NDSS 2026

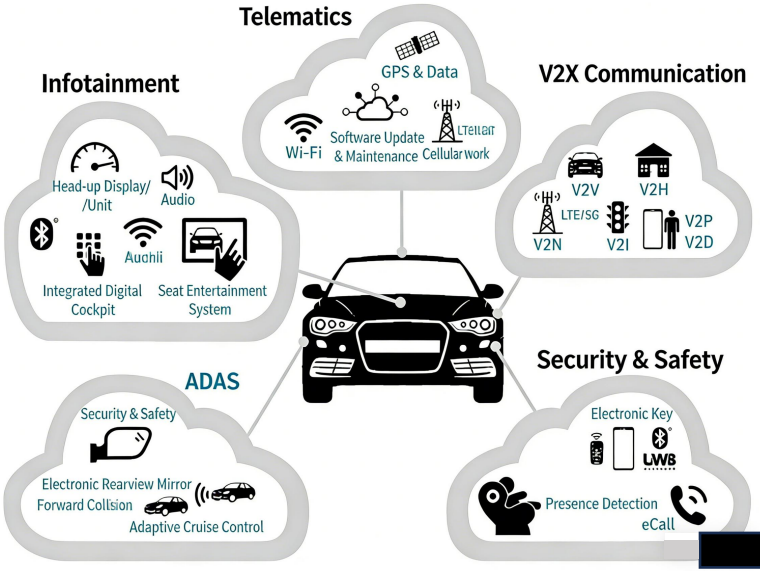
# Automating Function-Level TARA for Automotive Full-Lifecycle Security

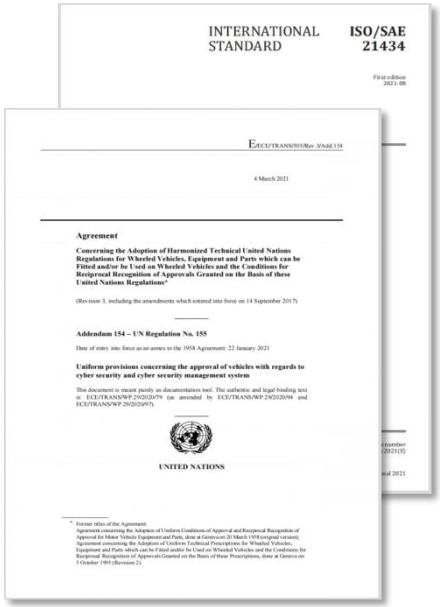
Yuqiao Yang, Yongzhao Zhang, Wenhao Liu, Jun Li, Pengtao Shi, DingYu Zhong, Jie Yang*, Ting Chen*, Jun Li, Sheng Cao, Yuntao Ren, Yongyue Wu, Xiaosong Zhang

电子科技大学
University of Electronic Science and Technology of China

犬安科技
GoGoByte

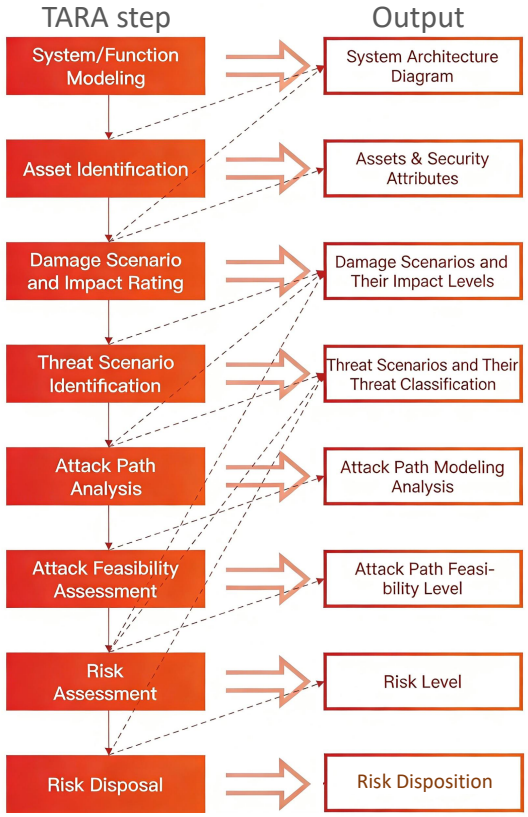
# Threat Analysis and Risk Assessment (TARA)



