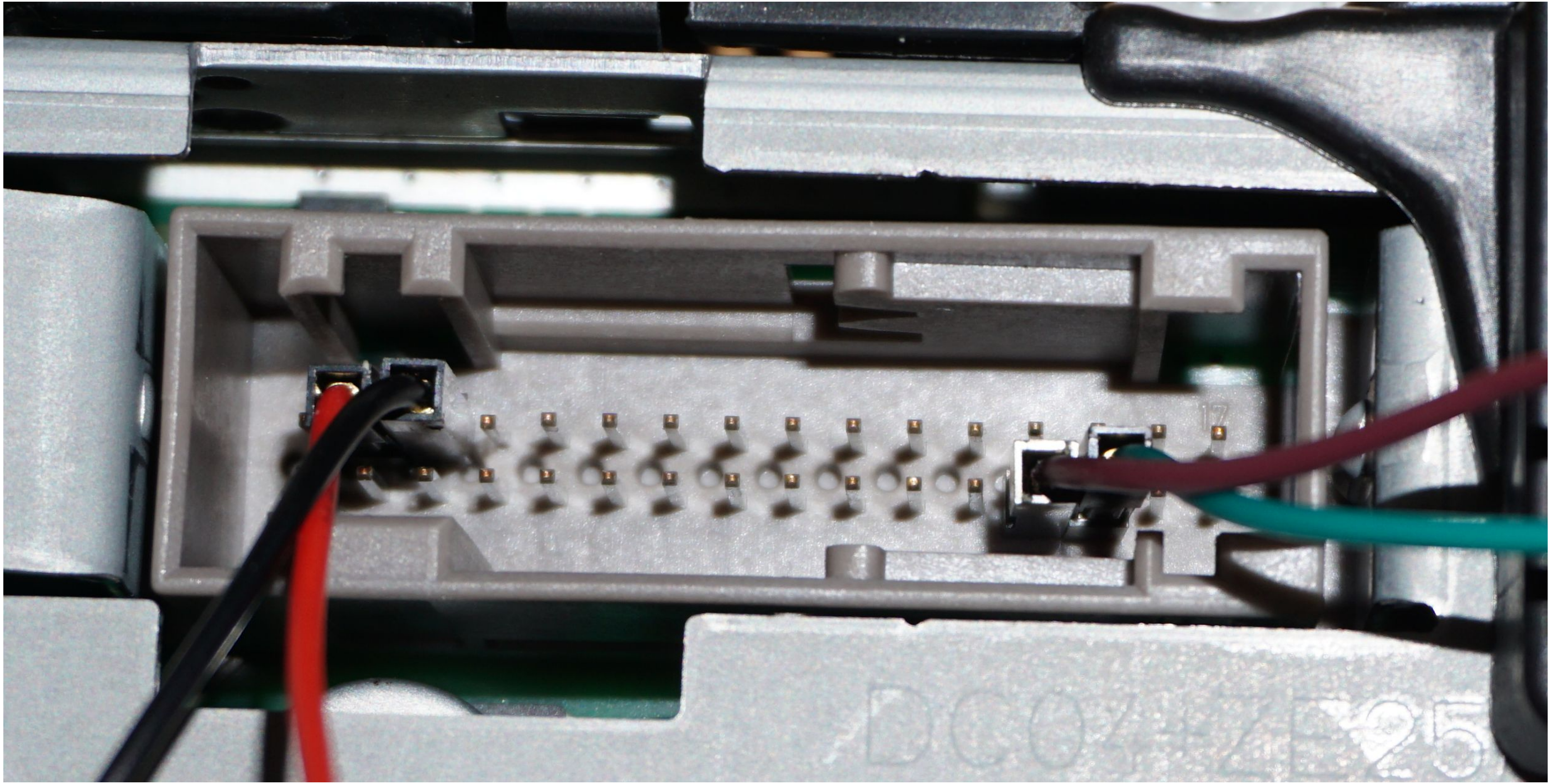
Artikelzustand: **Gebraucht**

EUR 99,00
(inkl. MwSt.)

