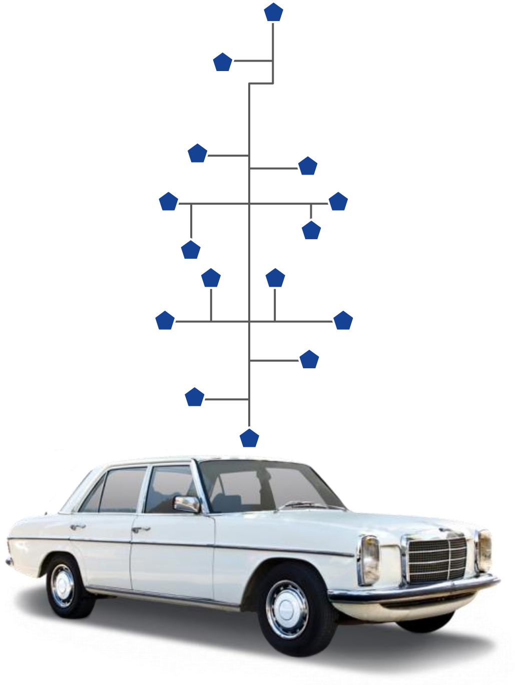
Need for a standardization of Ethernet firewalls in the automotive world

Presenter: Dr. Siddharth Shukla

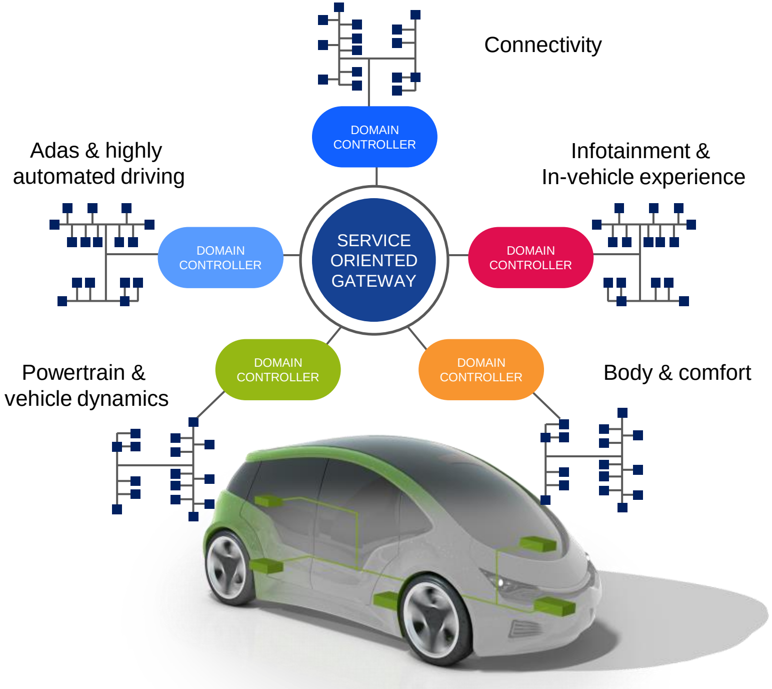
Firewalling in automotive

Firewalling in automotive

Trends in EE-architecture



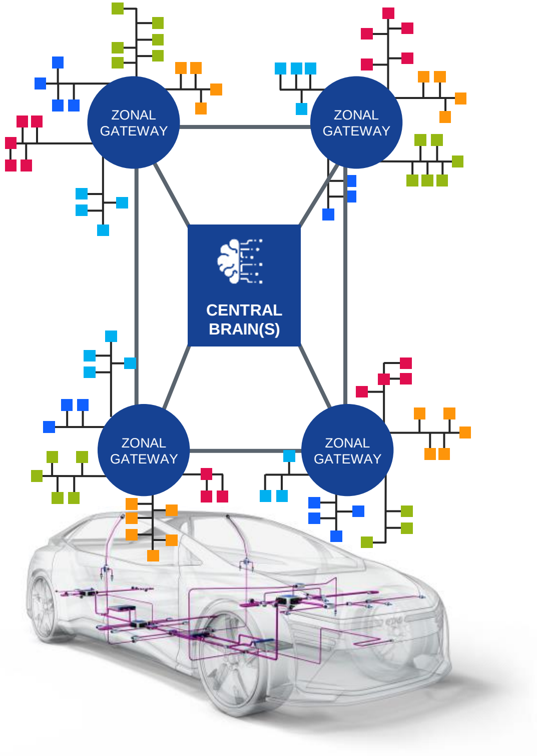
Unfit to future mobility



Logical restructure | **Domains**

Enabling autonomous vehicle

- Improved security and bandwidth
- Limited cross domain communication



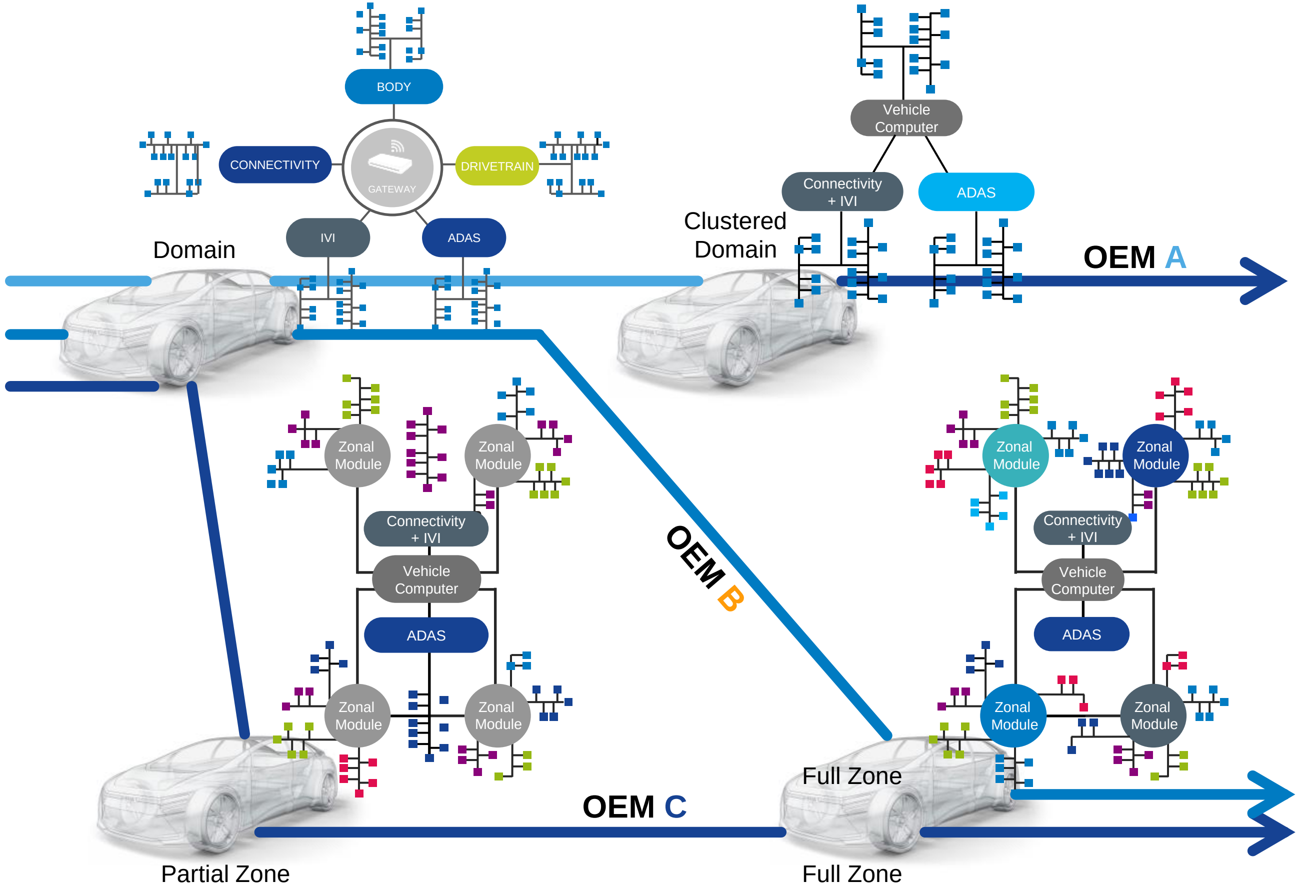
Physical restructure | **Zones**

Enabling software defined vehicle

- Shorter vehicle wiring harness
- High bandwidth communication link
- Re-use of hardware and software

