
SBA
Research

The Private Cloud within Your Car

Containers, Podman and Kubernetes in your automotive products?

Bundesministerium
Klimaschutz, Umwelt,
Energie, Mobilität,
Innovation und Technologie

Bundesministerium
Digitalisierung und
Wirtschaftsstandort

FFG
Forschung wirkt.

wirtschafts
agentur
wien
Ein Fonds der
Stadt Wien

European
Commission

FWF
Der Wissenschaftsfonds.

netidee
OPEN INNOVATIONS

K8s plugins: Kubectl your Tesla

Savithru Lokanath · Follow

3 min read · Apr 1, 2020

5

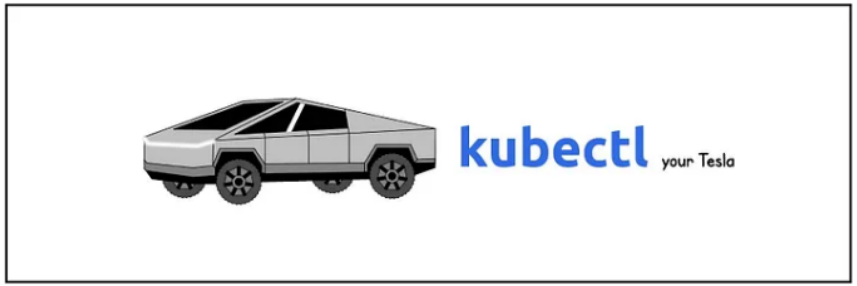
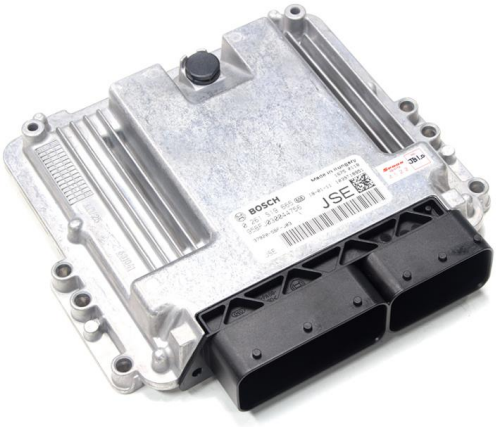


Fig1. Kubectl plugins

... is this safe?

https://savvythrough.medium.com/plugins-kubectl-your-tesla-59bccf0d4e88


BOSCH
JSE


1/min
x1000
SET
km/h

Images: Knight Rider, Universal Television

Car Architecture and Ecosystem

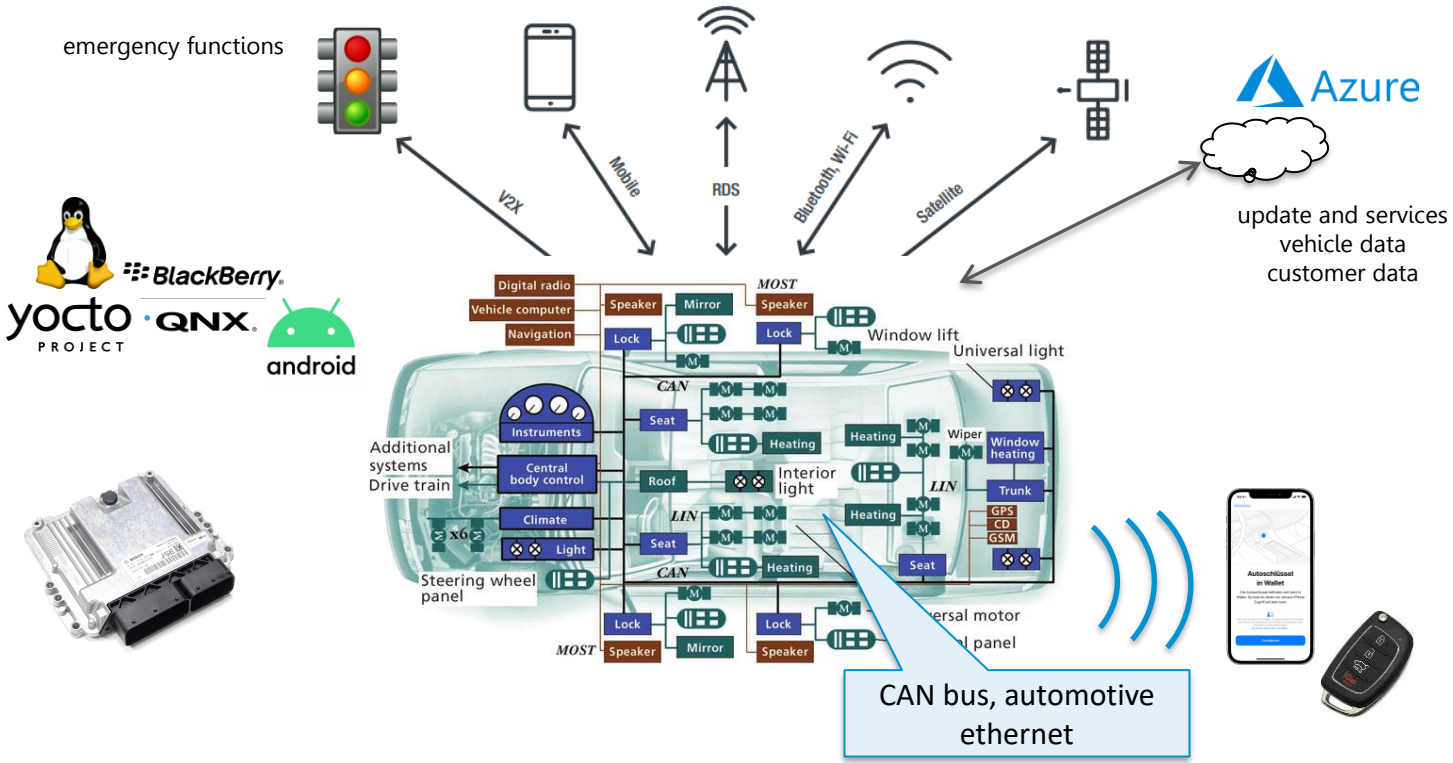
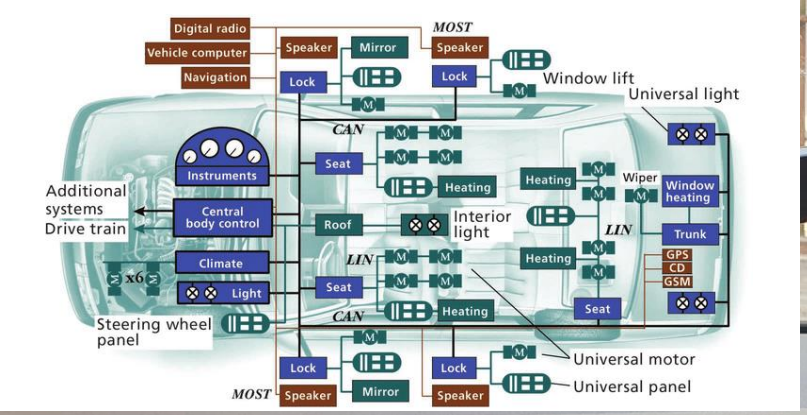


Image original: https://www.researchgate.net/figure/One-subset-of-a-modern-vehicles-network-architecture-showing-the-trend-toward_fig1_2955571


Digital radio
Vehicle computer
Navigation
Speaker
Mirror
MOST
Speaker
Lock
Lock
Window lift
Universal light
CAN
Seat
Heating
Heating
Wiper
Window heating
Instruments
Additional systems
Drive train
Central body control
Roof
Interior light
LIN
Trunk
Climate
Heating
GPS
CD
GSM
x6
Light
Seat
Seat
Steering wheel panel
CAN
Heating
Universal motor
Universal panel
Lock
Lock
MOST
Speaker
Mirror
Speaker