A5 (8T)
blau 8073 Km
Annahmenr: 7744/1
A 0426/08
Kd.-Nr.: 126

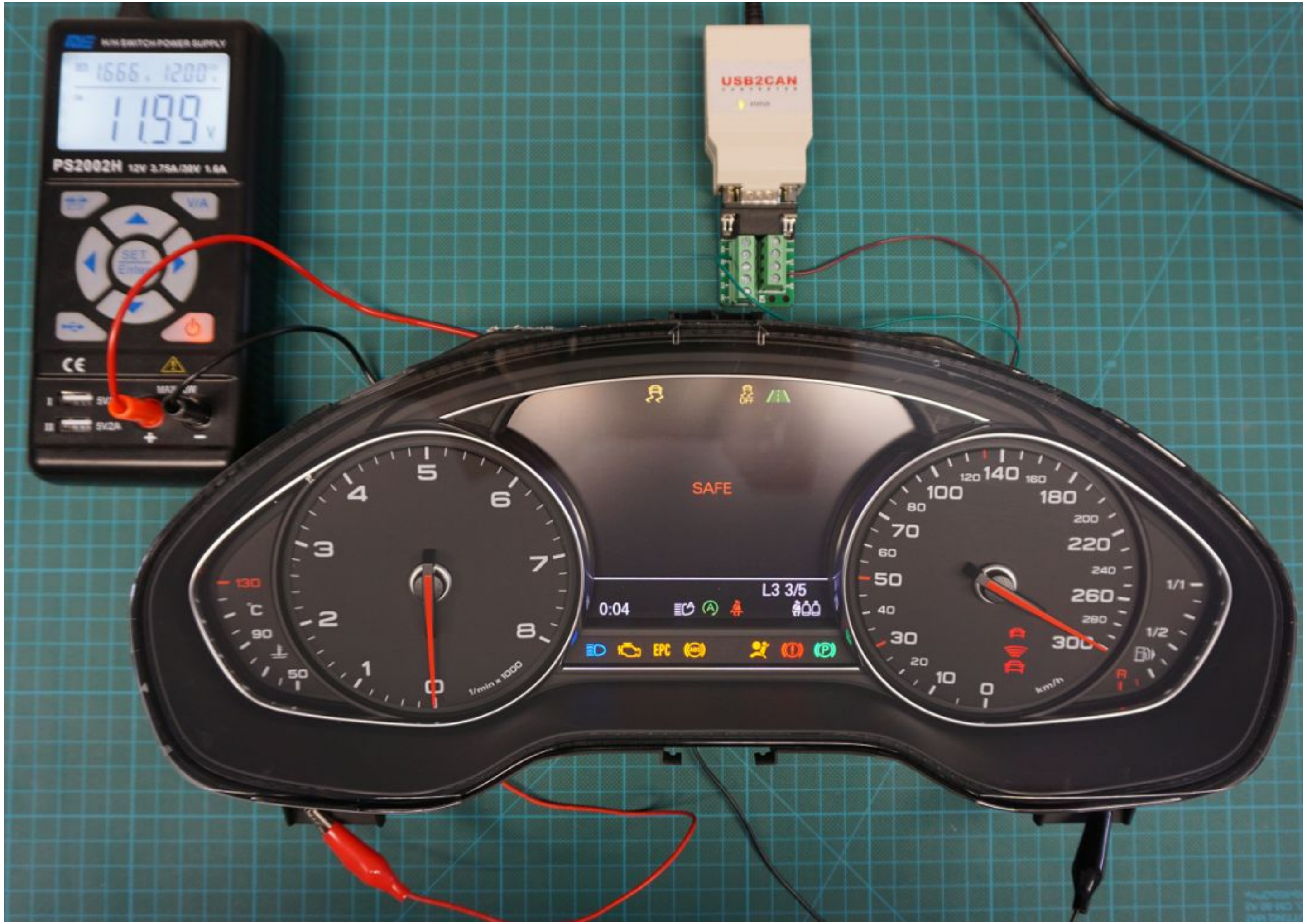

NO
472
S
510
RSS070
P05
011
0
S4
06
2051
1502

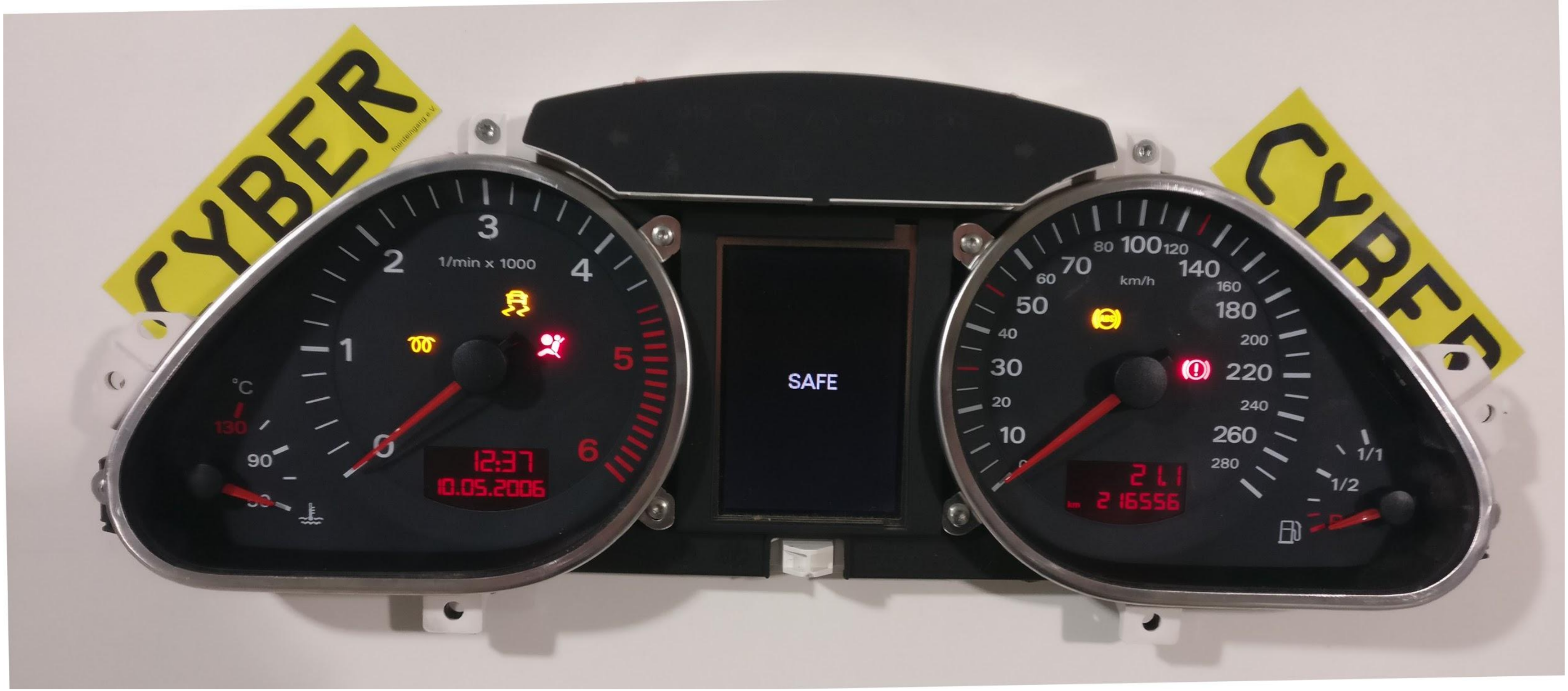

MANY PINS HERE
LAPTOP
INSTRUMENT CLUSTER
USB2CAN
+
-
POWER SUPPLY
MANY PINS HERE TOO
D-SUB BREAKOUT


17
DC04-ZE25

Ignition Packet(s)