Firewalling in automotive

OEM SPECIFIC ARCHITECTURAL TRANSITION

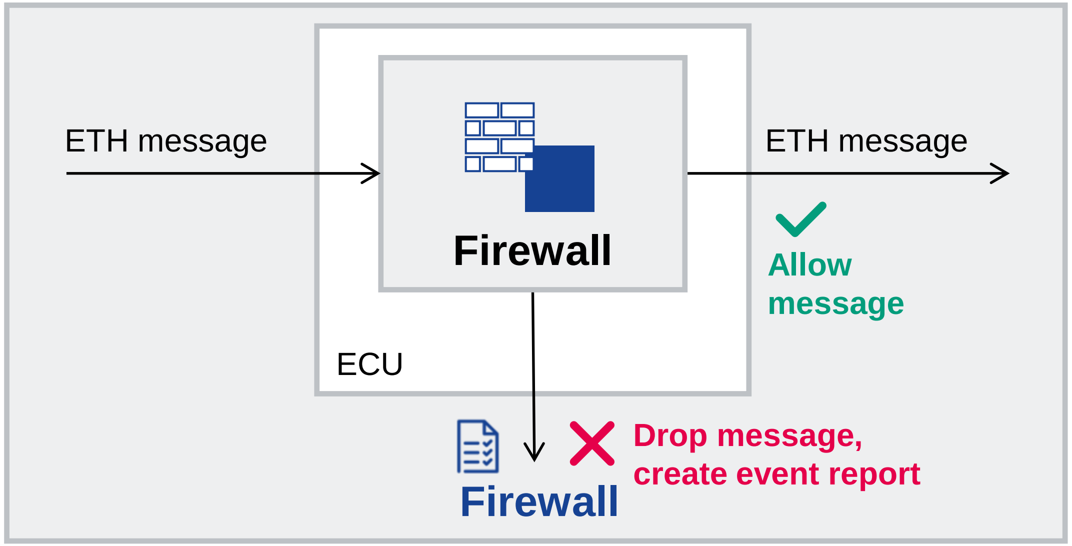


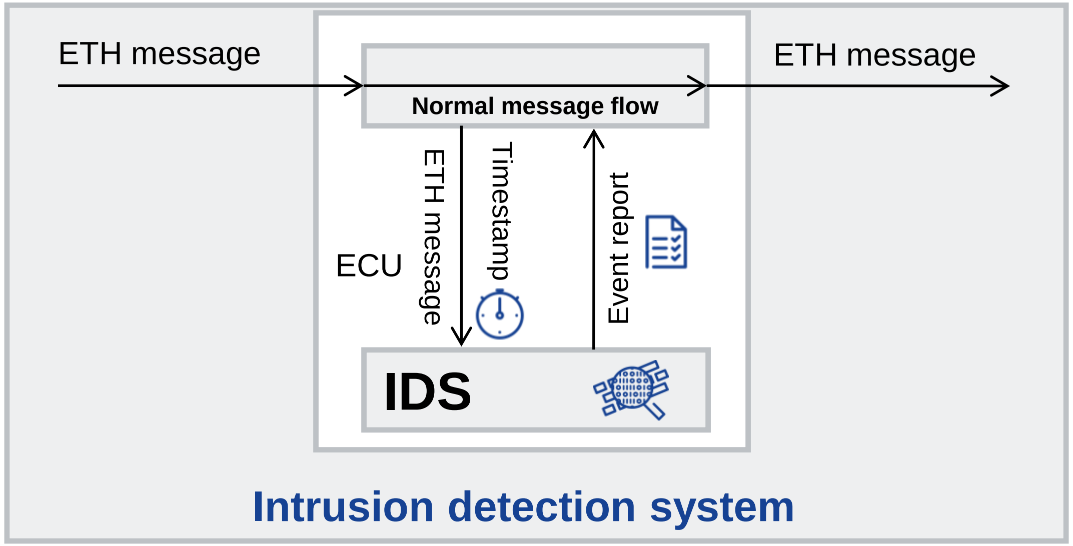
Transition is different from OEM to OEM

- Starting from different base architectures
- Different steps
- Hybrid solutions as the first step to zonal are very common
- IVI and ADAS are not included in physical zones and staying separate

Firewalling in automotive

Need for firewall in vehicles

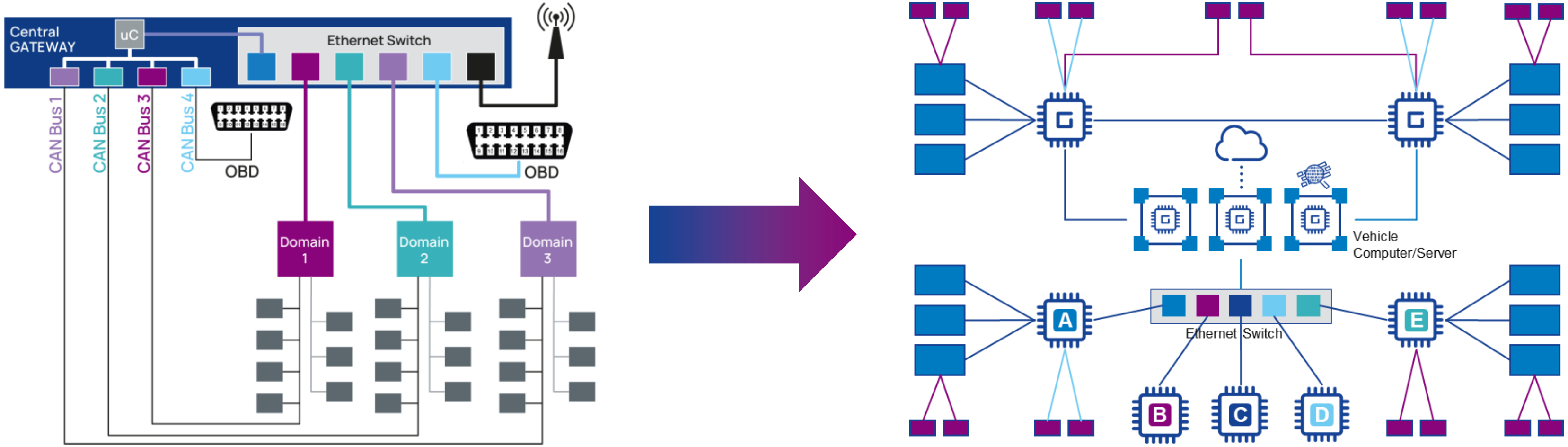




- Fulfill legislation requirement
  - GBT in China
  - UNECE
- Adding security check point at entry to stop unauthorized messages (defence in depth)
- We learned from IT world, use of ethernet requires firewall

Firewalling in automotive

New challenges when moving towards modern ee-architecture



- Distribution of domain specific sensor and actuator connectivity over the car to the zonal edge devices
- Domain functionality handled in the central compute, sometimes also local in the edges or distributed
- The connections from the edge get translated / packed into Ethernet frames and transmitted over the ethernet backbone
- Separation of compute and communication needs to happen in the center and in the edges
- Summary – communication policy is now complex and distributed (not logical but based on zones)

Firewalling in automotive

Key Ethernet use-cases for zonal E/E-architecture

1 Firewall and IDS on Vehicle computer

- Network separation using VLANs
- Firewall cross domain traffic
- Firewall end-to-end traffic
- Deep packet inspection for some frames
- Intrusion detection for ethernet