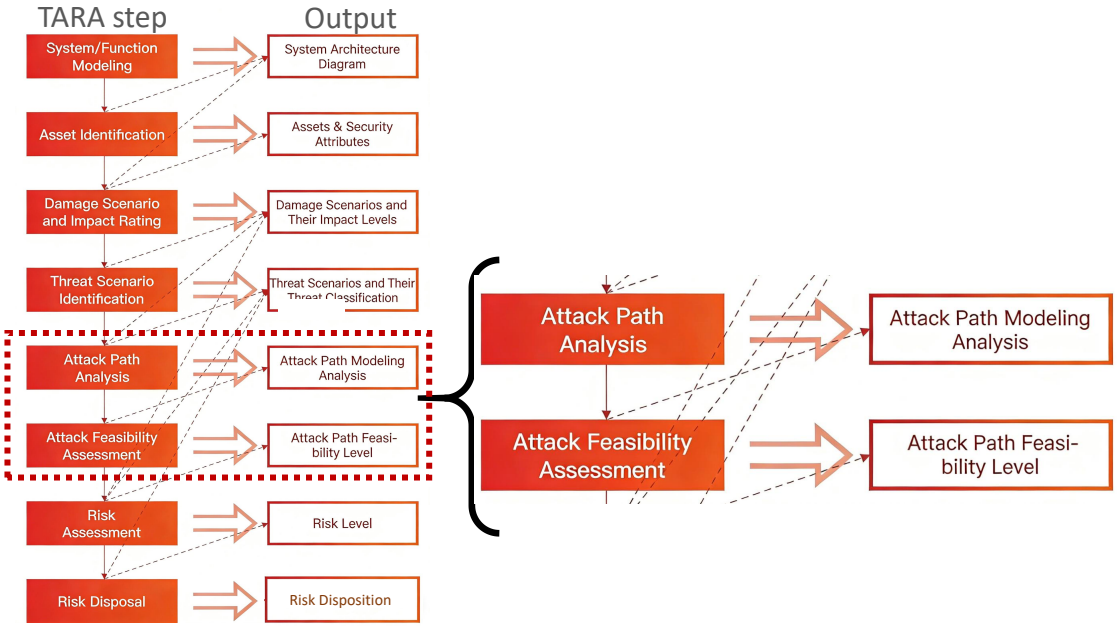

**Complex** intelligent auto parts, ever-expanding attack surface.

OEMs and suppliers conduct TARA per **regulatory and standard requirements**.




**TARA**

# Background: Industry Limitations



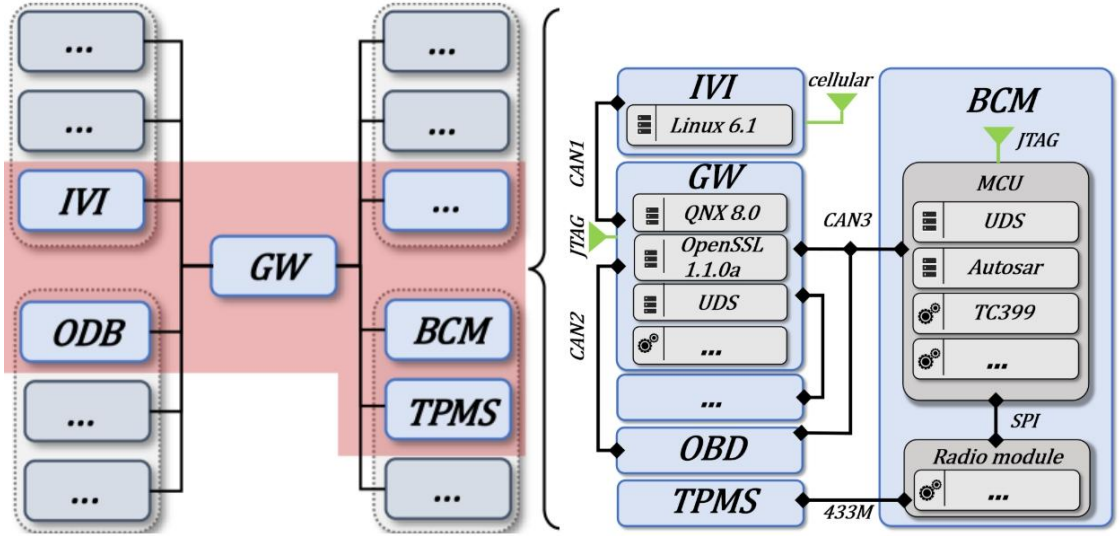

A TARA report takes a team **2–5 weeks** to complete, with most effort spent on **attack path analysis** and evaluation.

based on some trade-offs very similar to an architectural ... be revisited and updated based on vulnerabilities found later during the de... times, updating the TARA is overlooked or avoided given the project/product de...

The time needed to perform a TARA analysis depends on the ... complexity of the item under consideration. For a component like a battery thermal manag... ystem or an item of low complexity, it can take between 2-5 weeks. A TARA activity might involve m... le team members and is not done by a single individual. Ideally, a team composed of product/domain experts, cybersecurity engineers, cybersecurity managers, and HW/SW engineers are all involved in a TARA analysis. A team with diverse experience and cross-functional domain knowledge would be most relevant to do a TARA. One needs to be cautious of scope creep while doing TARA and understand that version one is better than version none. The TARA may evolve as you discuss along with your team and as the team matures to perform such analysis. Another very common challenge some customers face is to do a TARA as a part of an agile development sprint. It may not be possible to complete a TARA analysis in one sprint. Also, it may be possible that the functional requirements of the item are not clear while performing a TARA which may lead to incomplete TARA.

**Current TARA is manual, time-consuming, and hard to scale, requiring automation.**

# Vehicle-level TARA vs. Function-level TARA




**Vehicle-level TARA:**

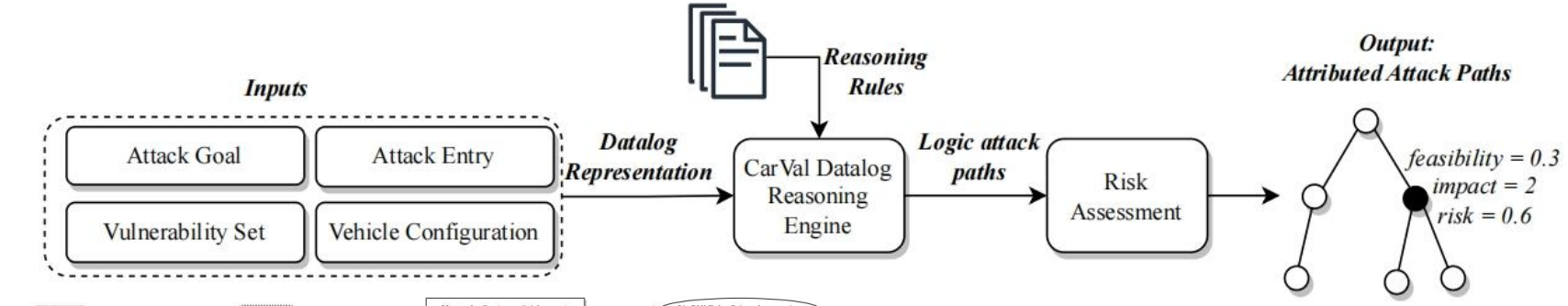
Assumes **attackers cannot access** internal components — **coarse-grained.**