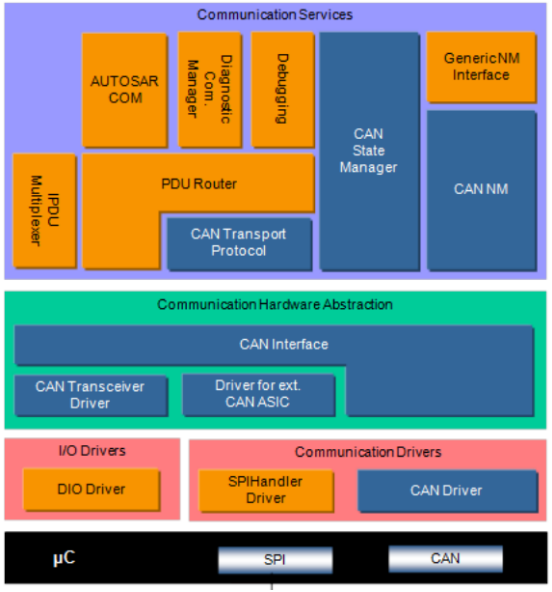
KEYSIGHT
DSOX1204G
Digital Storage Oscilloscope
70 MHz 2 GSa/s
MEGAZoom
InfiniiVision
Auto Scale
Default Setup
Horizontal
Entry
Measure
Analyze
Meas
Save Recall
Utility
Display
Vertical
FFT
Math
Help
Back
BW Limit
Fine
Invert
Probe
Channel 2
Coupling
AC
ΔX:
+34.01360544kHz
ΔY(1):
-40.0000V
Demo
Probe Comp
Gen Out
10:03
P

File Edit View Terminal Tabs Help

ba@automotive-security: ~/embedded-container × sba@raspberrypi: ~

```
49|ms | ID   | data ...      < can0 # l=20 h=100 t=500 slots=67 >
00100 | 151  | 00 E0 80 60                  ...`
00099 | 35B  | 00 32 0C 80 00 3E 0E E3 .2...>..
00199 | 367  | A0 00 00 27 7F 00 02 00 ...'....
00199 | 369  | FC 00 26 40 10 00 00 00 ..&@....
00099 | 3C3  | 85 81 00 00 80 40 21 B9 .....@!.
00099 | 3E1  | 12 00 26 0C 7E 32 82 00 ..&.~2..
01997 | 484  | 37 94 90 82 13 45 28 63 7....E(c
00799 | 485  | 07 0C 10 18 D9 C2 27 65 ......'e
01799 | 486  | B6 AE DE 72 FF C4 07 CE ...r....
00047 | 531  | 00 00 80 80                  ....
00100 | 555  | E8 F9 7B 01 32 10 04 63 ..{.2..c
00498 | 571  | AF 00 00 00 00 00            ......
00298 | 58C  | 10 81 00 00 26 10 00 02 ....&...
01001 | 5A1  | C1 89 A7 17 65 80 70 00 ....e.p.
41996 | 5A3  | 30 20 58 10 46 3F 93 0C 0 X.F?..
00199 | 5B5  | 0C 03 00 00 00 00 00 0F ........
01009 | 63C  | 3B CF 05                     ;..
01000 | 65D  | 92 65 A9 A1 76 BB F0 07 .e..v...
00199 | 65F  | 02 56 35 30 31 33 30 36 .V501306
00999 | 66C  | 3C 52 05 00                  <R..
00990 | 66F  | 3A CF 00                     :..
00999 | 67D  | 3C 8F 00                     <..
02000 | 6DA  | 32 82 03 00 0A 00 03 00 2.......
00440 | 6DB  | 30 5A 03                     0Z.
00999 | 6DC  | 32 D0 64                     2.d
```

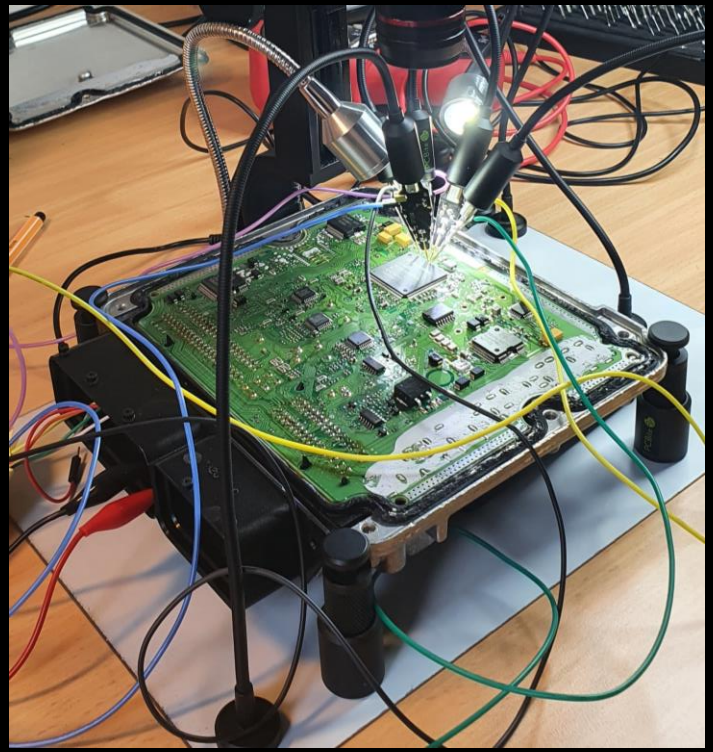
Example: AutoSAR, Platform, Chip, Software



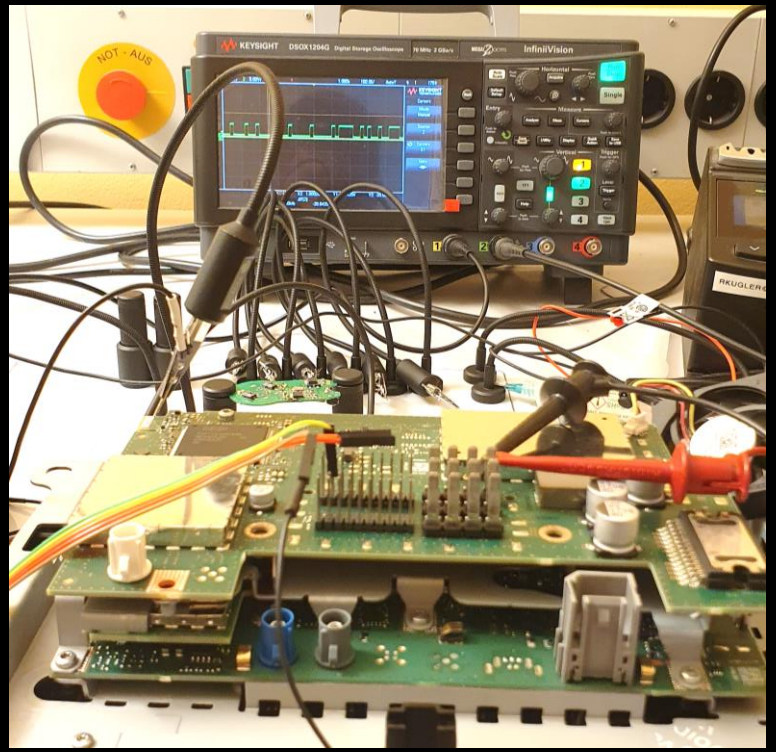
between 70 and 100 ECUs being installed in every modern vehicle

Image: Specification of CAN Interface AUTOSAR CP Release 4.3.1, p11

Electronic Control Units (opened)