2 Firewall on Ethernet switch

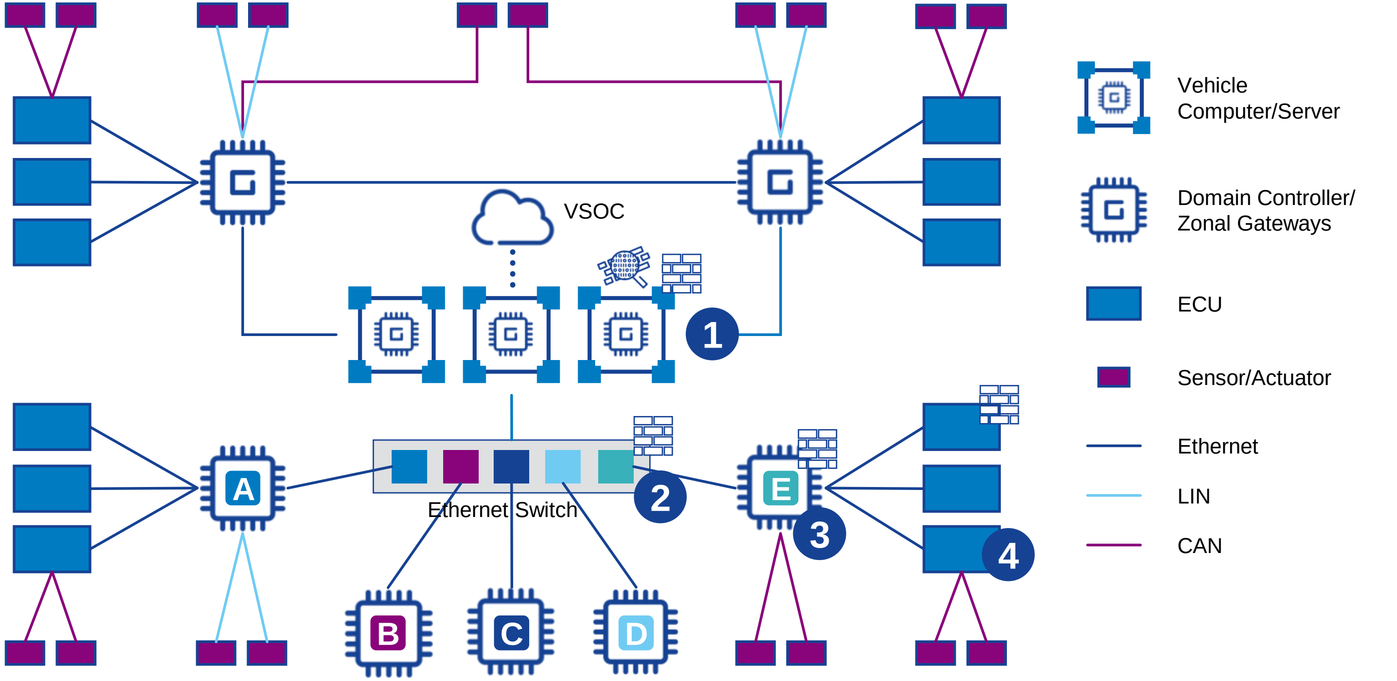
- Network separation using VLANs between domains A, B, C, D and E
- Firewall cross domain traffic at high speed between domains A, B, C, D and E
- Access control for vehicle server

3 Firewall and IDS on Domain controller

- Access Control and Firewall zonal traffic

4 Firewall on end ECUs

- Firewall for specific applications like EV charging ECU



Firewalling in automotive

Challenges

No standardized way to configure a firewall
High synchronization effort between OEM/Tier1, configuration process prone to errors

No harmonized connection to the IDS
Lack of standardized security events leads to high analysis efforts in the VSOC

No agreed minimal set of firewall functionality
High efforts in SW development to accommodate for all OEM specifications

Firewall standardization in AUTOSAR can address all of these challenges!

Firewall standardization in AUTOSAR

Addressing the challenges

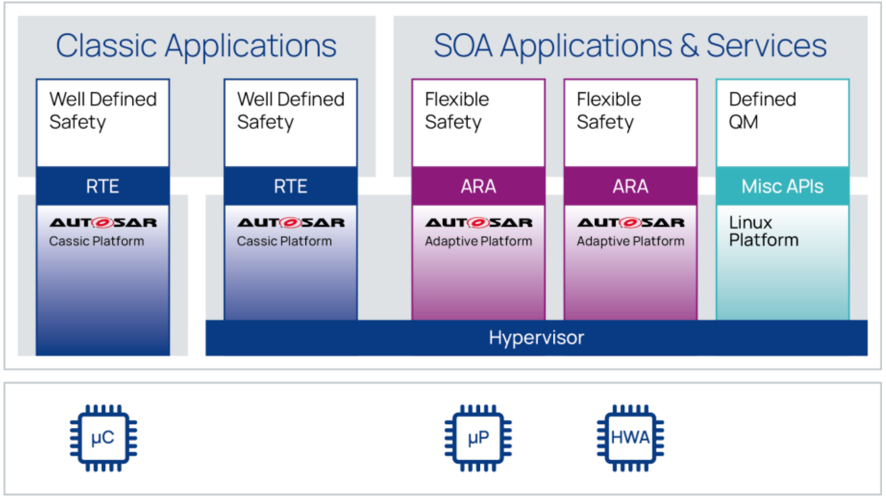
Firewall in AUTOSAR

AUTOSAR overview

What is AUTOSAR?

AUTOSAR is a standardized middleware for automotive ECUs.

Classic AUTOSAR: Safety, real-time OS → µCs
Adaptive AUTOSAR: Performance, flexible safety → µPs



Why use AUTOSAR to address the firewall challenges?

– Widely used in the automotive industry

– AUTOSAR toolchain can be used for firewall configuration

– AUTOSAR is industry consortium → Final solution aligned with needs of automotive industry

Firewall in AUTOSAR

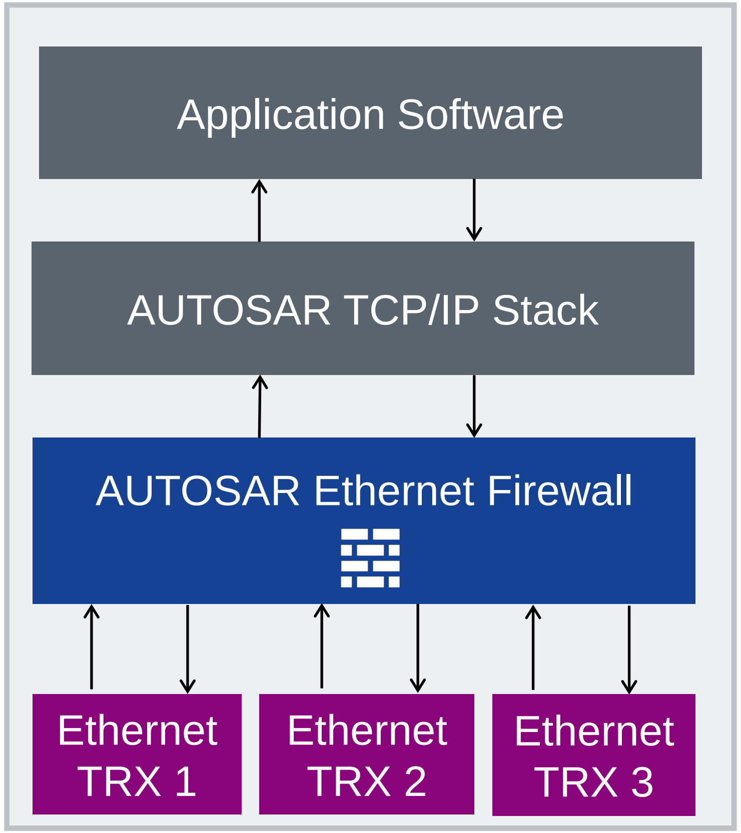
Firewall in AUTOSAR

Goals/Use-Cases

- Filtering of incoming/outgoing communication according to given ruleset
  - Stateless filtering
  - Stateful filtering
  - Deep packet inspection (e.g., SOME/IP, DoIP)
- **Standardization language for firewall filter rule configuration**
- Vehicle state sensitive firewall rule sets
- Standardized security events for IdsM

Applicable AUTOSAR standards

- All AUTOSAR (Classic/Adaptive) standards applicable
- Focus first on Adaptive – Classic/Switches in later step
- Standardized firewall configuration language available in ARXML
  → Can also be used in non-AUTOSAR projects