1. Fuzz until it turns on
2. Assisted Packet Filtering
3. Once determined: Send in loop (also with ***CANalyzat0r***)
4. Proceed with analysis

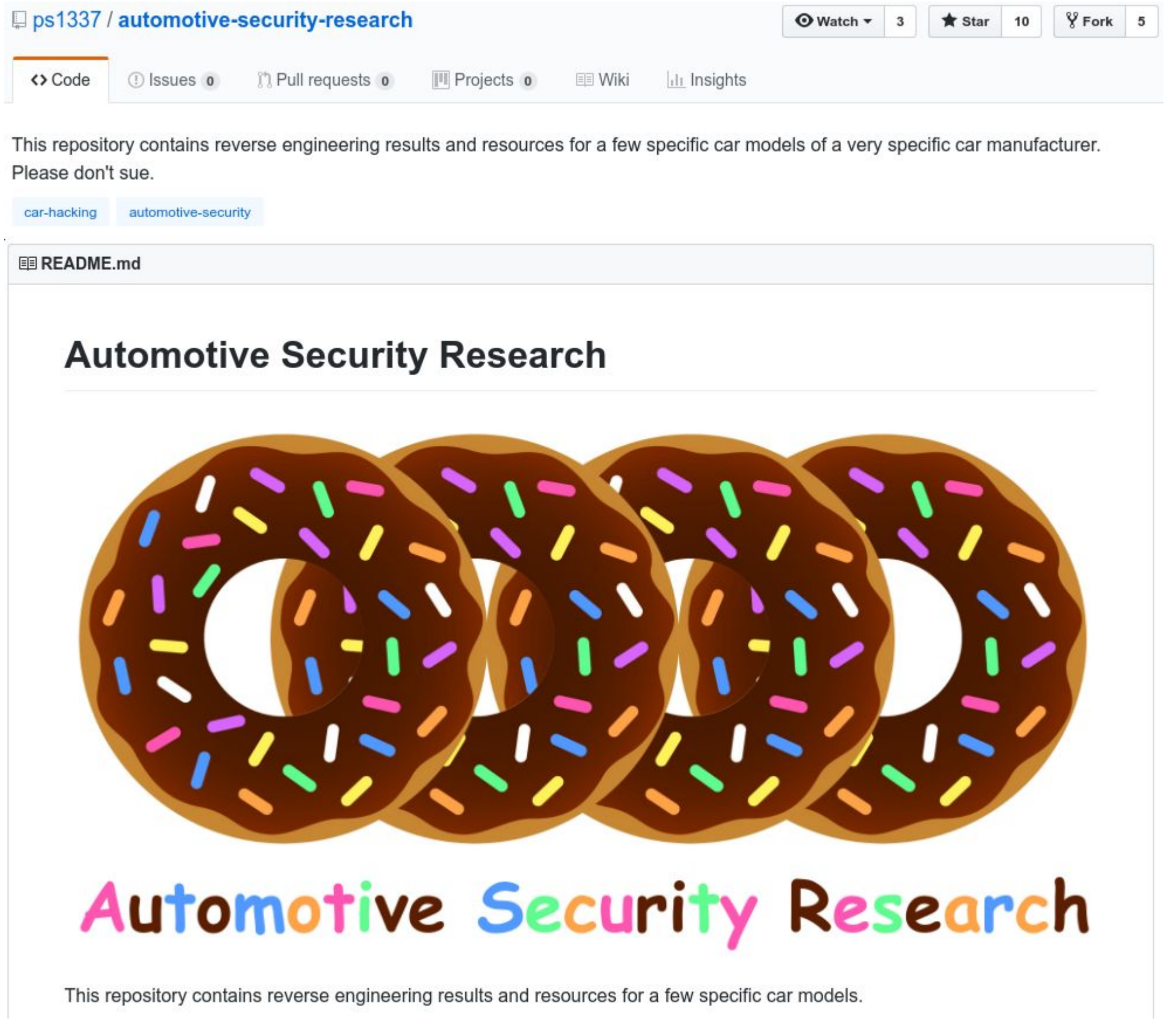

N/N SWITCH POWER SUPPLY
16.66
12.00
11.99 V
PS2002H 12V 3.75A/30V 1.6A
V/A
SET Enter
USB2CAN
status
SAFE
L3 3/5
0:04
EPC
1/min x 1000
km/h
130
90
50
°C
1/1
1/2
R