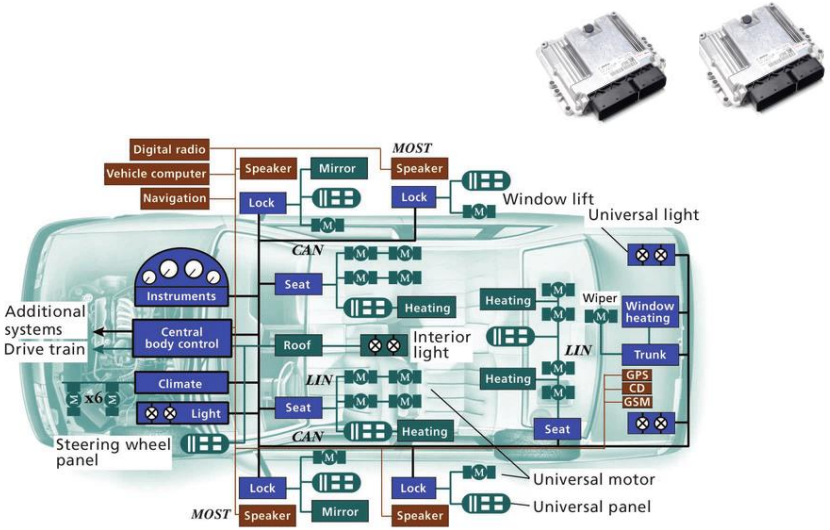


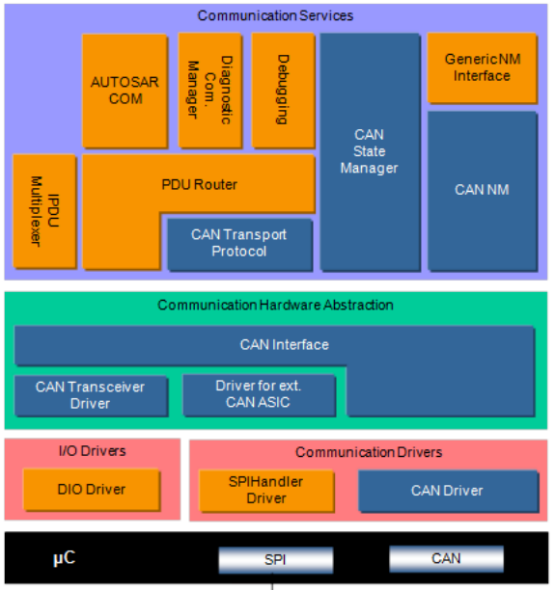
JTAG access on the PCB

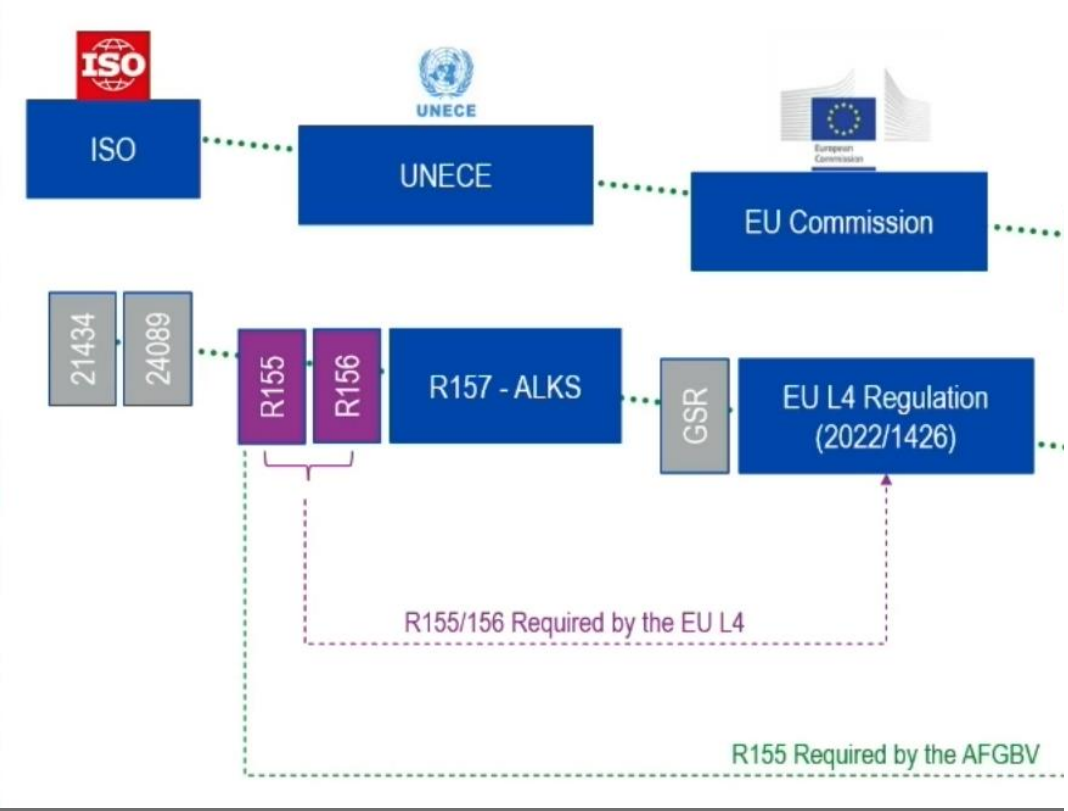


CAN access + on-board console via UART

How to maintain/update/configure hundrets of ECUs?







The road to AV appl

ISO 24089:2023

Road vehicles — Software update engineering

Abstract

Preview

This document specifies requirements and recommendations for software update engineering for road vehicles on both the organizational and the project level.

This document is applicable to road vehicles whose software can be updated.

The requirements and recommendations in this document apply to vehicles, vehicle systems, ECUs, infrastructure, and the assembly and deployment of software update packages after the initial development.

This document is applicable to organizations involved in software update engineering for road vehicles. Such organizations can include vehicle manufacturers, suppliers, and their subsidiaries or partners.

This document establishes a common understanding for communicating and managing activities and responsibilities among organizations and related parties.

The development of software for vehicle functions, except for software update engineering, is outside the scope of this document.

Finally, this document does not prescribe specific technologies or solutions for software update engineering.

General information

Status : Published

Publication date : 2023-02

Edition : 1

Number of pages : 24

Running containers in cars

October 19, 2022 | Pierre-Yves Chibon, Daniel J Walsh, Alexander Larsson

< Back to all posts

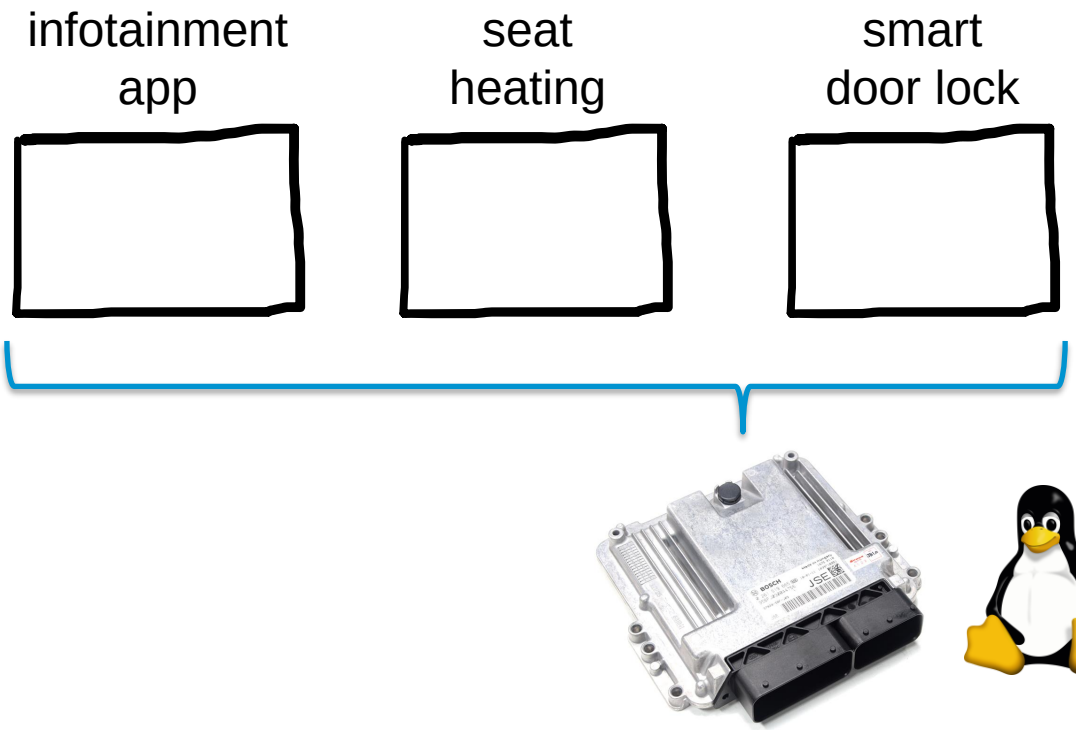
Tags: *Containers*

A little over a year ago, Red Hat announced its intention to collaborate with the automotive industry to help drive the transition to software-defined vehicles (SDVs). In May 2022, this intention became concrete with Red Hat and General Motors announcing their collaboration to help trailblaze SDVs at the edge. Our goal is to produce a base operating system to run all sorts of in-vehicle software for safety-critical use cases as well as non-safety ones.