Firewall in AUTOSAR

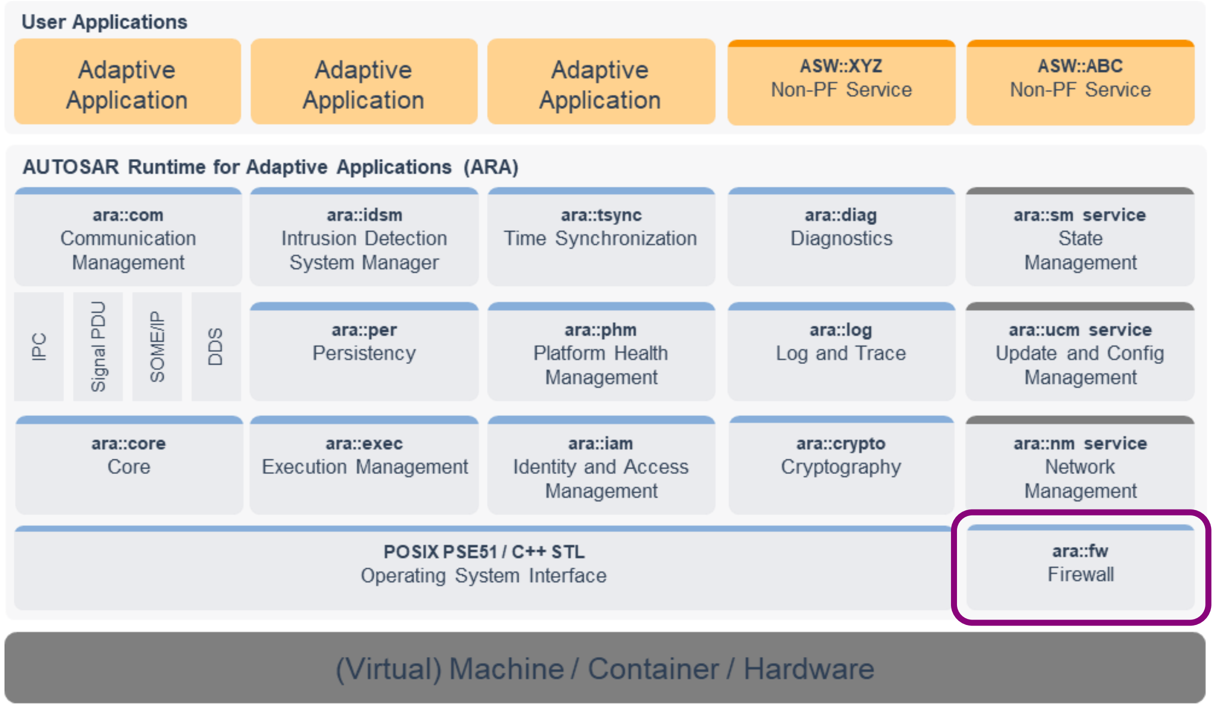
Host firewall in Adaptive AUTOSAR

Firewall available for Adaptive AUTOSAR with the AUTOSAR R22-11 release!

Firewall functionality can be found in the new functional cluster ara::fw

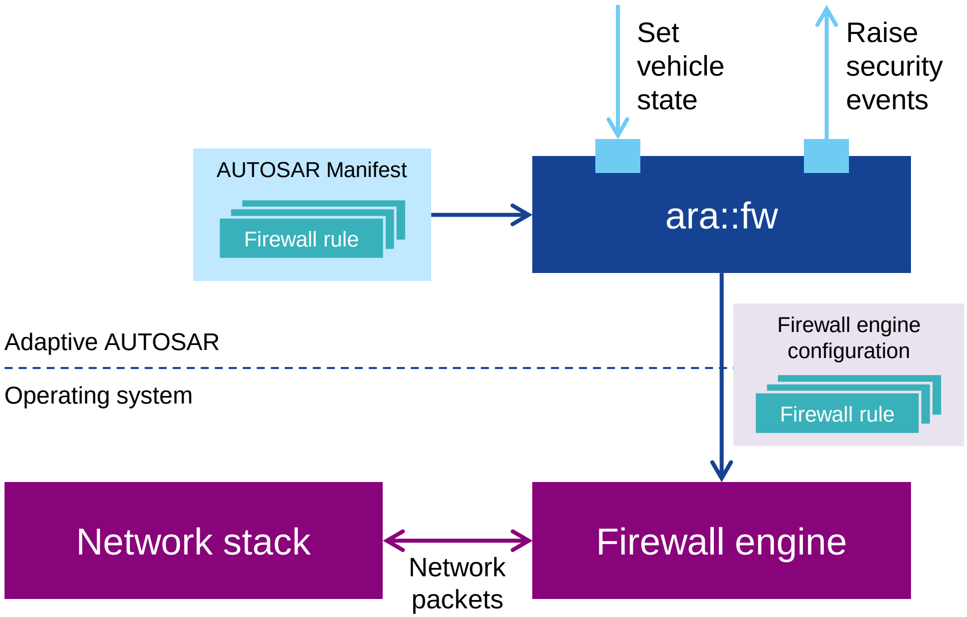
→ **Let‘s dive deeper into the specification**

– Firewall architecture
– Standardization language for firewall filter rule configuration
– Vehicle-state-based packet inspection
– Connection to the IDPS ecosystem


User Applications
Adaptive Application
Adaptive Application
Adaptive Application
ASW::XYZ Non-PF Service
ASW::ABC Non-PF Service
AUTOSAR Runtime for Adaptive Applications (ARA)
ara::com Communication Management
ara::idsm Intrusion Detection System Manager
ara::tsync Time Synchronization
ara::diag Diagnostics
ara::sm service State Management
IPC
Signal PDU
SOME/IP
DDS
ara::per Persistency
ara::phm Platform Health Management
ara::log Log and Trace
ara::ucm service Update and Config Management
ara::core Core
ara::exec Execution Management
ara::iam Identity and Access Management
ara::crypto Cryptography
ara::nm service Network Management
POSIX PSE51 / C++ STL Operating System Interface
ara::fw Firewall
(Virtual) Machine / Container / Hardware

Firewall in AUTOSAR

Firewall in Adaptive AUTOSAR



ara::fw is a **management module:**
→ Takes firewall configuration in **AUTOSAR format**
→ Configures underlying **firewall engine** with firewall rules

Firewall engine is typically integrated on OS level
– Linux: iptables
– QNX: pfilter
– Proprietary firewall engines also possible

Interfaces of ara::fw
– Setting the vehicle state
– Raising security events

Firewall in AUTOSAR

Standardized filter rule configuration

Challenge

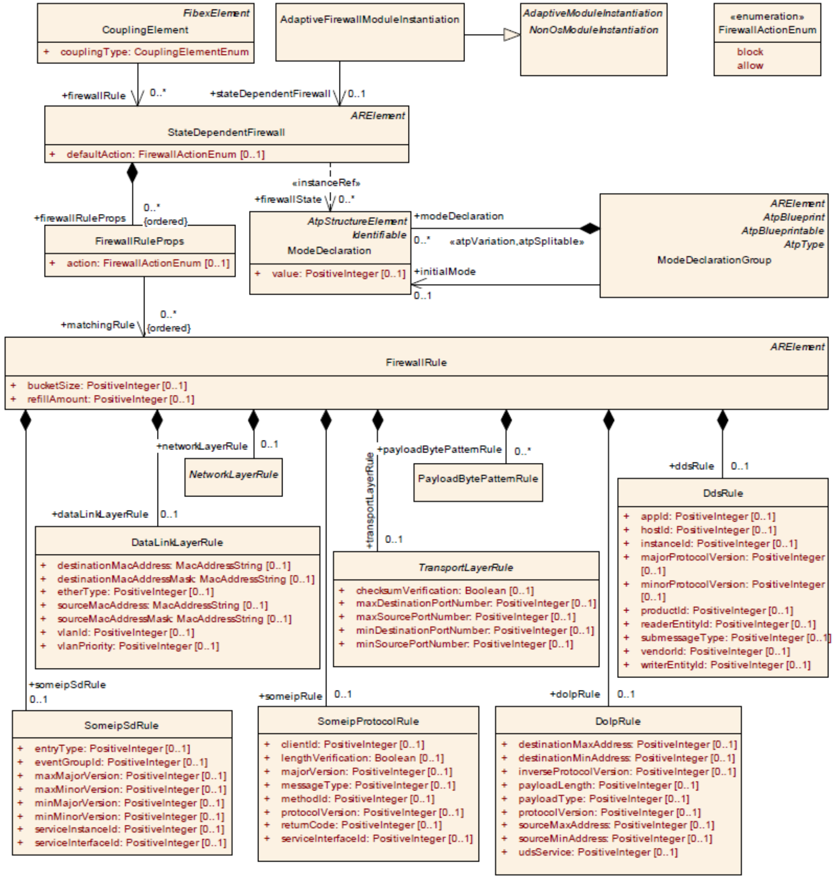
- No common firewall configuration scheme
- High effort for harmonizing OEM requirement with firewall configuration
- Requirements translation process prone to errors

AUTOSAR firewall solution

- Introduce common language for configuring firewalls
- Standardized ARXML exchange format
- AUTOSAR tooling support allows for easy allowlist generation from communication matrix

Firewall configuration language defined in the AUTOSAR manifest specification as UML