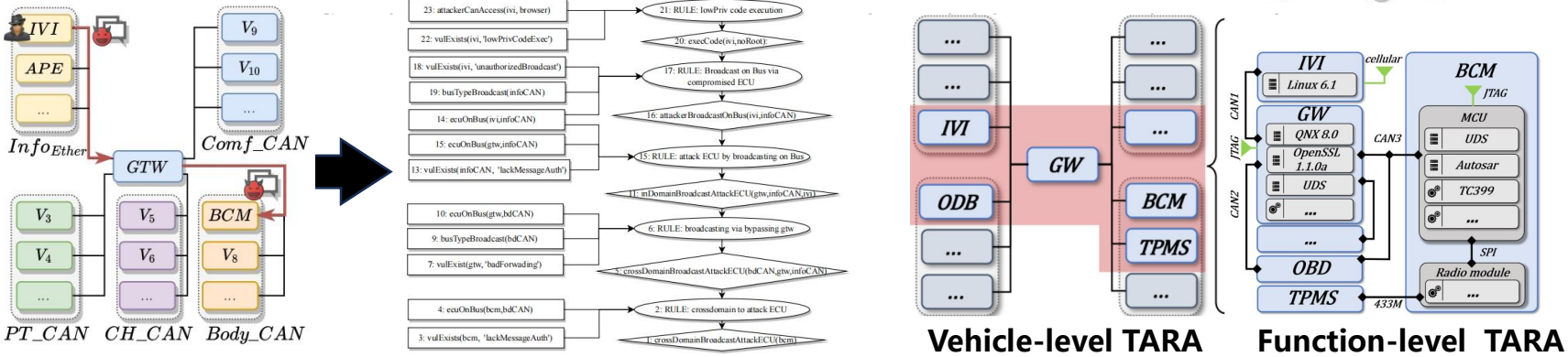
**Function-level TARA:**

Assumes **attackers can access** internal components — **fine-grained.**

**Function-level TARA is more fine-grained and more challenging than Vehicle-level TARA.**

# A promising attempt: based on data logs.



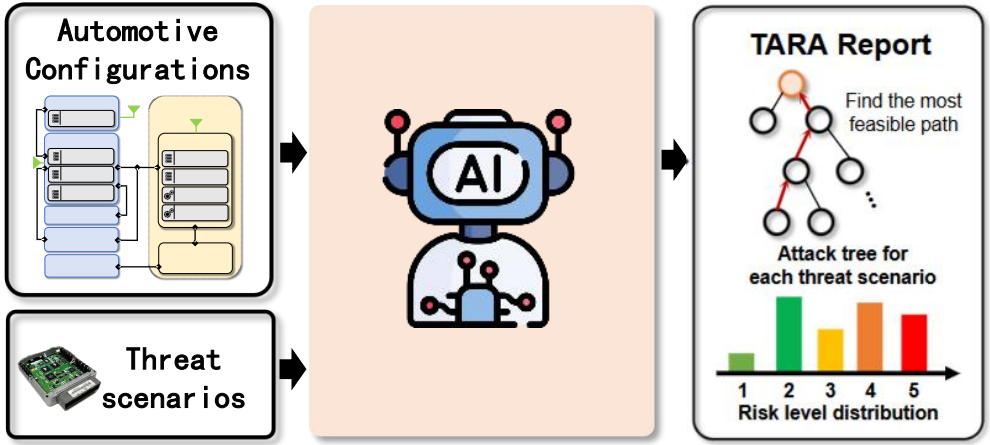




**log-based automation tools are not suitable for Function-level TARA**

[1] *Jing, Pengfei, et al. "Revisiting automotive attack surfaces: A practitioners' perspective." 2024 IEEE Symposium on Security and Privacy (SP 24).*

# Approach and Framework Diagram

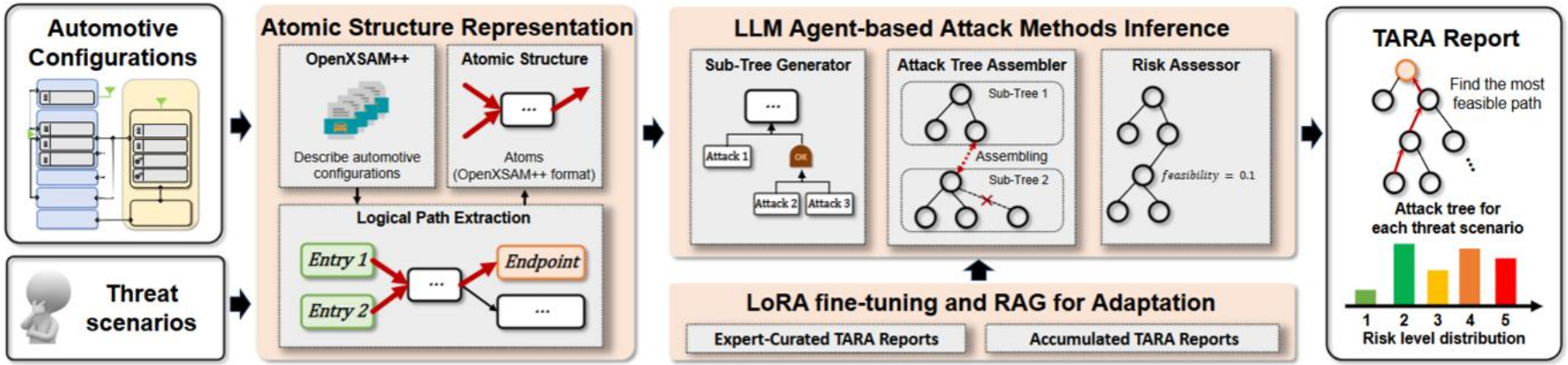
## Framework of DefenseWeaver




- Users only need to input automotive configurations and threat scenarios to automatically perform Function-level TARA analysis and generate a report.