CYBER
CYBER
1/min x 1000
0
1
2
3
4
5
6
°C
130
90
50
12:37
10.05.2006
SAFE
km/h
0
10
20
30
40
50
60
70
80
100
120
140
160
180
200
220
240
260
280
211
km 216556
1/1
1/2
R

Some Tips

1. Don’t fuzz in ***YOLO*** mode
2. Make sure to get all ignition packets
3. Extending: Get more hardware

Analysis Results

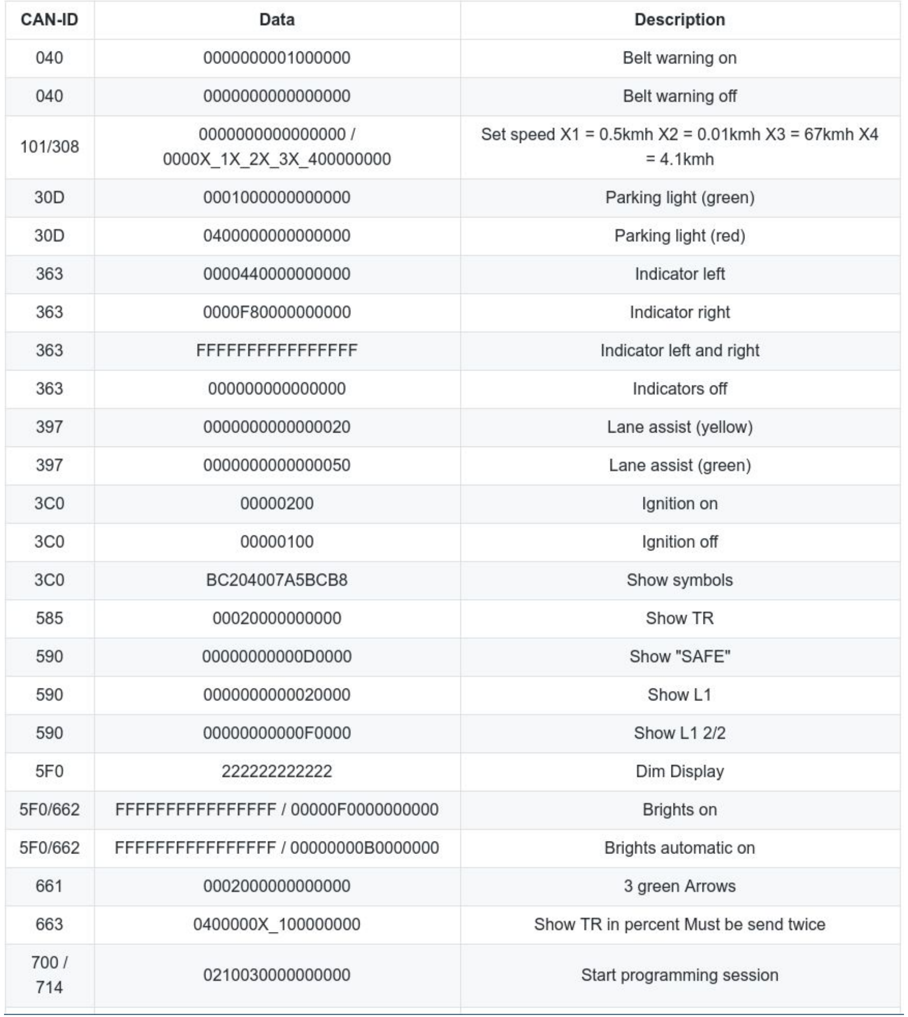


ps1337 / automotive-security-research

This repository contains reverse engineering results and resources for a few specific car models of a very specific car manufacturer. Please don't sue.

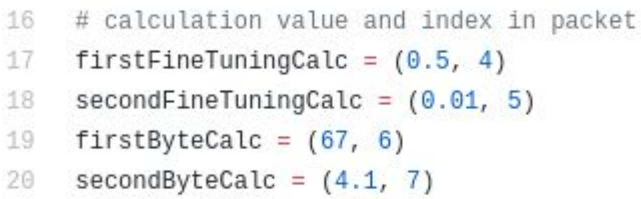
Automotive Security Research

This repository contains reverse engineering results and resources for a few specific car models.