Containers have become the de facto standard in the wider IT industry, and they are front and center in the vision of the **software-defined vehicle**, allowing for applications to be isolated, providing more flexibility for developers and deployment, and generally allowing for faster innovation. Red Hat leads the work on containers in the cloud, and now it's time to take that expertise to the automotive industry.

https://www.redhat.com/en/blog/running-containers-cars

Functions in application containers



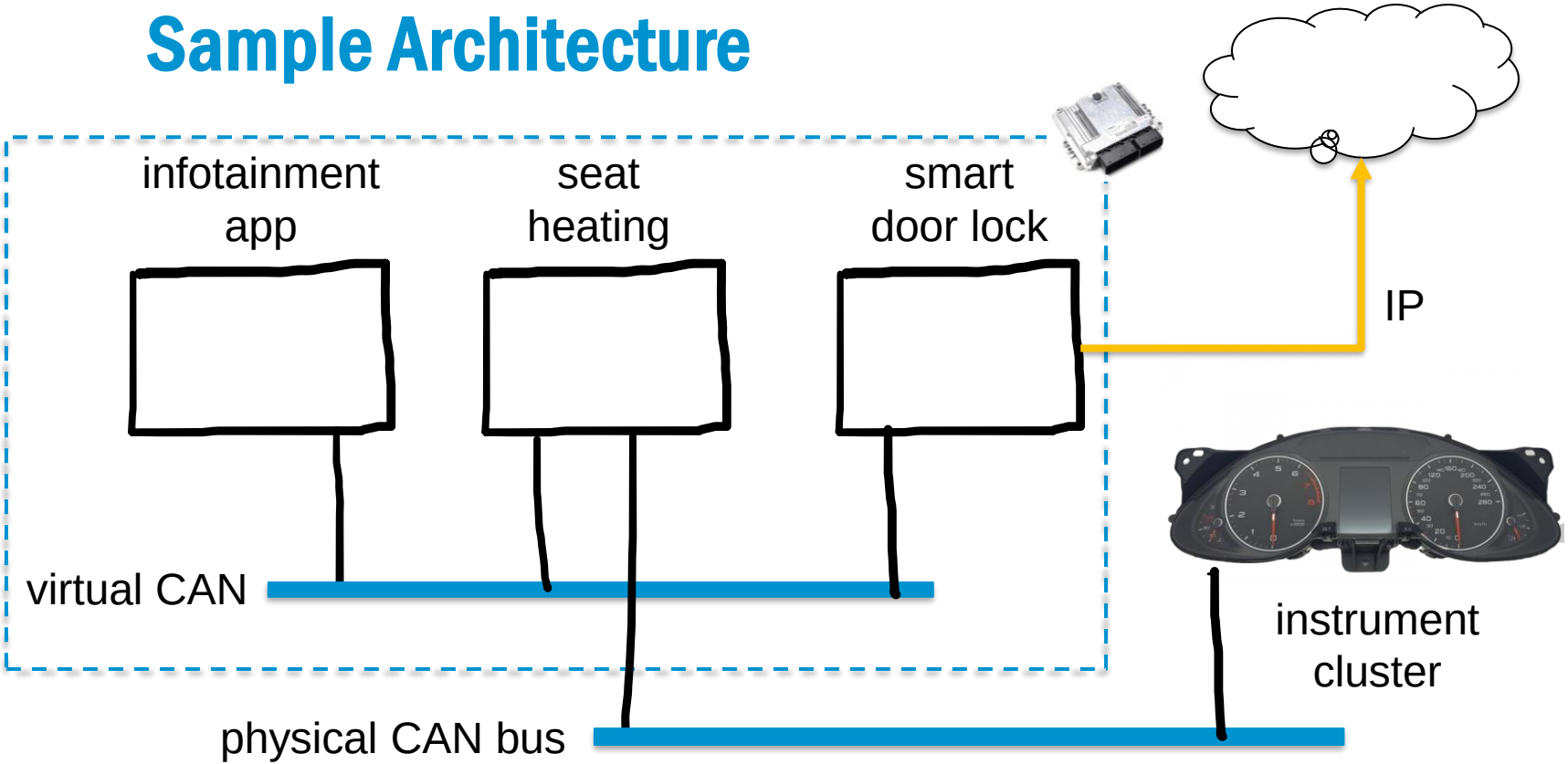
File Edit View Terminal Tabs Help

ba@automotive-security: ~/embedded-container/sim-container/rootfs

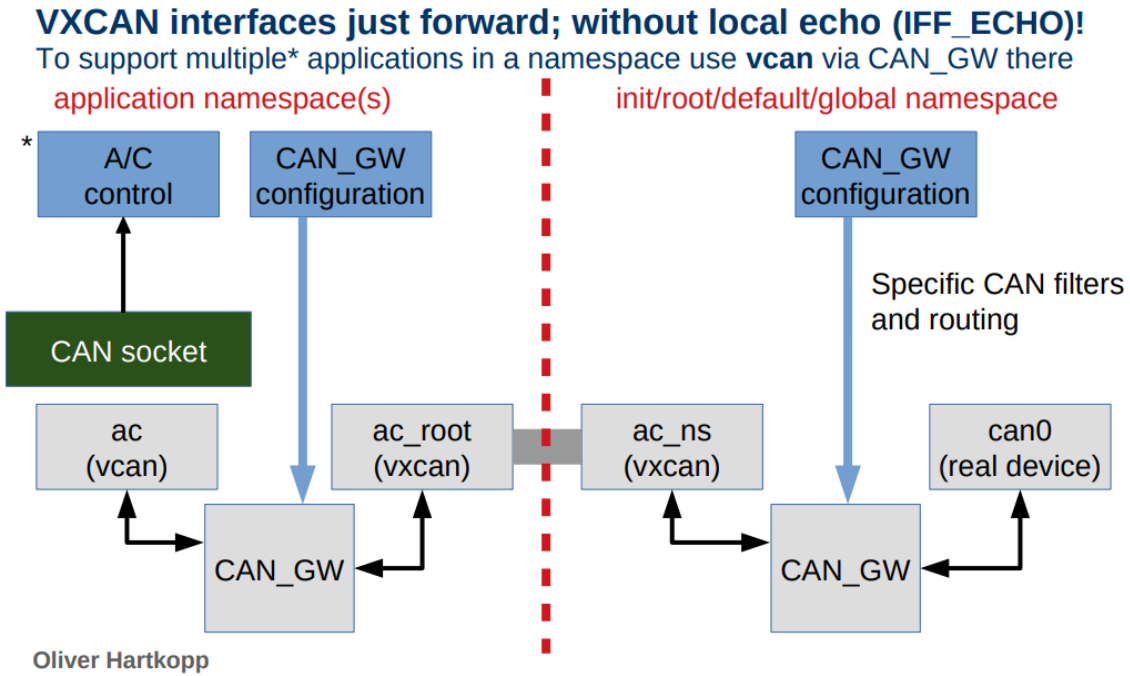
sba@automotive-security: ~/kitt

```
┌──(sba㉿automotive-security)-[~/kitt]
└─$
```

Sample Architecture

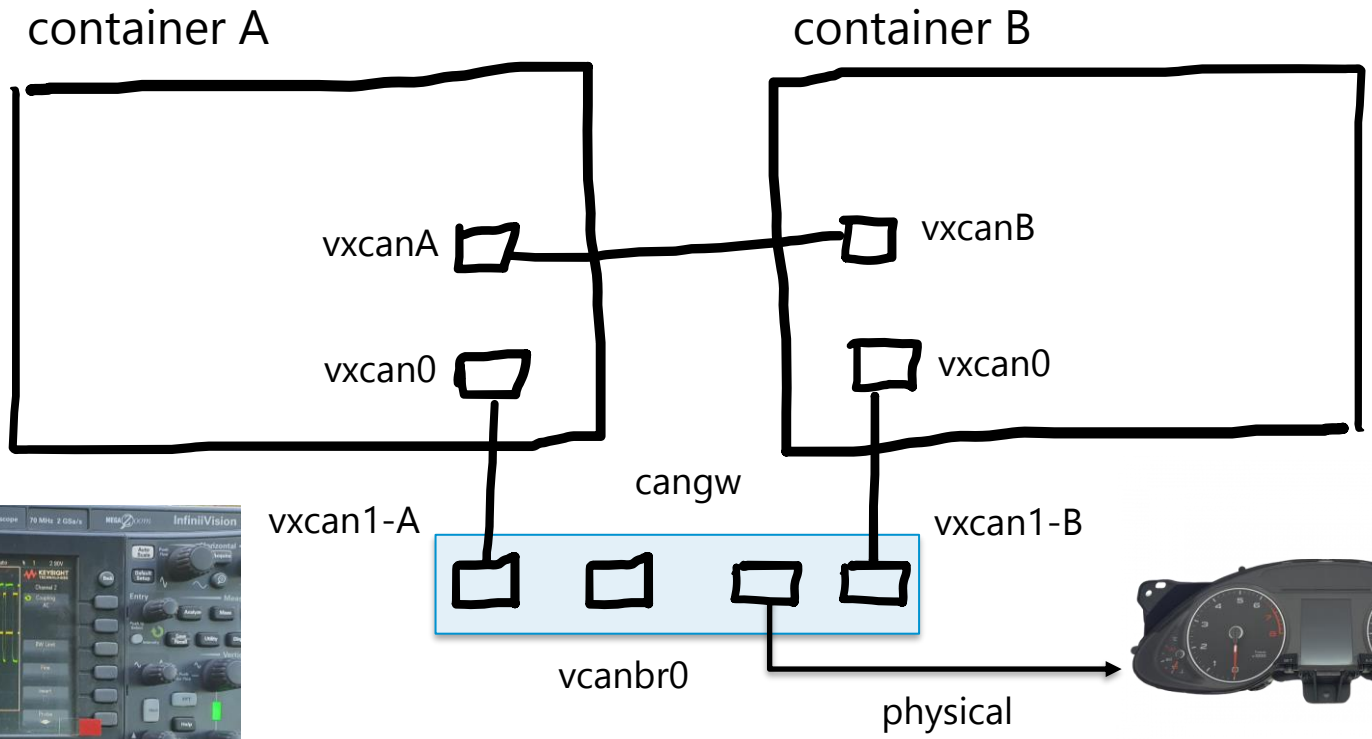


CAN and Containers, Automotive Grade Linux



https://wiki.automotivelinux.org/_media/agl-distro/agl2018-socketcan.pdf

Routing and Network namespaces (ethernet)