# Approach and Framework Diagram

## Framework of DefenseWeaver

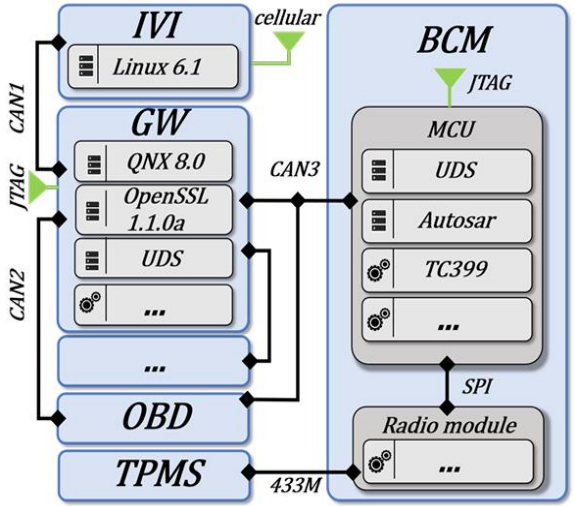



- Users only need to input **automotive configurations** and **threat scenarios** to **automatically** perform **Function-level** TARA analysis and generate a report.

- When components are **updated**, **simply updating** the vehicle configuration enables rapid re-execution of component-level TARA, **ensuring lifecycle security**

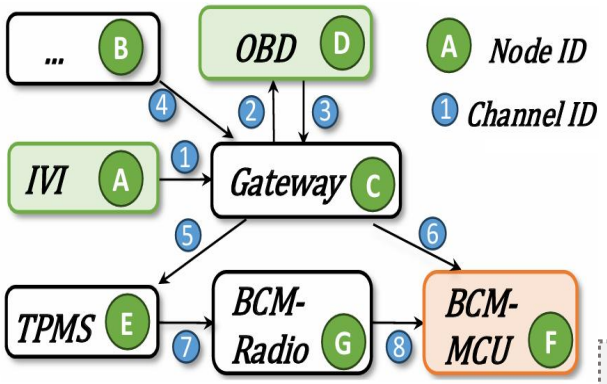
# 1.Structured description of configuration

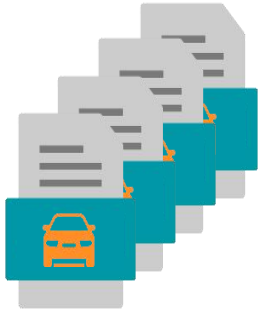
Function-level Automotive Configurations




Automotive configurations are usually diagram-based and **hard for LLMs to parse.**

Structured description of automotive configurations



OpenXSAM++

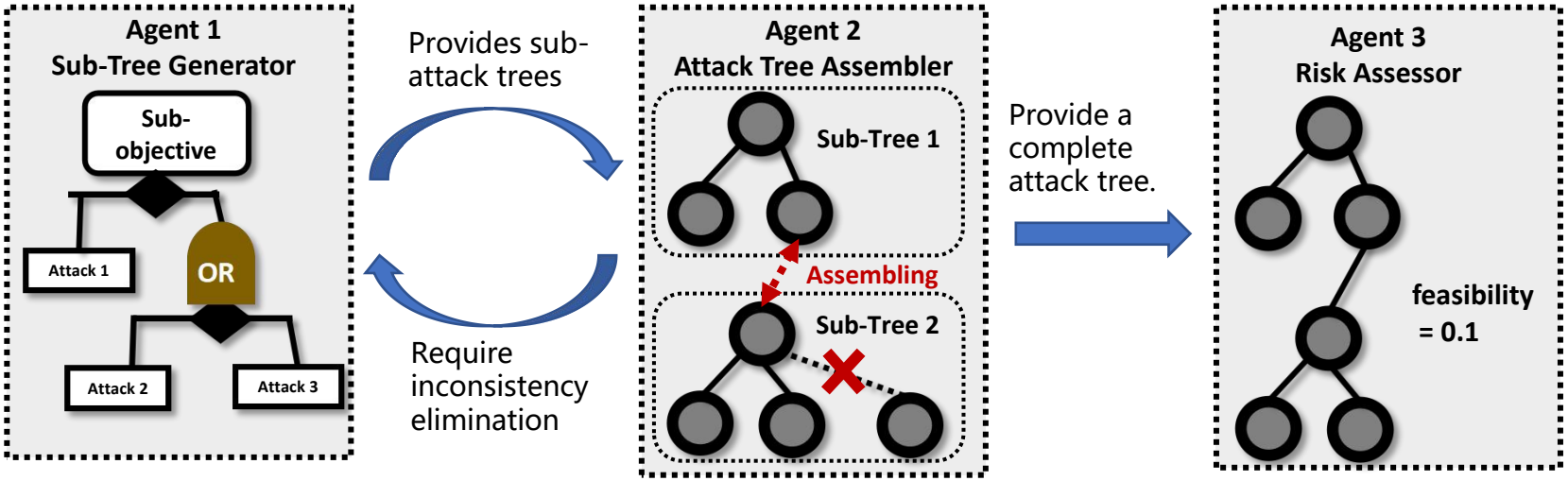
Design an XML-based **OpenXSAM++** file to **structurally describe** component software, hardware, interfaces, channels, and interconnections.