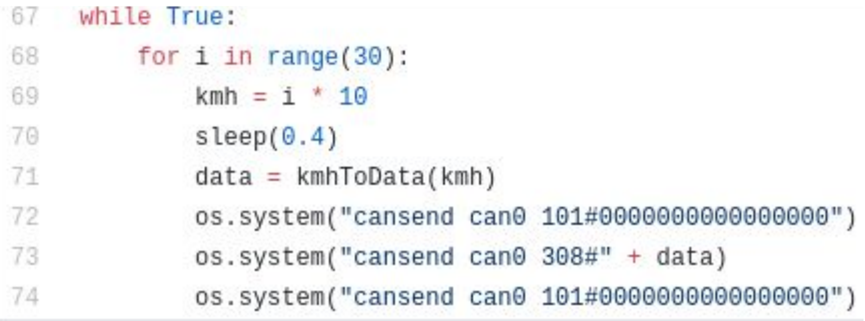


CAN-ID	Data	Description
040	0000000001000000	Belt warning on
040	0000000000000000	Belt warning off
101/308	0000000000000000 / 0000X_1X_2X_3X_400000000	Set speed X1 = 0.5kmh X2 = 0.01kmh X3 = 67kmh X4 = 4.1kmh
30D	0001000000000000	Parking light (green)
30D	0400000000000000	Parking light (red)
363	0000440000000000	Indicator left
363	0000F80000000000	Indicator right
363	FFFFFFFFFFFFFFFF	Indicator left and right
363	000000000000000	Indicators off
397	0000000000000020	Lane assist (yellow)
397	0000000000000050	Lane assist (green)
3C0	00000200	Ignition on
3C0	00000100	Ignition off
3C0	BC204007A5BCB8	Show symbols
585	00020000000000	Show TR
590	00000000000D0000	Show "SAFE"
590	0000000000020000	Show L1
590	00000000000F0000	Show L1 2/2
5F0	222222222222	Dim Display
5F0/662	FFFFFFFFFFFFFFFF / 00000F0000000000	Brights on
5F0/662	FFFFFFFFFFFFFFFF / 00000000B0000000	Brights automatic on
661	0002000000000000	3 green Arrows
663	0400000X_100000000	Show TR in percent Must be send twice
700 / 714	0210030000000000	Start programming session

Setting Arbitrary Speed Values



```python
# calculation value and index in packet
firstFineTuningCalc = (0.5, 4)
secondFineTuningCalc = (0.01, 5)
firstByteCalc = (67, 6)
secondByteCalc = (4.1, 7)
```



```python
while True:
    for i in range(30):
        kmh = i * 10
        sleep(0.4)
        data = kmhToData(kmh)
        os.system("cansend can0 101#0000000000000000")
        os.system("cansend can0 308#" + data)
        os.system("cansend can0 101#0000000000000000")
```

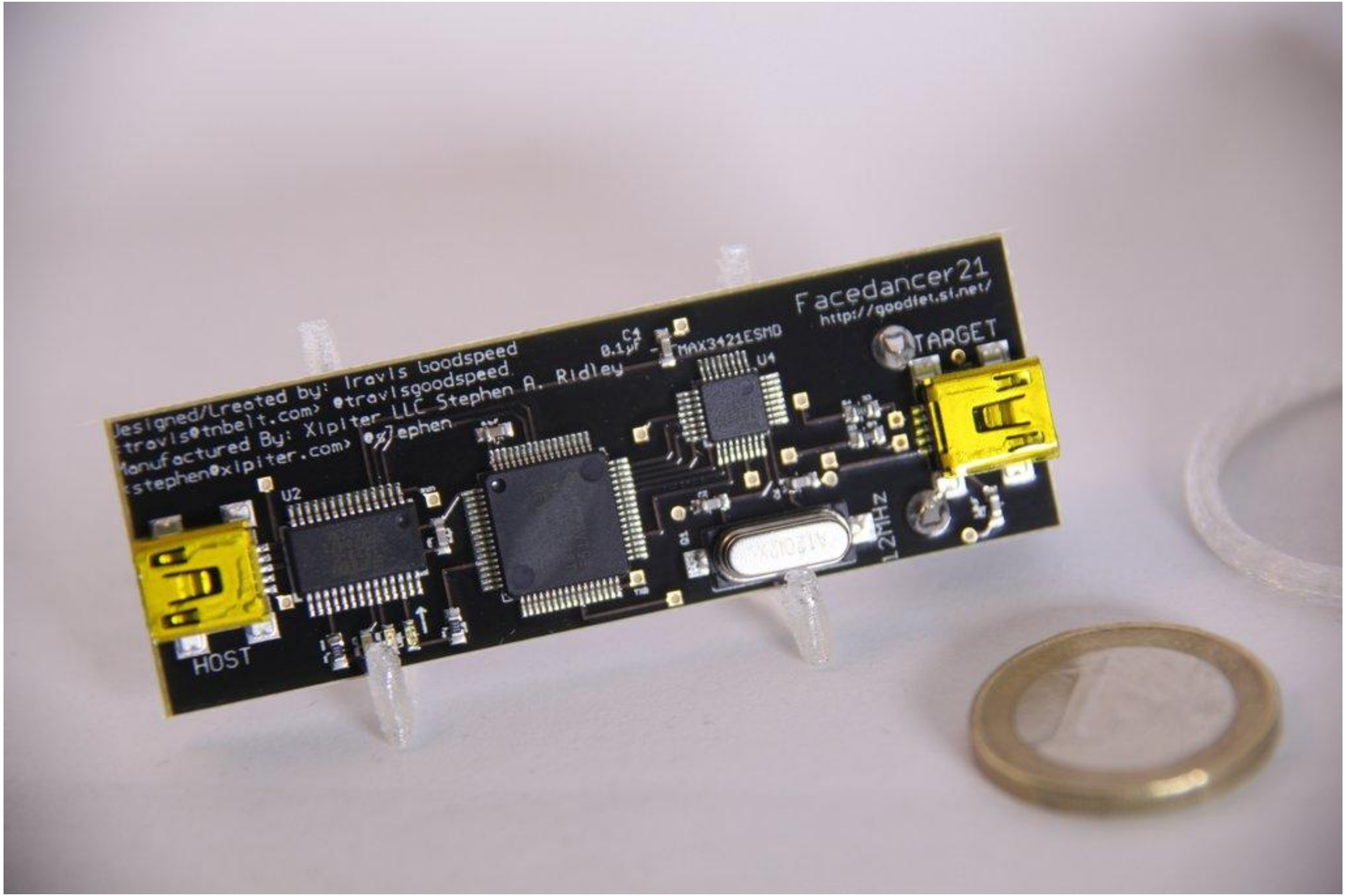
Fuzzing

- Media parsers
  - exotic file formats
  - vCard

  → radamsa

- Open ports: also radamsa
  - Also: Local PrivEsc possible?