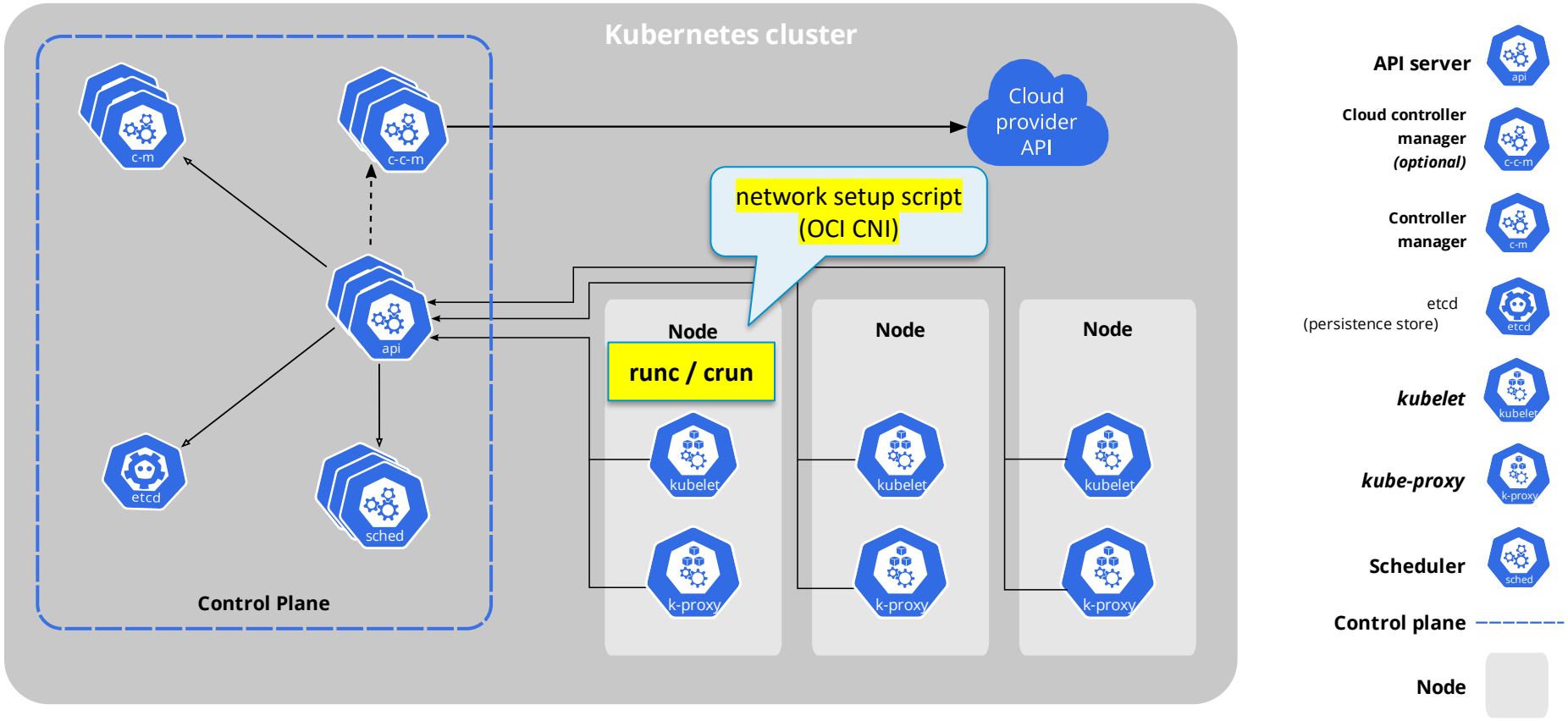


Image: https://kubernetes.io/docs/concepts/overview/components/

CNI

CNI - the Container Network Int

What is CNI?

CNI (*Container Network Interface*), a Cloud Native Computing Foun
libraries for writing plugins to configure network interfaces in Linux
plugins. CNI concerns itself only with network connectivity of conta
container is deleted. Because of this focus, CNI has a wide range of
implement.

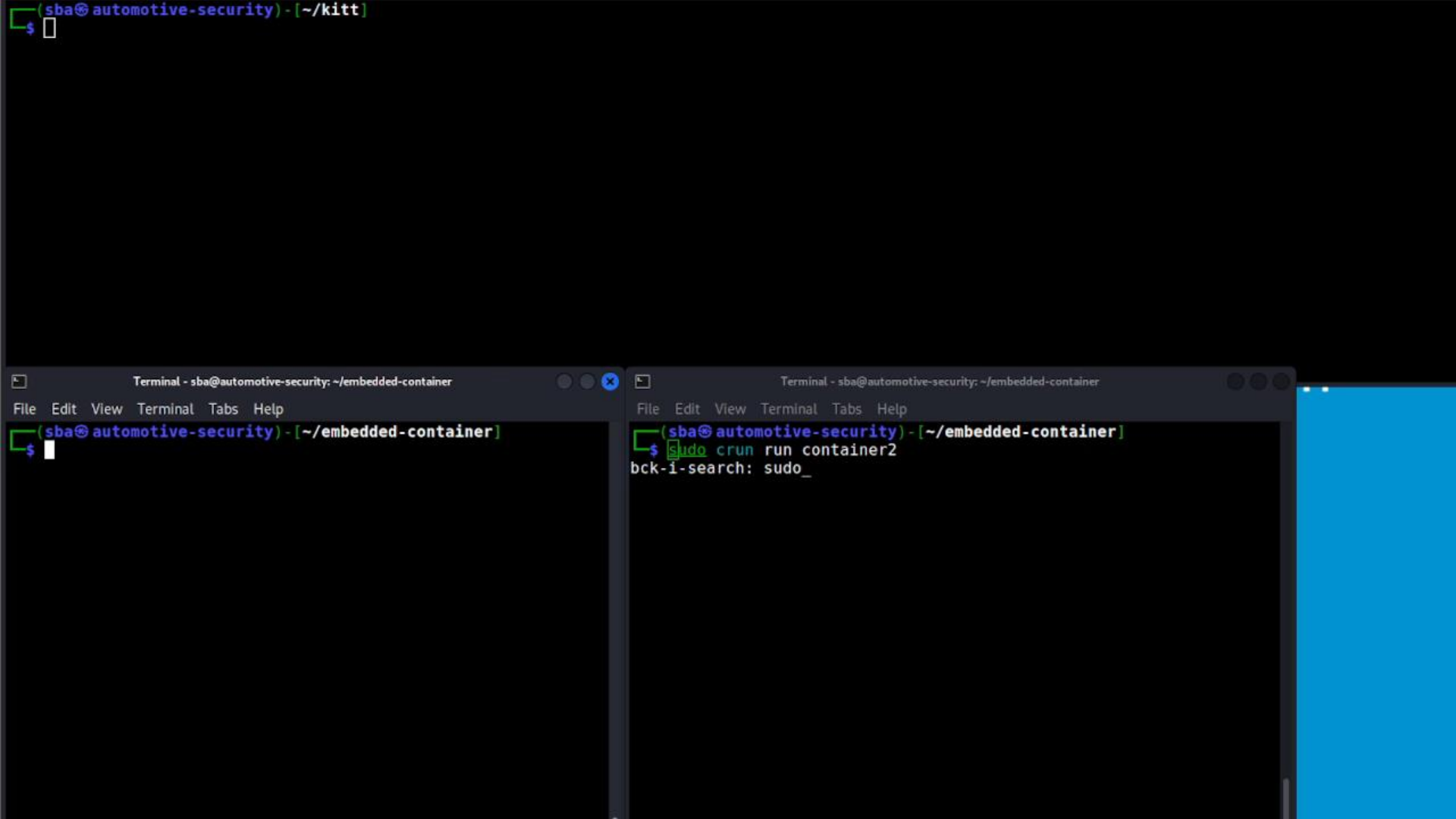
As well as the specification, this repository contains the Go source
applications and an example command-line tool for executing CNI
plugins and a template for making new plugins.