**Provide a structured Function-level representation to enable comprehensive LLM understanding.**

# 2. Multi-agent collaboration

**Complex attack path analysis and evaluation tasks lead to scattered attention in LLM**

**Multi-agent collaboration simplifies and breaks down tasks**



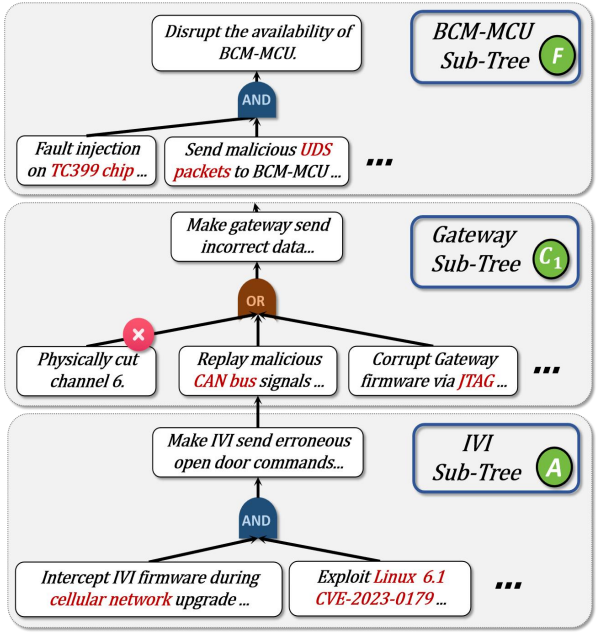

**Generate local attack paths.**

**Construct a complete attack tree.**

**Perform a comprehensive attack risk assessment.**

**Use multi-agent collaboration to reduce complexity and prevent attention dilution in a single LLM.**

# 3. Adaptive Optimization




LLMs are not tailored for TARA

**LoRA:** **Corrected subtrees are used for LoRA fine-tuning to learn expert logic.**


**Data sources**

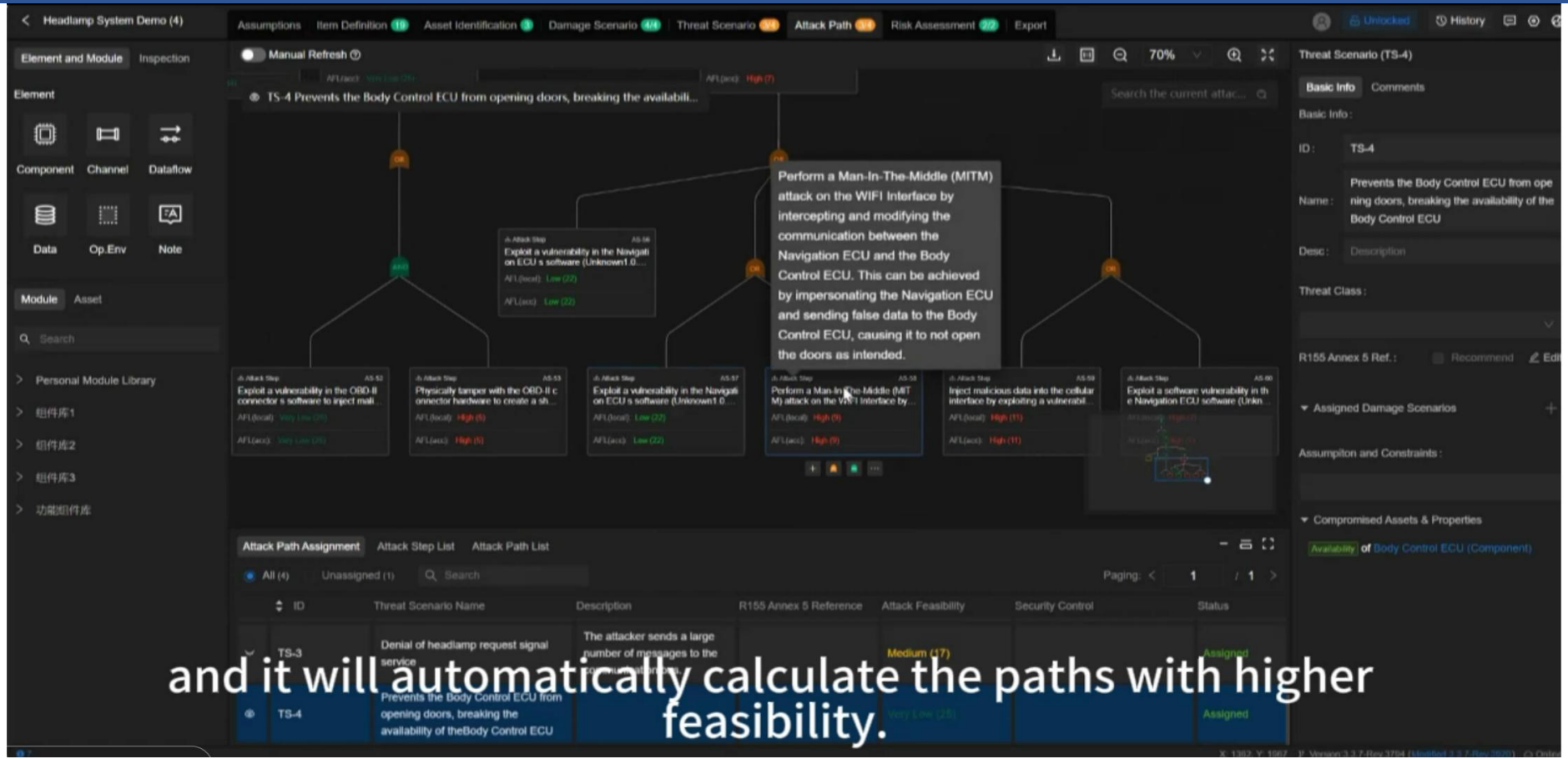
**RAG:** **Updated scores are instantly integrated into RAG for evaluation**

| | Company A | Company B |
|---|---|---|
| Time Cost | ≤1 Week | ≤1 Month |
| Professional Skills | Expert | Expert |
| Required Information | Public Information | Restricted Information |
| Opportunity Window | Unlimited | Medium |
| Required Equipment | Professional | Professional |
| Attack Feasibility | High | Low |

Evaluation criteria vary across companies.