→ **Let‘s have a detailed look**



FibexElement
CouplingElement
+ couplingType: CouplingElementEnum

AdaptiveFirewallModuleInstantiation

AdaptiveModuleInstantiation
NonOsModuleInstantiation

«enumeration»
FirewallActionEnum
block
allow

ARElement
StateDependentFirewall
+ defaultAction: FirewallActionEnum [0..1]

FirewallRuleProps
+ action: FirewallActionEnum [0..1]

AtpStructureElement
Identifiable
ModeDeclaration
+ value: PositiveInteger [0..1]

ARElement
AtpBlueprint
AtpBlueprintable
AtpType
ModeDeclarationGroup

ARElement
FirewallRule
+ bucketSize: PositiveInteger [0..1]
+ refillAmount: PositiveInteger [0..1]

NetworkLayerRule

PayloadBytePatternRule

DataLinkLayerRule
+ destinationMacAddress: MacAddressString [0..1]
+ destinationMacAddressMask: MacAddressString [0..1]
+ etherType: PositiveInteger [0..1]
+ sourceMacAddress: MacAddressString [0..1]
+ sourceMacAddressMask: MacAddressString [0..1]
+ vlanId: PositiveInteger [0..1]
+ vlanPriority: PositiveInteger [0..1]

TransportLayerRule
+ checksumVerification: Boolean [0..1]
+ maxDestinationPortNumber: PositiveInteger [0..1]
+ maxSourcePortNumber: PositiveInteger [0..1]
+ minDestinationPortNumber: PositiveInteger [0..1]
+ minSourcePortNumber: PositiveInteger [0..1]

DdsRule
+ appId: PositiveInteger [0..1]
+ hostId: PositiveInteger [0..1]
+ instanceId: PositiveInteger [0..1]
+ majorProtocolVersion: PositiveInteger [0..1]
+ minorProtocolVersion: PositiveInteger [0..1]
+ productId: PositiveInteger [0..1]
+ readerEntityId: PositiveInteger [0..1]
+ submessageType: PositiveInteger [0..1]
+ vendorId: PositiveInteger [0..1]
+ writerEntityId: PositiveInteger [0..1]

SomeipSdRule
+ entryType: PositiveInteger [0..1]
+ eventGroupId: PositiveInteger [0..1]
+ maxMajorVersion: PositiveInteger [0..1]
+ maxMinorVersion: PositiveInteger [0..1]
+ minMajorVersion: PositiveInteger [0..1]
+ minMinorVersion: PositiveInteger [0..1]
+ serviceInstanceId: PositiveInteger [0..1]
+ serviceInterfaceId: PositiveInteger [0..1]

SomeipProtocolRule
+ clientId: PositiveInteger [0..1]
+ lengthVerification: Boolean [0..1]
+ majorVersion: PositiveInteger [0..1]
+ messageType: PositiveInteger [0..1]
+ methodId: PositiveInteger [0..1]
+ protocolVersion: PositiveInteger [0..1]
+ returnCode: PositiveInteger [0..1]
+ serviceInterfaceId: PositiveInteger [0..1]

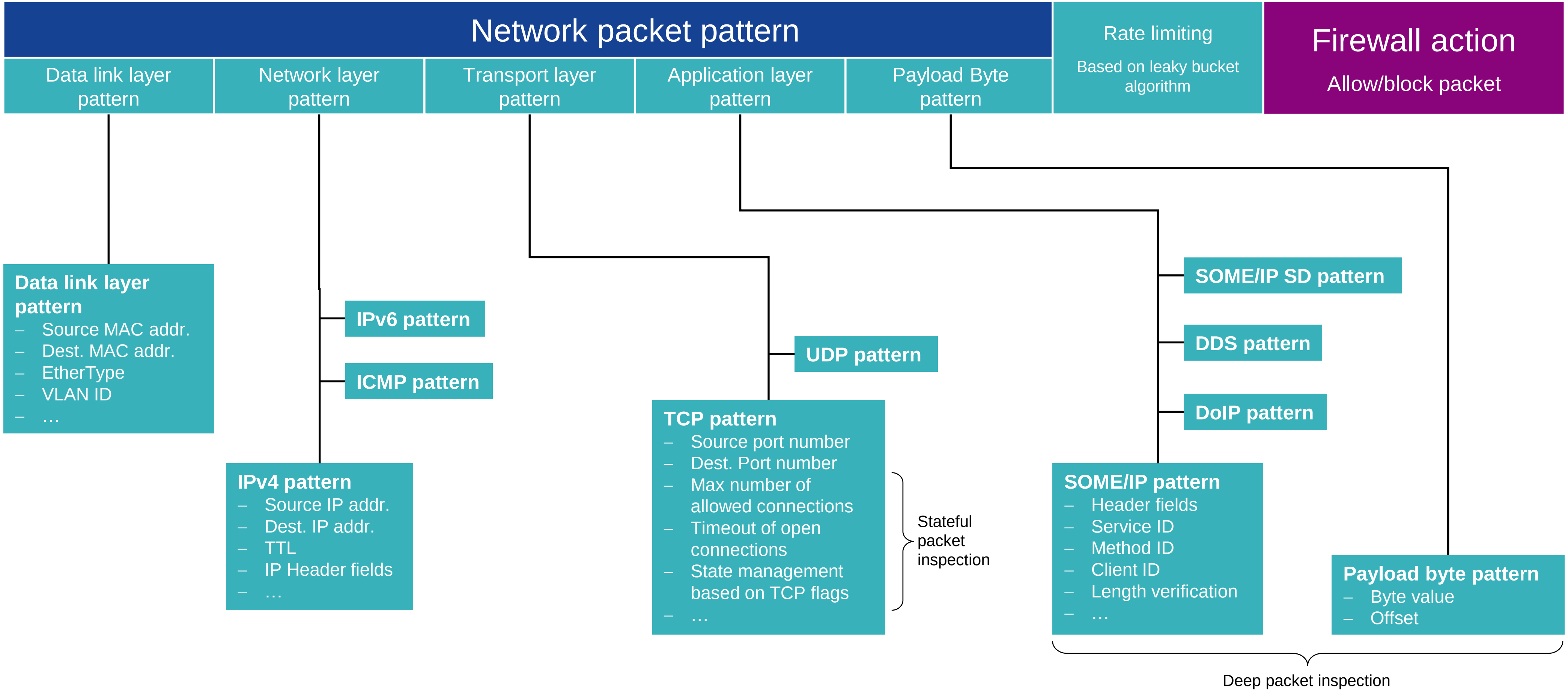
DoIpRule
+ destinationMaxAddress: PositiveInteger [0..1]
+ destinationMinAddress: PositiveInteger [0..1]
+ inverseProtocolVersion: PositiveInteger [0..1]
+ payloadLength: PositiveInteger [0..1]
+ payloadType: PositiveInteger [0..1]
+ protocolVersion: PositiveInteger [0..1]
+ sourceMaxAddress: PositiveInteger [0..1]
+ sourceMinAddress: PositiveInteger [0..1]
+ udsService: PositiveInteger [0..1]

Figure 10.49: Modeling of the Firewall

Source: AUTOSAR Specification of Manifest

Firewall in AUTOSAR

Standardized filter rule configuration



Data link layer pattern
- Source MAC addr.
- Dest. MAC addr.
- EtherType
- VLAN ID
- …

IPv4 pattern
- Source IP addr.
- Dest. IP addr.
- TTL
- IP Header fields
- …

TCP pattern
- Source port number
- Dest. Port number
- Max number of allowed connections
- Timeout of open connections
- State management based on TCP flags
- …

SOME/IP pattern
- Header fields
- Service ID
- Method ID
- Client ID
- Length verification
- …