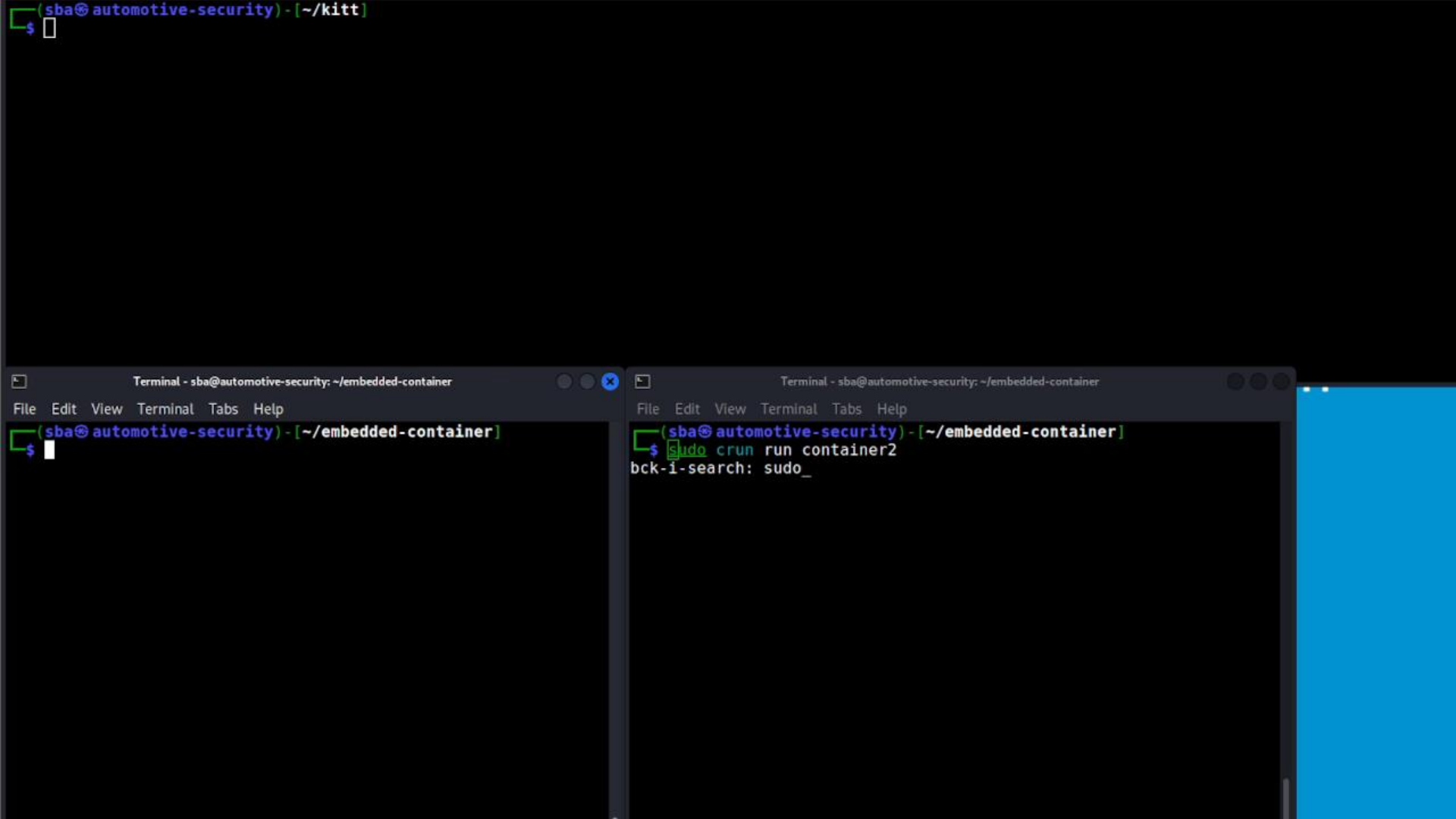
The template code makes it straight-forward to create a CNI plugin
also makes a good framework for creating a new container networ

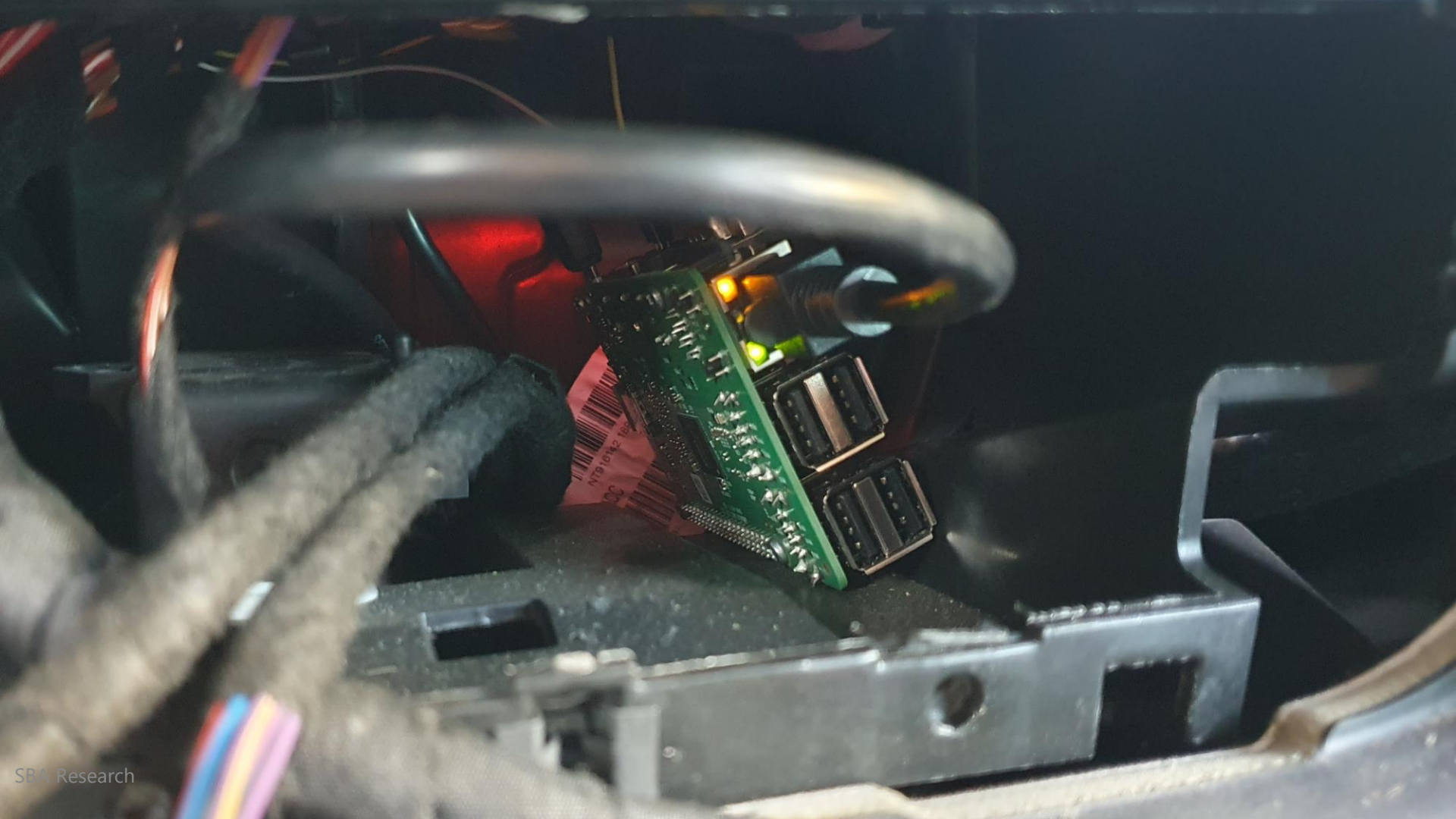
Here are the recordings of two sessions that the CNI maintainers h