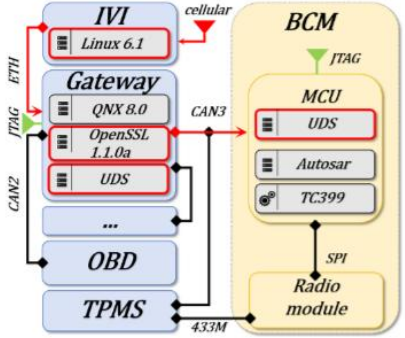
**LoRA is more suitable for TARA analysis;**
**RAG is better for meeting diverse evaluation criteria.**

# Live Demo

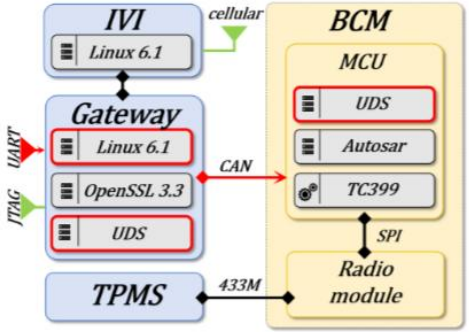



and it will automatically calculate the paths with higher feasibility.

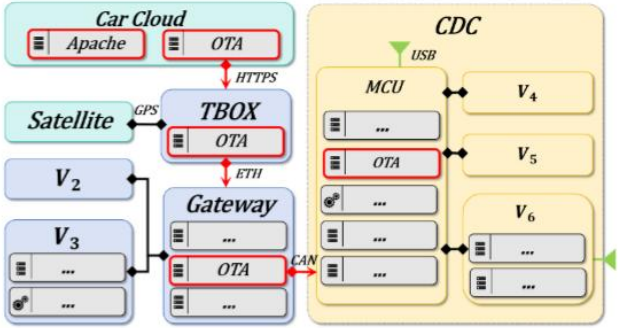
# Evaluation



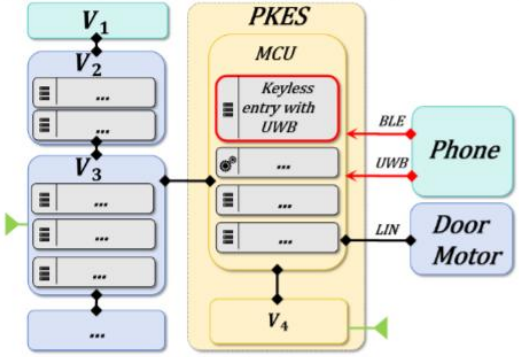

(a) Body Control Moudle (Car $A$)



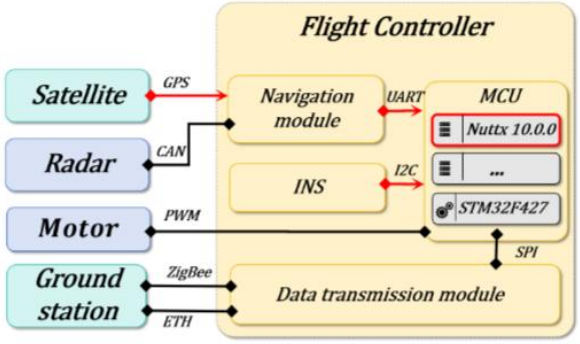

(b) Body Control Moudle (Car $B$)



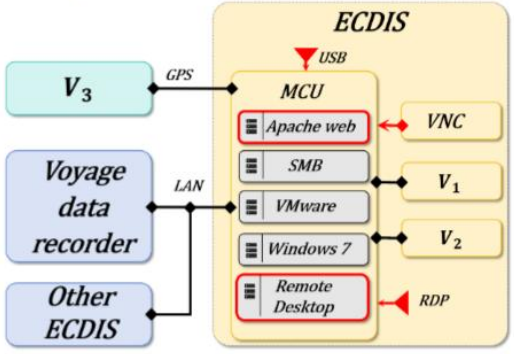

(c) Cockpit Domain Controller (Car $C$)




(d) Passive Keyless Entry and Start (Car $D$)



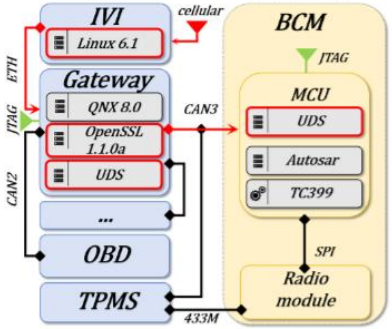

(e) PX4 (UAV)



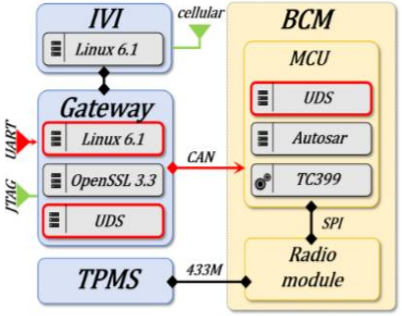

(f) ECDIS (Ship)