- Specialized tools for interfaces
  - USB: Facedancer


Facedancer21
http://goodfet.sf.net/
Designed/Created by: Travis Goodspeed
<travis@tnbelt.com> @travisgoodspeed
Manufactured By: Xipiter LLC Stephen A. Ridley
<stephen@xipiter.com> @s7ephen
C4
0.1µF
MAX3421ESMD
U4
U2
TARGET
HOST
12MHz

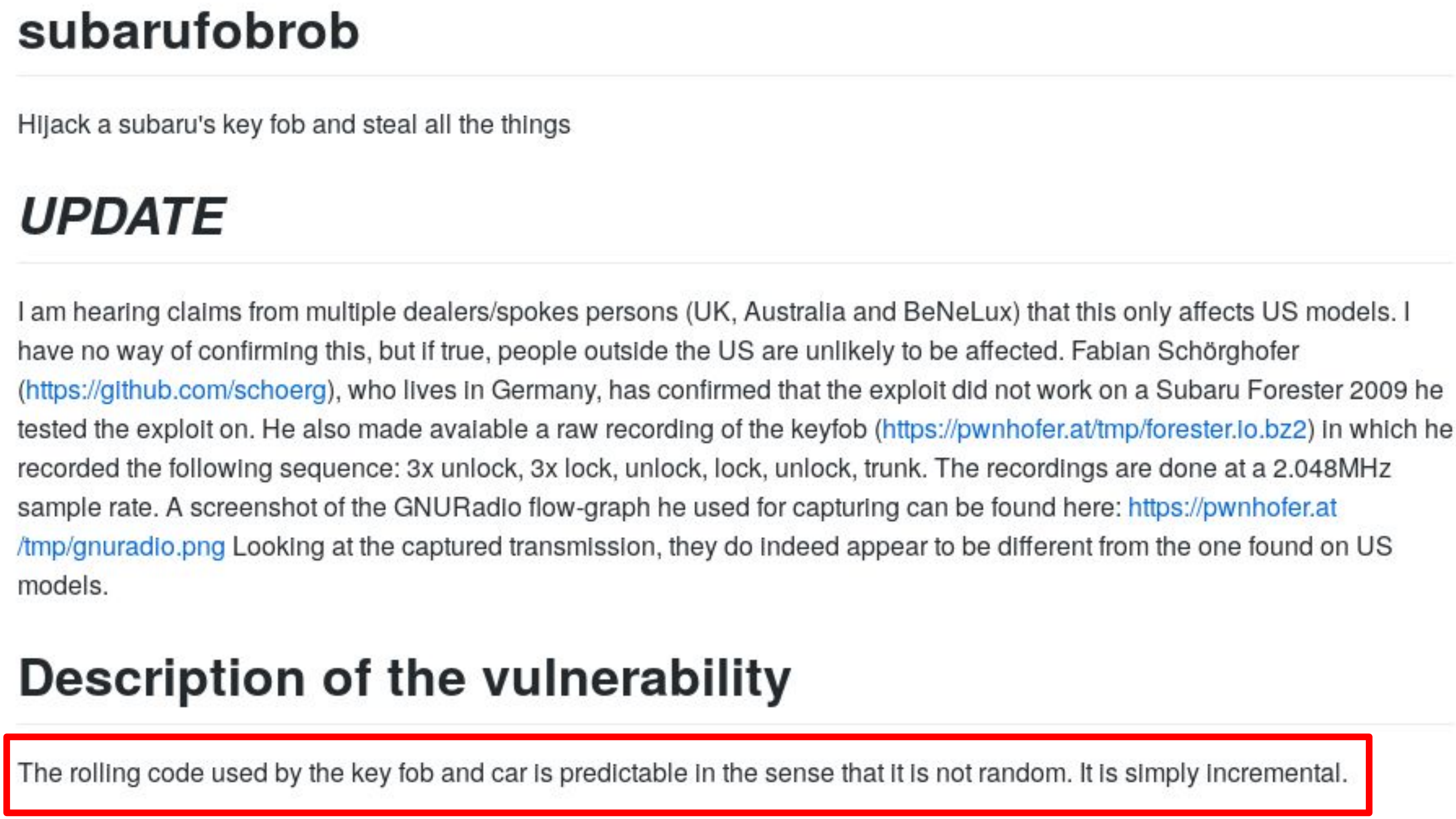
[5]

Other Stuff

- Java Services
  - Decompile

- There are web browsers

- Check out software update process
  - signature validation
  - install via USB
  - Check out Subaru Starlink analysis [7]



subarufobrob

Hijack a subaru's key fob and steal all the things

UPDATE

I am hearing claims from multiple dealers/spokes persons (UK, Australia and BeNeLux) that this only affects US models. I have no way of confirming this, but if true, people outside the US are unlikely to be affected. Fabian Schörghofer (https://github.com/schoerg), who lives in Germany, has confirmed that the exploit did not work on a Subaru Forester 2009 he tested the exploit on. He also made avalable a raw recording of the keyfob (https://pwnhofer.at/tmp/forester.io.bz2) in which he recorded the following sequence: 3x unlock, 3x lock, unlock, lock, unlock, trunk. The recordings are done at a 2.048MHz sample rate. A screenshot of the GNURadio flow-graph he used for capturing can be found here: https://pwnhofer.at/tmp/gnuradio.png Looking at the captured transmission, they do indeed appear to be different from the one found on US models.