- Introduction to CNI
- CNI deep dive

```
# check if process is running (debug)
ps $containerpid || exit 1

# set up a vcan bridge, all containers terminate there (like a d
ip link show $canbridgename
if [ $? -ne 0 ] ; then
    ip link add name $canbridgename type vcan
    ip link set $canbridgename up
fi

ip link add $vxcanname_container type vxcan peer name $vxcanname
ip link set $vxcanname_host up
ip link set $vxcanname_container netns $containerpid

# set up and rename to vxcan0 (container namespace)
nsenter -n -t $containerpid ip link set name vxcan0 dev $vxcanna
nsenter -n -t $containerpid ip link set vxcan0 up

# set up a bridge (gw) between can bridge and container vxcan (h
cangw -A -s $canbridgename -d $vxcanname_host -e
cangw -A -s $vxcanname_host -d $canbridgename -e

# flow in (physical to container)
#cangw -A -s "$physical_can" -d "$vxcanname_host" -e
# flow out (container to physical)
#cangw -A -d "$physical_can" -s "$vxcanname_host" -e
```


(sba㉿automotive-security)-[~/kitt]
$
Terminal - sba@automotive-security: ~/embedded-container
File Edit View Terminal Tabs Help
(sba㉿automotive-security)-[~/embedded-container]
$
Terminal - sba@automotive-security: ~/embedded-container
File Edit View Terminal Tabs Help
(sba㉿automotive-security)-[~/embedded-container]
$ sudo crun run container2
bck-i-search: sudo_


(sba㉿automotive-security)-[~/kitt]
$
Terminal - sba@automotive-security: ~/embedded-container
File Edit View Terminal Tabs Help
(sba㉿automotive-security)-[~/embedded-container]
$
Terminal - sba@automotive-security: ~/embedded-container
File Edit View Terminal Tabs Help
(sba㉿automotive-security)-[~/embedded-container]
$ sudo crun run container2
bck-i-search: sudo_



Is this safe?

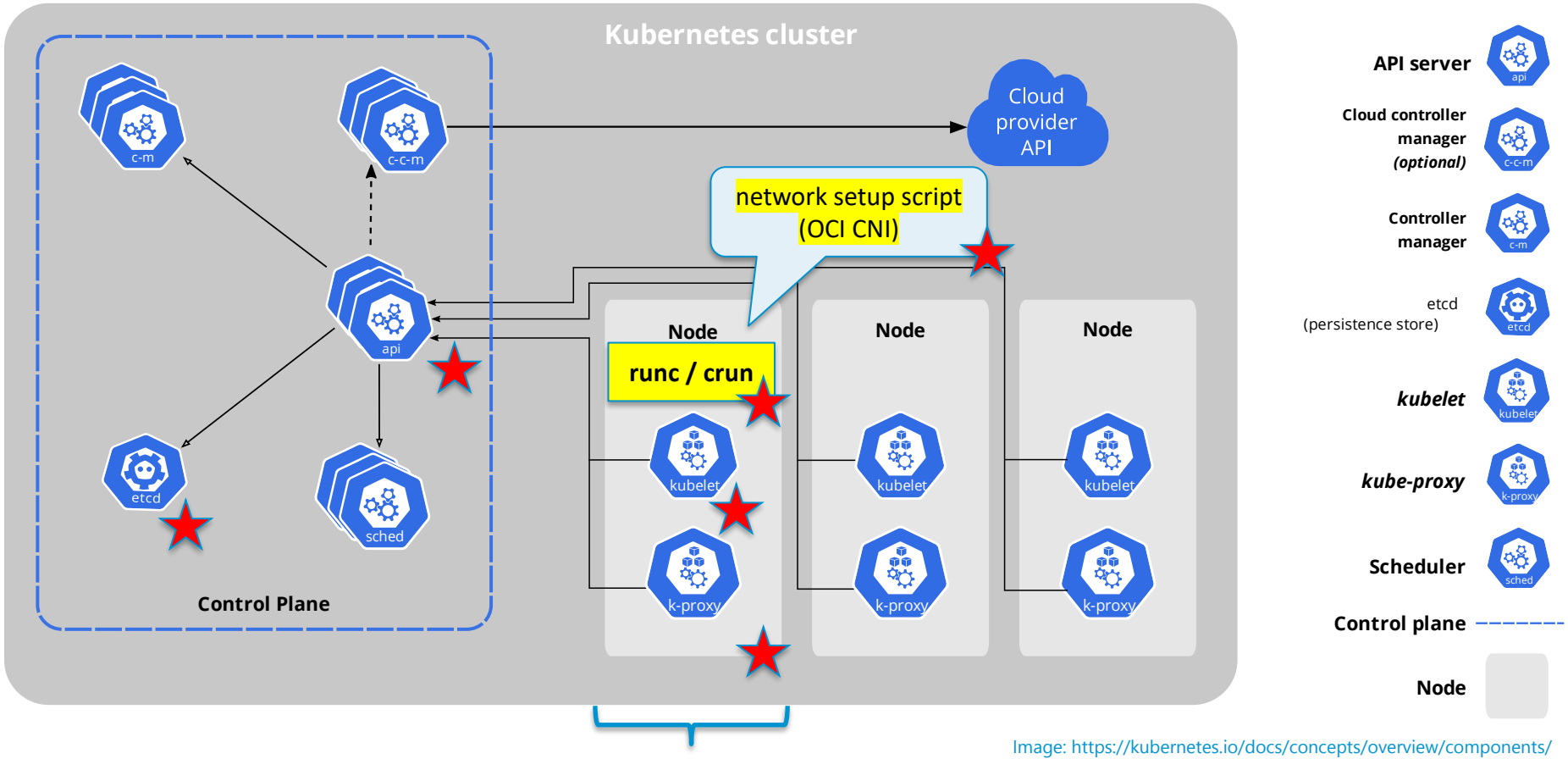
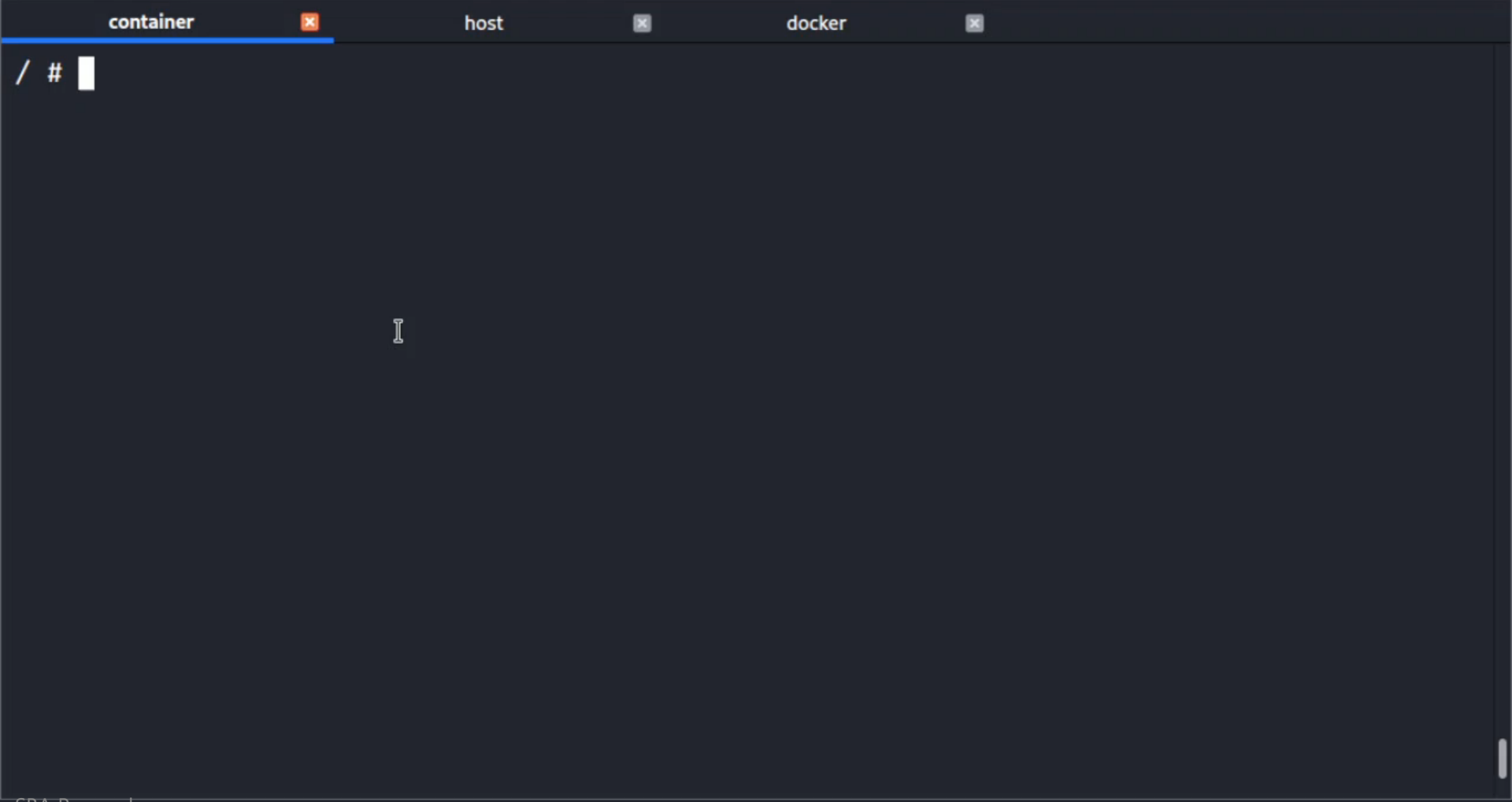