**11 real attack paths found across 6 systems.**

# Case 1： Body Control Module (BCM)

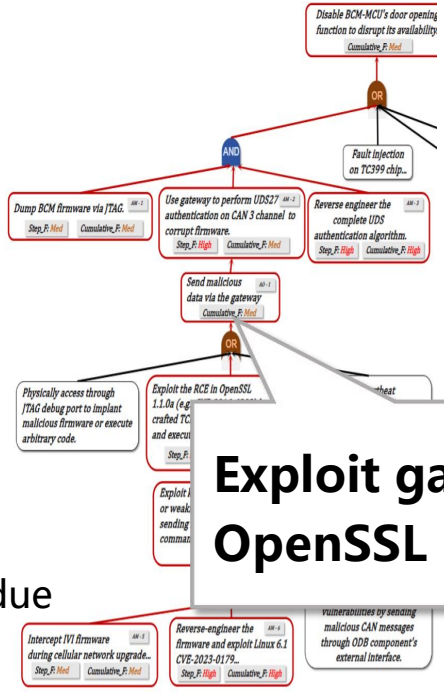



BCM Car A

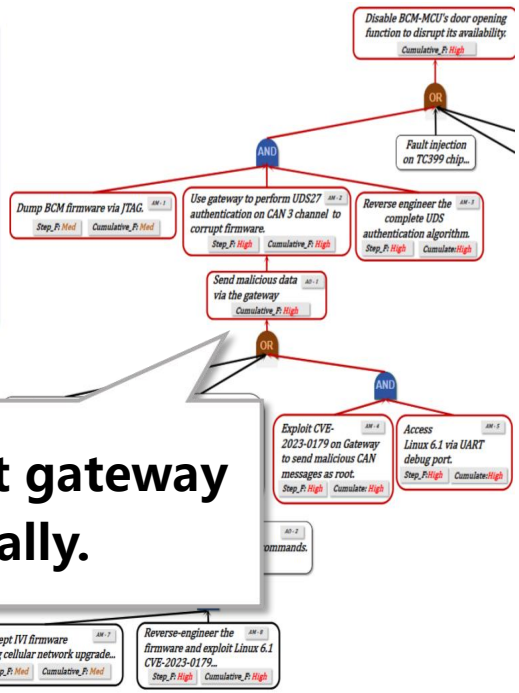



BCM Car B

**Insight:**
**Same logic** (IVI-GW-BCM)
, results in **different** concrete attack paths due to function-level differences.