Description of the vulnerability

The rolling code used by the key fob and car is predictable in the sense that it is not random. It is simply incremental.

The Future

Future Stuff

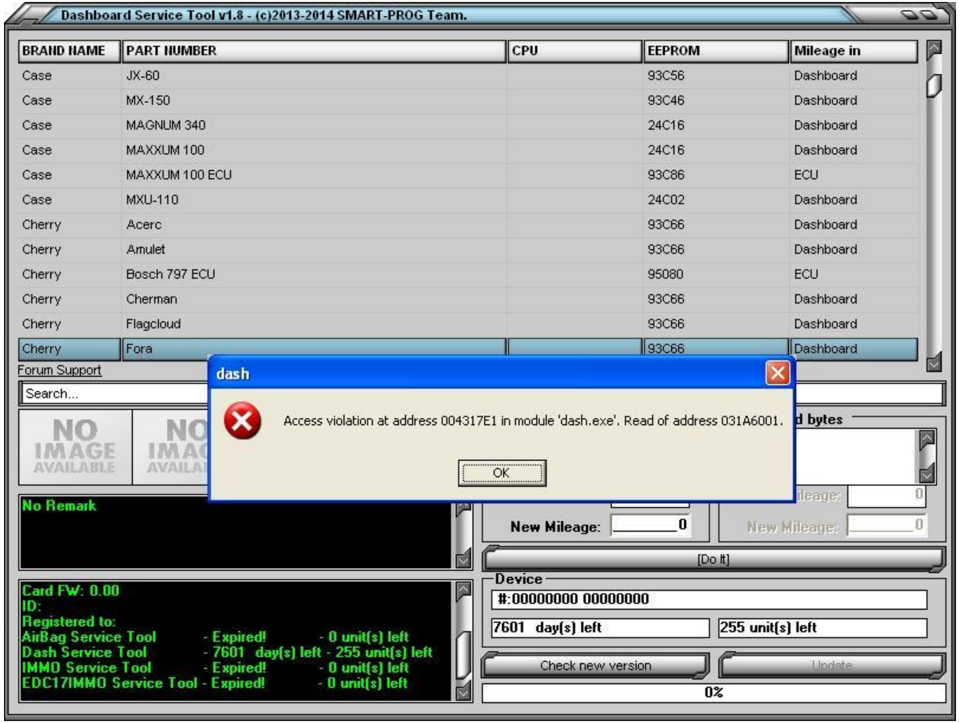
- AUTOSAR / SecOC:
    - Analyze things left off in standard that manufacturers build (or let build) themselves
        - e.g. key distribution

- Do even moar ***CANalyzat0r***?

- Containers in cars?
    - least privilege
    - proper isolation of 3rd party blobs

Stuff Worth Checking Out

mhhauto



Dashboard Service Tool v1.8 - (c)2013-2014 SMART-PROG Team.

BRAND NAME	PART NUMBER	CPU	EEPROM	Mileage in
Case	JX-60		93C56	Dashboard
Case	MX-150		93C46	Dashboard
Case	MAGNUM 340		24C16	Dashboard
Case	MAXXUM 100		24C16	Dashboard
Case	MAXXUM 100 ECU		93C86	ECU
Case	MXU-110		24C02	Dashboard
Cherry	Acerc		93C66	Dashboard
Cherry	Amulet		93C66	Dashboard
Cherry	Bosch 797 ECU		95080	ECU
Cherry	Cherman		93C66	Dashboard
Cherry	Flagcloud		93C66	Dashboard
Cherry	Fora		93C66	Dashboard

dash

Access violation at address 004317E1 in module 'dash.exe'. Read of address 031A6001.

```
No Remark
```

```
Card FW: 0.00
ID:
Registered to:
AirBag Service Tool         - Expired!          - 0 unit(s) left
Dash Service Tool           - 7601  day(s) left - 255 unit(s) left
IMMO Service Tool           - Expired!          - 0 unit(s) left
EDC17IMMO Service Tool - Expired!          - 0 unit(s) left
```