Payload byte pattern
- Byte value
- Offset

Firewall in AUTOSAR

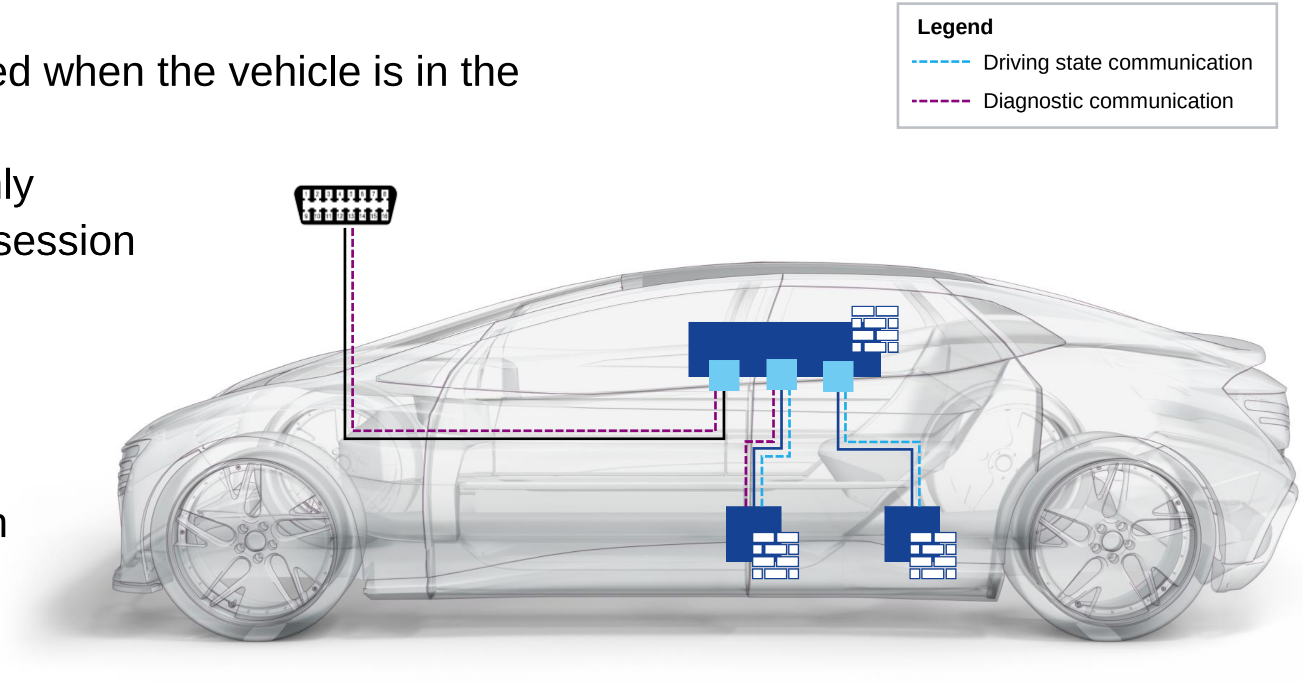
Vehicle state dependent filtering

Challenge

- Network traffic depends strongly on vehicle state
  - e.g. driving, parking, in a diagnostic session
- Specific network packets should only be allowed when the vehicle is in the correct state
- Example: Diagnostic communication should only be allowed when the vehicle is in a diagnostic session

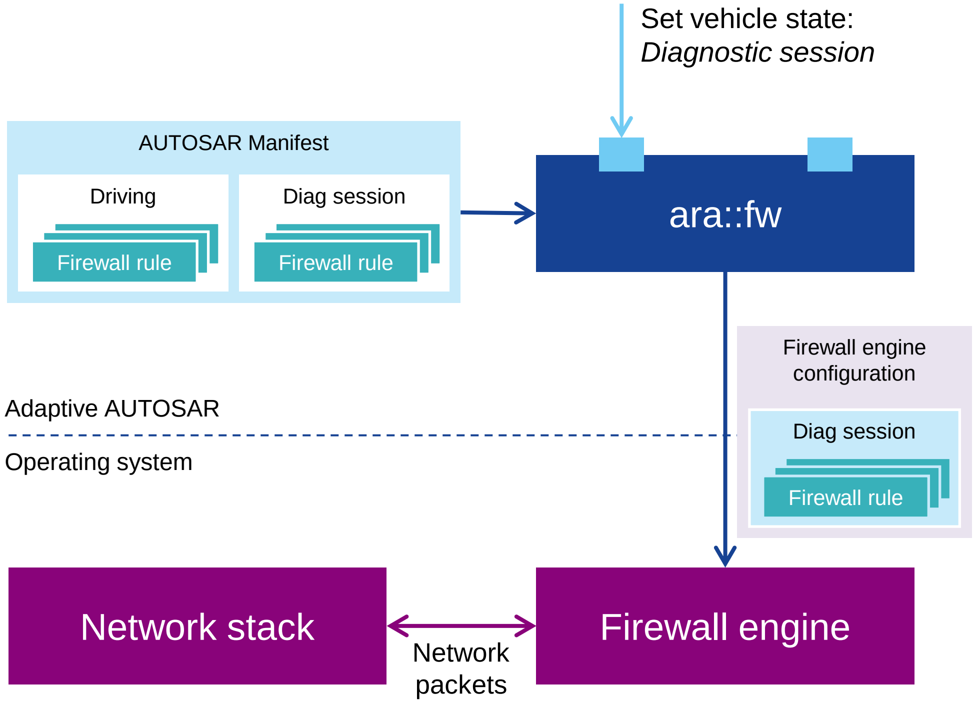
AUTOSAR firewall solution

- Define set of project-specific vehicle states
- Connect firewall rules to vehicle states
- Allow switching of vehicle states via application



Firewall in AUTOSAR

Vehicle state dependent filtering



How does the firewall accomodate state switches?

– Multiple firewall rules can be grouped in firewall vehicle states

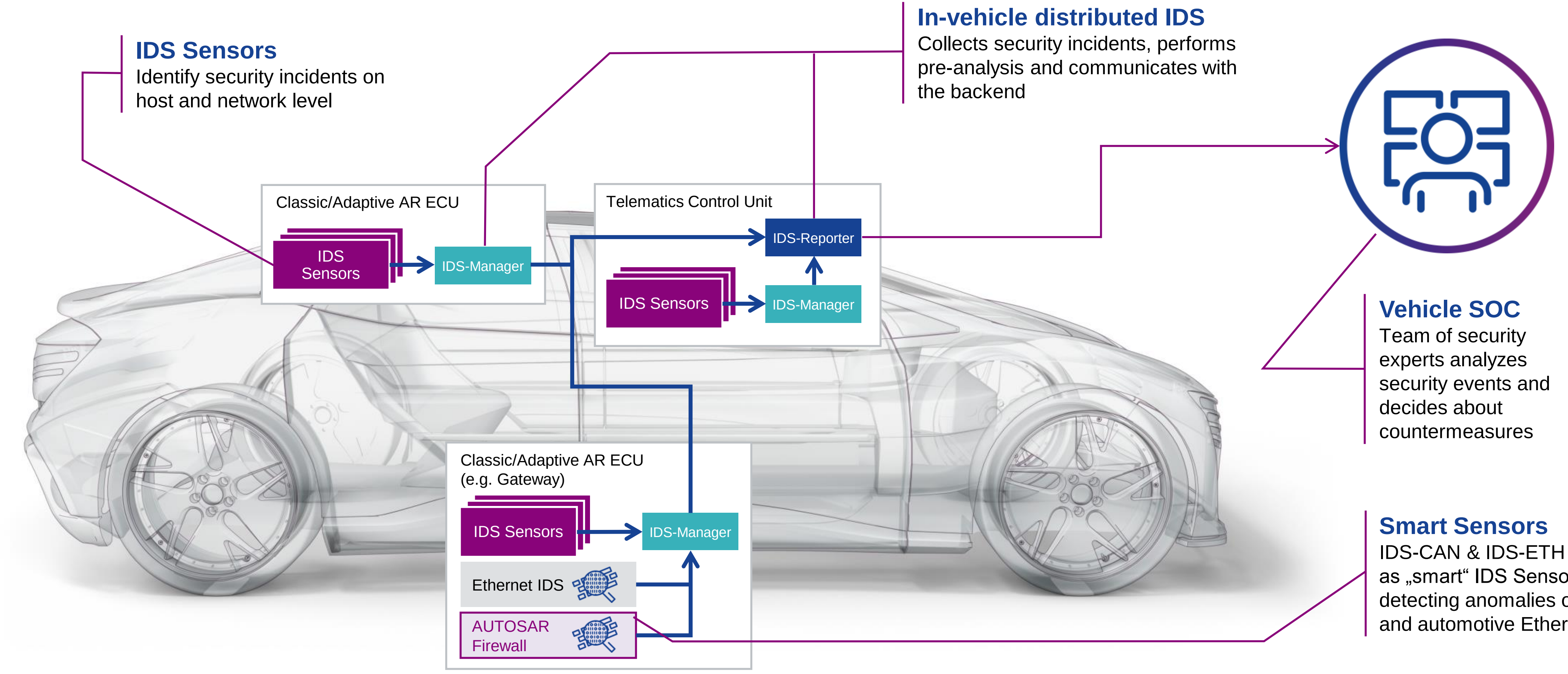
– An application can switch between different states using the `ara::fw::FirewallStateSwitchInterface`