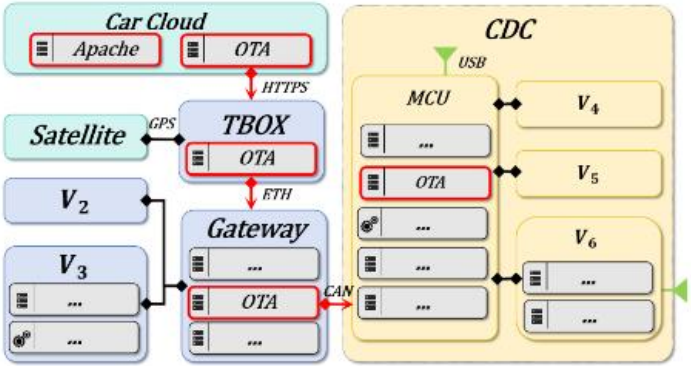

**Exploit gateway OpenSSL remotely.**

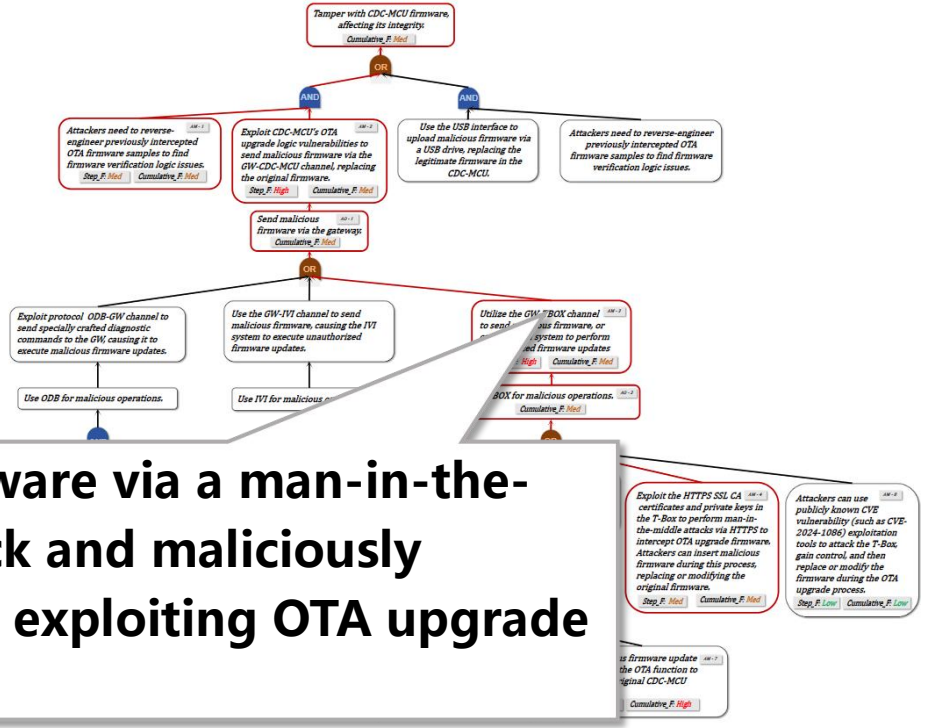



**Exploit gateway OS locally.**

**Generate configuration-specific attack paths.**

# Case 2: Cockpit Domain Controller (CDC)



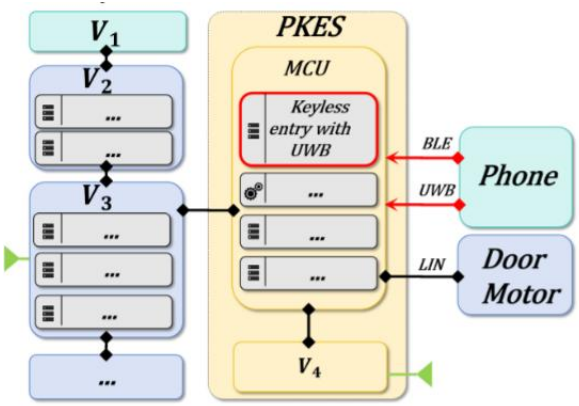




**Insight:**
Inject malicious code into core automotive systems via **OTA** updates delivered through the vehicle cloud platform.

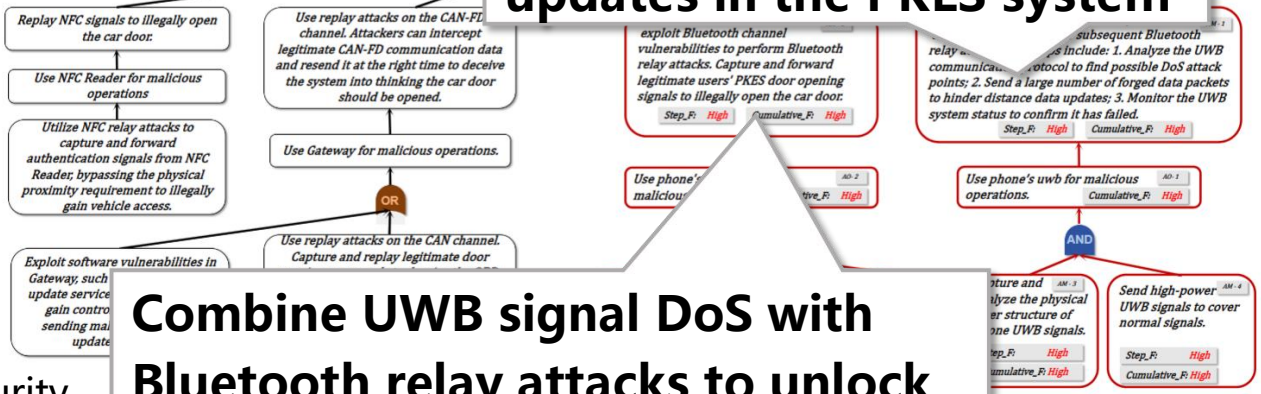
**Obtain firmware via a man-in-the-middle attack and maliciously replace it by exploiting OTA upgrade logic flaws.**

**Cover peripherals and cloud vectors.**

# Case 3: Passive Keyless Entry System (PKES)




**Use jamming attacks to prevent UWB distance updates in the PKES system**

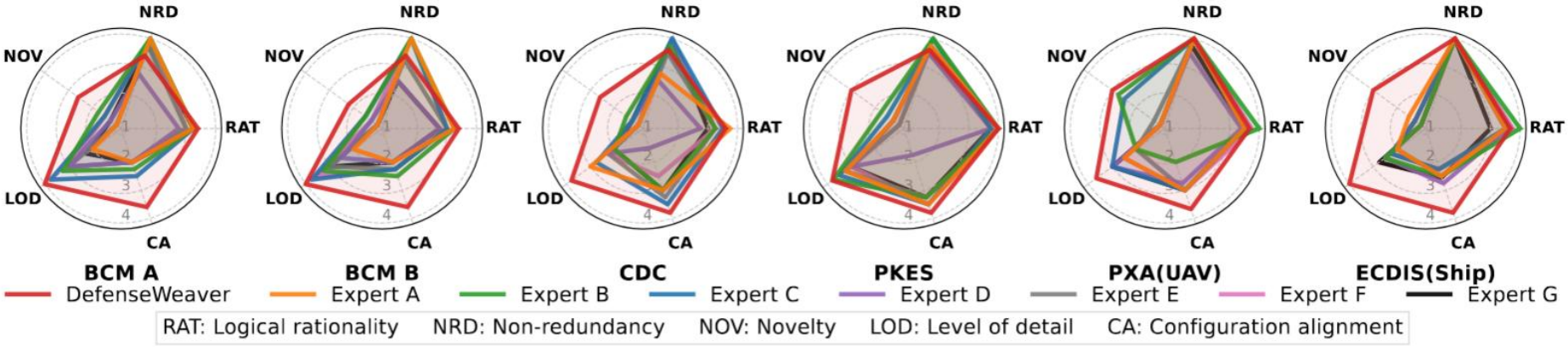



**Combine UWB signal DoS with Bluetooth relay attacks to unlock or start the vehicle illegally.**

**Insight:**
**Ultra-Wideband (UWB)** is designed to enhance PKES security, but when combined with relay attacks on other channels, it can itself become an attack entry point.。

**Integrate multiple techniques into coherent and novel attack paths.**

# Defenseweaver vs. Human Expert




**DefenseWeaver:**

- Generate **novel** and **comprehensive** **paths**
- Remove subjective **bias**.
- Configuration-**specific** paths.

**VS**

**Human Experts:**

- Struggle with **new** configurations.
- **Subjective assumptions** introduce errors.。
- Miss **subtle** system differences.

**Higher-quality and broader attack trees.**

# Applications

**OEM:** Xiaomi, BESTUNE, AVATR

**Tier One:** UAES, iFLYTEK

**Testing Bodies:** 招商车研 CMVR (检 致 精 · 行 致 远), CEPREI




# Thank you！

chenting19870201@163.com