Image: https://kubernetes.io/docs/concepts/overview/components/

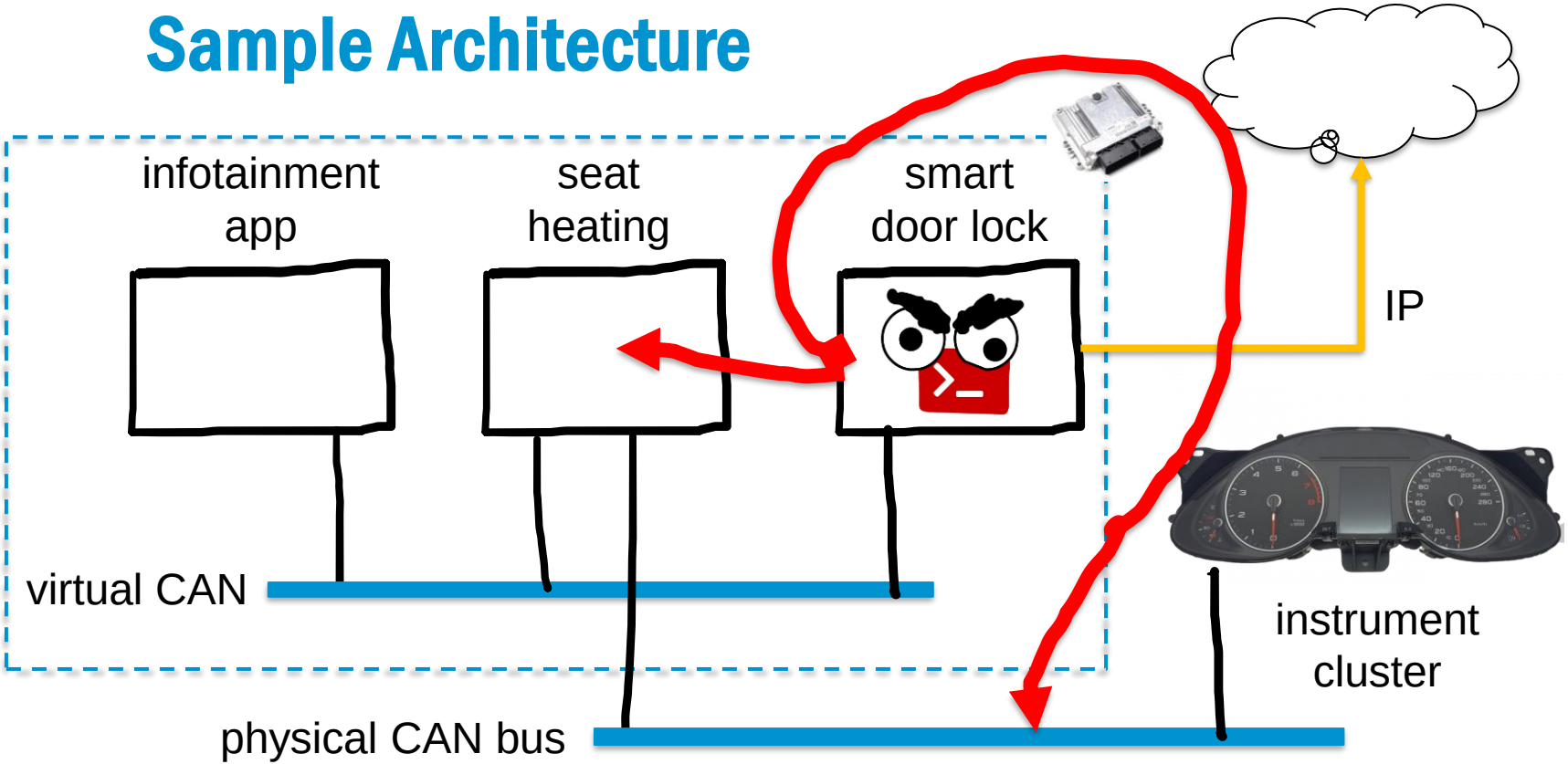
first steps

Vulnerabilities (CVE+user config)

- CVE-2014-???: Symlink docker cp
- CVE-2015-{3627,3629,3630,3631}: docker config, Volume is insane
- CVE-2016-9962: File descriptor open in the root filesystem, /proc/$pid/fd/$n
- CVE-2018-15664: Path senitasation, RENAME_EXCHANGE, LXD does it wierdly but good, Double check via proc
- CVE-2019-5736 18:00: /proc/self/exe, Reopen (O_RDONLY)
- CVE-2019-19921: Symlink exchange via VOLUME


container
host
docker
/ #

Sample Architecture



Future of Containers in Automotive („EDGE“)

- safety certified platforms
- software updates for fleets (for real)
- „zero-touch“ (over the air)
  - centralized configuration (pay-as-you-go)
  - centralized maintenance/updates
- DevOps for automotive applications

References and further reading

- Socket CAN
  - https://docs.kernel.org/networking/can.html
- Can-utils
  - https://github.com/linux-can/can-utils
- CANalyzat0r
  - https://github.com/schutzwerk/CANalyzat0r
- Caring Caribou
  - https://github.com/CaringCaribou/caringcaribou
- Scapy CAN layer
  - https://scapy.readthedocs.io/en/latest/api/scapy.layers.can.html
- Raspberry Pi/PiCan 3 shield
  - https://buyzero.de/products/pican-3
- ICSim
  - https://github.com/zombieCraig/ICSim
- Automotive Security Research Group (ASRG)
  - https://asrg.io/
- Design & separation of CAN applications
  - https://wiki.automotivelinux.org/_media/agl-distro/agl2018-socketcan.pdf

- Koscher, K., Czeskis, A., Roesner, F., Patel, S., Kohno, T., Checkoway, S., McCoy, D., Kantor, B., Anderson, D., Shacham, H., Savage, S.: Experimental Security Analysis of a Modern Automobile. In: 2010 IEEE Symposium on Security and Privacy. pp. 447–462 (May 2010). https://doi.org/10.1109/SP.2010.34
- Antonioli, Daniele, and Mathias Payer. "On the Insecurity of Vehicles Against Protocol-Level Bluetooth Threats." 2022 IEEE Security and Privacy Workshops (SPW). IEEE, 2022.
- Dr. Charlie Miller and Chris Valasek. Remote Exploitation of an Unaltered Passenger Vehicle. DEF CON 23 Hacking Conference. Las Vegas, NV: DEF CON. Aug. 2015.
- Florian Sommer, Jürgen Dürrwang, and Reiner Kriesten. “Survey and Classification of Automotive Security Attacks”. In: Information 10.4 (Apr. 2019), p. 148. ISSN: 2078-2489. DOI: 10.3390/info10040148. URL: http://dx.doi.org/10.3390/info10040148.
- ISO Central Secretary: Road vehicles – Unified diagnostic services (UDS) – Part 3: Unified diagnostic services on CAN implementation (UDSonCAN). Standard ISO 14229-3:2012, International Organization for Standardization, Geneva, CH (2012), https://www.iso.org/standard/55284.html
- Checkoway, S., McCoy, D., Kantor, B., Anderson, D., Shacham, H., Savage, S., Koscher, K., Czeskis, A., Roesner, F., Kohno, T.: Comprehensive Experimental Analyses of Automotive Attack Surfaces. In: Proceedings of the 20th USENIX Conference on Security. pp. 1–6. SEC’11, USENIX Association, USA (2011)

Reinhard Kugler

MATRIS Applied Research Consulting

SBA Research

Floragasse 7, 1040 Vienna

rkugler@sba-research.org

Bundesministerium Klimaschutz, Umwelt, Energie, Mobilität, Innovation und Technologie
Bundesministerium Digitalisierung und Wirtschaftsstandort
FFG Forschung wirkt.
wirtschaftsagentur wien Ein Fonds der Stadt Wien
European Commission
FWF Der Wissenschaftsfonds.
netidee OPEN INNOVATIONS