New Mileage: 0

Mileage: 0

New Mileage: 0

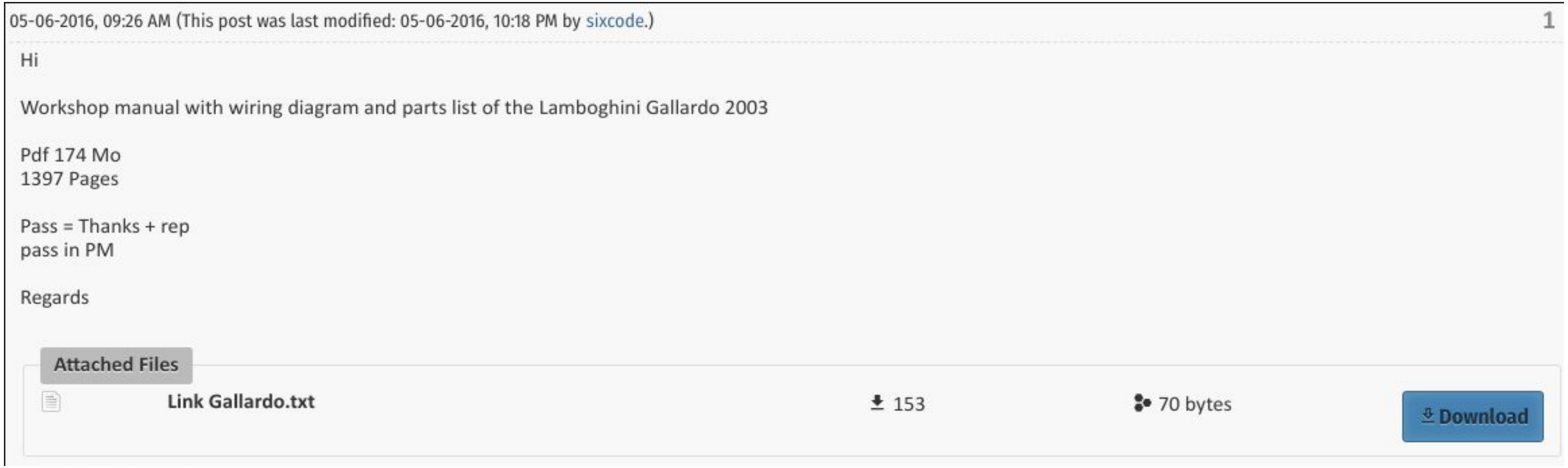
Device

#:00000000 00000000

7601 day(s) left

255 unit(s) left

mhhauto



05-06-2016, 09:26 AM (This post was last modified: 05-06-2016, 10:18 PM by sixcode.) 1

Hi

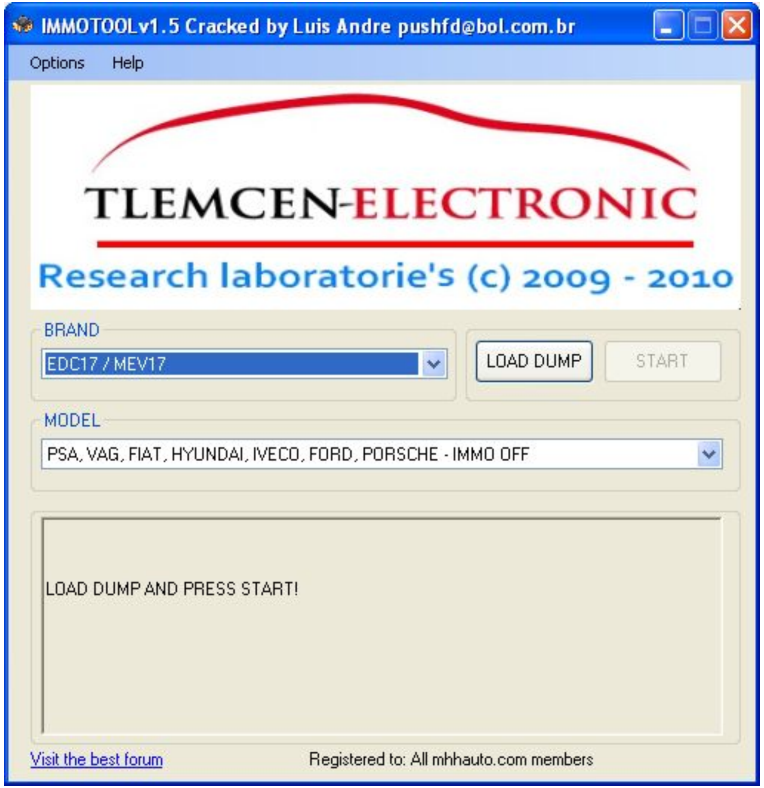
Workshop manual with wiring diagram and parts list of the Lamboghini Gallardo 2003

Pdf 174 Mo
1397 Pages

Pass = Thanks + rep
pass in PM

Regards

mhhauto



IMMOTOOLv1.5 Cracked by Luis Andre pushfd@bol.com.br

BRAND: EDC17 / MEV17

MODEL: PSA, VAG, FIAT, HYUNDAI, IVECO, FORD, PORSCHE - IMMO OFF

LOAD DUMP AND PRESS START!

Registered to: All mhhauto.com members

Also Interesting

- KeenLab BMW Research
- Miller/Valasek Research
- QNX Security

Also Interesting

- KeenLab BMW Research
- Miller/Valasek Research
- QNX Security
- SCHUTZWERK - We're hiring

References

[1] https://res.cloudinary.com/teepublic/image/private/s--3CAlo5WS--/t_Preview/b_rgb:ffffff,c_limit,f_jpg,h_630,q_90,w_630/v1534119152/production/designs/3012836_0.jpg

[2] https://keenlab.tencent.com/en/2018/05/22/New-CarHacking-Research-by-KeenLab-Experimental-Security-Assessment-of-BMW-Cars/

[3] https://de.wikipedia.org/wiki/Controller_Area_Network#/media/File:CAN-Bus-frame_in_base_format_without_stuffbits.svg

[4] https://a2-freun.de/forum/forums/topic/27793-liste-can-ids/

[5] https://cdn.shopify.com/s/files/1/0244/5107/products/IMG_0012_1024x1024.jpg?v=1371786976

[6] https://www.8devices.com/products/usb2can

[7] https://github.com/sgayou/subaru-starlink-research/blob/master/doc/README.md#harman-and-qnx

[8] https://www.autosar.org/fileadmin/user_upload/standards/classic/4-3/AUTOSAR_SWS_SecureOnboardCommunication.pdf

Thanks!

github/ps1337
@CaptnBanana