– ara::fw updates the firewall engine configuration on the fly

Important: Vehicle states are not standardized, but can be defined by every user according to their needs

Firewall in AUTOSAR

Connection to the IDPS ecosystem



IDS Sensors
Identify security incidents on host and network level

In-vehicle distributed IDS
Collects security incidents, performs pre-analysis and communicates with the backend

Vehicle SOC
Team of security experts analyzes security events and decides about countermeasures

Smart Sensors
IDS-CAN & IDS-ETH function as „smart“ IDS Sensors, detecting anomalies on CAN and automotive Ethernet/IP

Firewall in AUTOSAR

Connection to IDPS ecosystem

Challenge

Only few AUTOSAR-standardized security events available

→ OEMs define their own Ethernet security events

→ Non-uniform security events lead to high efforts in the VSOC

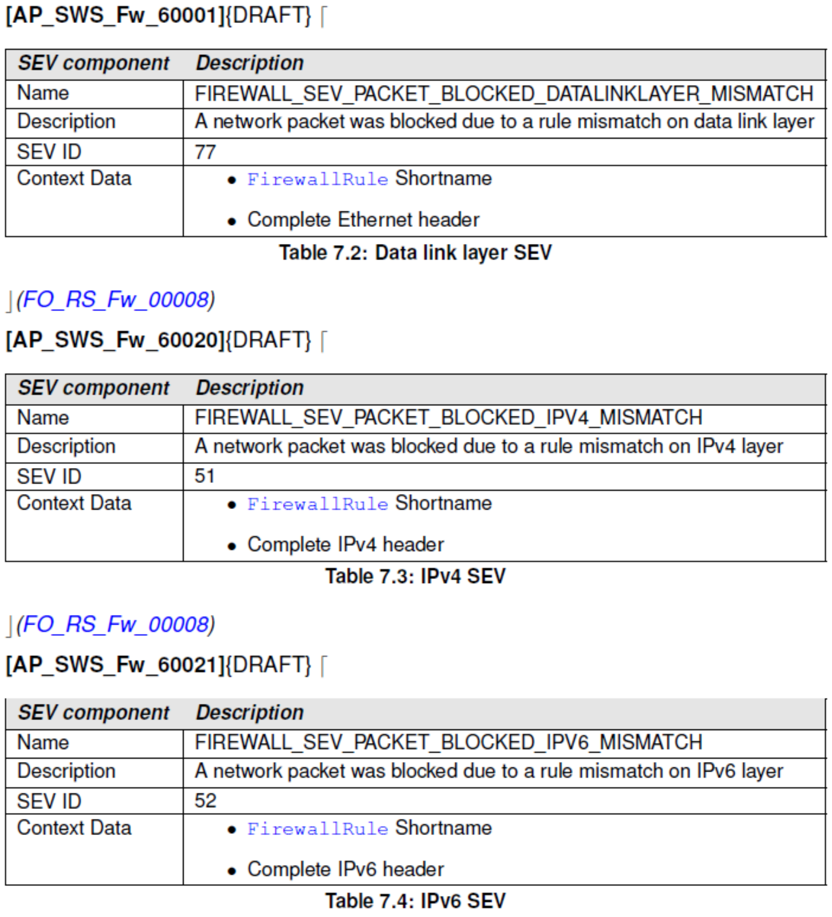
AUTOSAR firewall solution

- Provide standardized set of network security events
- Standardize associated context data for efficient analysis in VSOC

→ Uniform, standardized security event landscape

Result

- 15 new security events for the firewall defined
- Security events based on individual protocols and other firewall functionality (e.g. rate limit reached)
- Standardized context data: Network packet header provided as context data for analysis in VSOC



[AP_SWS_Fw_60001]{DRAFT} ⌈

SEV component	***Description***
Name	FIREWALL_SEV_PACKET_BLOCKED_DATALINKLAYER_MISMATCH
Description	A network packet was blocked due to a rule mismatch on data link layer
SEV ID	77
Context Data	• FirewallRule Shortname • Complete Ethernet header

Table 7.2: Data link layer SEV

⌋*(FO_RS_Fw_00008)*

[AP_SWS_Fw_60020]{DRAFT} ⌈

SEV component	***Description***
Name	FIREWALL_SEV_PACKET_BLOCKED_IPV4_MISMATCH
Description	A network packet was blocked due to a rule mismatch on IPv4 layer
SEV ID	51
Context Data	• FirewallRule Shortname • Complete IPv4 header

Table 7.3: IPv4 SEV

⌋*(FO_RS_Fw_00008)*

[AP_SWS_Fw_60021]{DRAFT} ⌈

SEV component	***Description***
Name	FIREWALL_SEV_PACKET_BLOCKED_IPV6_MISMATCH
Description	A network packet was blocked due to a rule mismatch on IPv6 layer
SEV ID	52
Context Data	• FirewallRule Shortname • Complete IPv6 header

Table 7.4: IPv6 SEV

Source: AUTOSAR Specification of Firewall in Adaptive Platform

Firewall standardization in AUTOSAR

What else is there to come?

Recap: Future zone-based E/E-architecture

Current status of firewall standardization

1 Firewall and IDS on Vehicle computer

- Network separation using VLANs
- Firewall cross domain traffic
- Firewall end-to-end traffic
- Deep packet inspection for some frames
- Intrusion detection for ethernet

2 Firewall on Ethernet switch

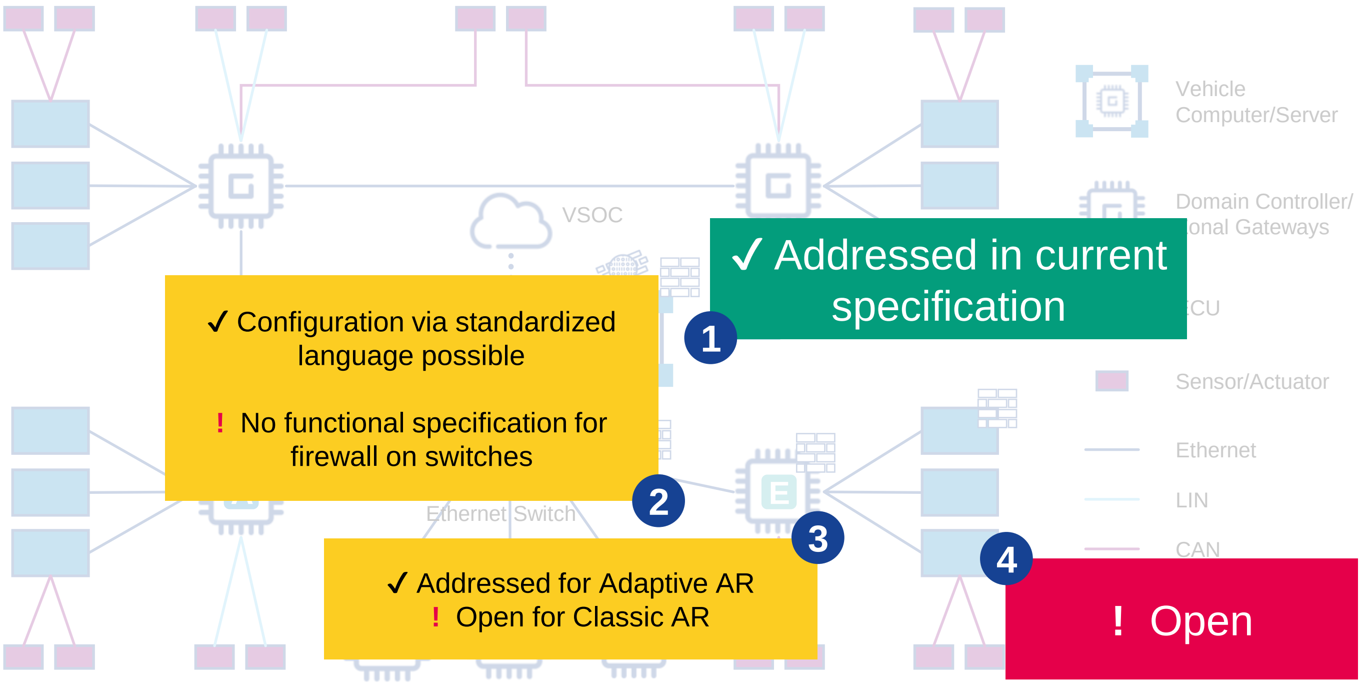
- Network separation using VLANs between domains A, B, C, D and E
- Firewall cross domain traffic at high speed between domains A, B, C, D and E
- Access control for vehicle server

3 Firewall and IDS on Domain controller

- Access Control and Firewall zonal traffic

4 Firewall on end ECUs

- Firewall for specific applications like EV charging ECU



✓ Addressed in current specification

✓ Configuration via standardized language possible

! No functional specification for firewall on switches

✓ Addressed for Adaptive AR
! Open for Classic AR

! Open

Firewall standardization in AUTOSAR

Outlook: Classic AUTOSAR

Current focus of work

Firewall standardization for Classic AUTOSAR

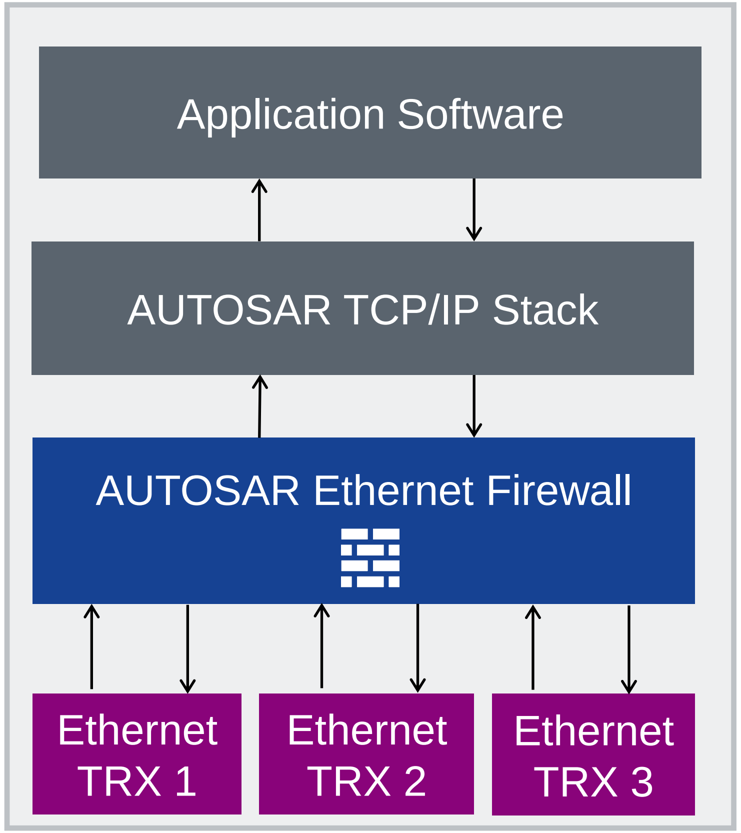
Goal

Same feature set as in Adaptive AUTOSAR

- Filtering of network traffic (stateless, stateful deep packet inspection)
- Re-usage of standardized firewall configuration language
- Dynamic firewall rules based on vehicle state
- Security events raised by firewall

Release timeline

Next AUTOSAR release R23-11



Firewall standardization in AUTOSAR

Outlook: Firewall on switches

Modern switches with dedicated CPU can run AUTOSAR

– Allows re-usage of existing AUTOSAR modules

– Allows leveraging of AUTOSAR tooling support

→ The AUTOSAR firewall specification shall also support the deployment on switches

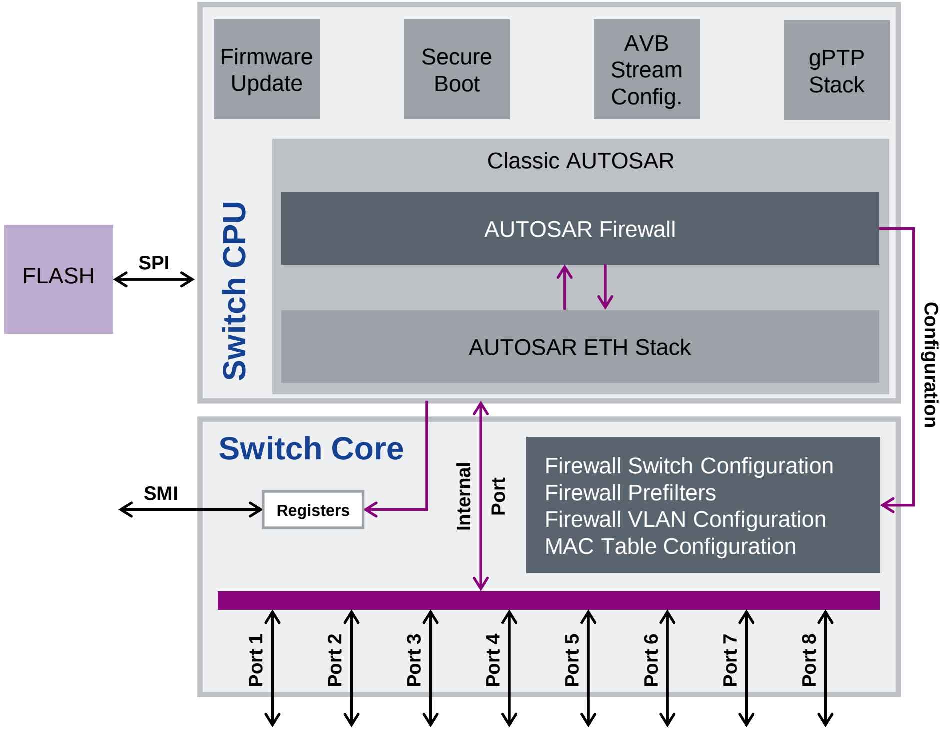
Additional features for switch deployment

– Configuration of filtering mechanisms in switch core (e.g. (T)CAM rules)

– Extension of firewall configuration language to include (T)CAM rule configuration

Release timeline

Next AUTOSAR release R23-11



Firewall standardization in AUTOSAR

Summary/Conclusion

Increasing **need for firewall** in automotive, but deployment oftentimes cumbersome

– High-effort alignment process, prone to errors

AUTOSAR firewall standardization addresses this issue by specifying a **common language for firewall configuration**

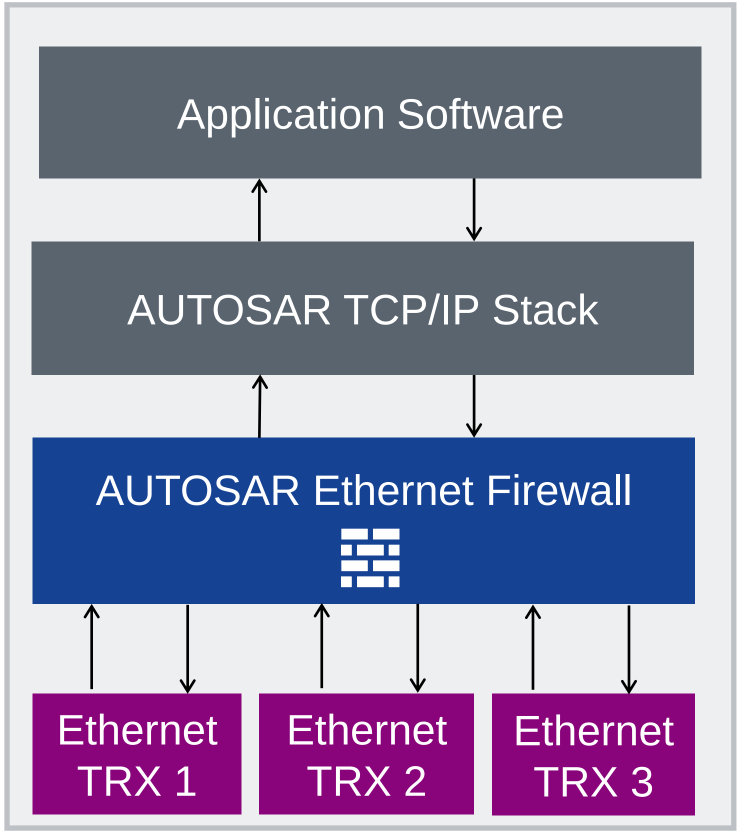
Additional **firewall features**

– Stateless, stateful and deep packet inspection

– Filtering based on vehicle state

– Standardized security events for IDS

Specification available for Adaptive AUTOSAR, **Classic AUTOSAR and switches are planned for the next release R23-11**



Thank you!