OMNIAIR®
CONSORTIUM
HACKATHON

Inaugural

OmniAir Hackathon

Sponsored by

OmniTrust™

June 3, 2026 | **8:30 - 3:30 pm** | **Johns Hopkins University at Mt. Washington**

OmniAir Consortium® is hosting its first hackathon for connected vehicles. This event will explore a critical capability needed to ensure trust and security in connected vehicle deployment: Misbehavior Detection and Prevention in V2X systems.

Register Now!

Open to all **University Teams**
Students can present a live or recorded demonstration
Applications/ registration is due by March 27
Email questions to: info@omniair.org

Register at https://bit.ly/hckathn26 by March 27th to participate in the challenge.

Disruptive Misbehavior in V2X Systems

Vehicle-to-Everything (V2X) technologies depend on trusted, standards-based communication to deliver safety and mobility benefits. However, even when messages are valid, authenticated, and standards-compliant, systems may still behave in unexpected, unsafe, or disruptive ways.

This challenge invites teams to creatively design and demonstrate V2X misbehavior. Successful teams will demonstrate situations where a V2X system reacts incorrectly to information that appears legitimate. The objective is to highlight the value of misbehavior detection and stimulate discussion on future countermeasures.

Challenge Objective:

1. Demonstrate a disruptive V2X misbehavior
2. Show the real-world impact of that misbehavior
3. Propose countermeasures that could detect or prevent it in future systems
4. Misbehavior should be standards-aware, realistic, and technically credible.

Rules of the Hackathon/Game:

1. Each team must have a designated faculty supervisor affiliated with the university or entity sponsoring the team.
2. Each team must consist of 2-5 participants. A great target number is three.
3. Each team will have an OmniAir member mentor. Mentors will meet with teams virtually for up to 10 hours a month – approximately 2 hours a week. The teams should report to this mentor on milestones and objectivity.

What Is "Misbehavior"?

For this challenge, misbehavior is defined as: The generation, use, or interpretation of valid V2X messages that cause unexpected, unsafe, or undesirable system behavior, without breaking cryptography or using invalid certificates.

Misbehavior may be:

1. Intentional, accidental, or the result of misconfiguration
2. Emergent from interactions between multiple systems

Technical Scope: Teams may work with:

1. Vehicles (real, test, or simulated)
2. Intersections or roadside units (RSUs)
3. Network V2X (cloud-assisted or V2N)
4. SAE J2735 messages (e.g., BSM, SPaT, MAP, TIM, PSM, SDSM)

Out of scope:

1. Breaking cryptography or SCMS
2. Forging certificates
3. RF jamming or denial-of-service attacks

Why This Challenge Matters:

This challenge demonstrates that secure messages do not always guarantee safe behavior. By exposing gaps in today's systems, students help inform stronger, more resilient V2X deployments of the future.

Example Misbehavior Scenarios (Non-Exhaustive):

1. A vehicle causes false collision alerts or hard braking
2. An intersection broadcasts misleading but valid SPaT or MAP data
3. A work zone or traveler alert is shifted in location or time
4. Multiple participants reinforce a false but believable condition

Required Demonstration Elements: Each team must provide:

1. A live or recorded demonstration
2. A clear explanation of:
    - What messages were used
    - Why did the system accept them
    - Why was the outcome disruptive
    - A brief impact assessment (safety, operations, trust)
    - Implementation is optional, but proposals must be technically sound and realistic.

Countermeasure Requirement:

For every misbehavior, teams must propose at least one countermeasure; implementation is optional, but proposals must be technically sound and realistic.

Evaluation Criteria: Teams will be evaluated on:

1. Creativity and originality
2. Technical credibility
3. Demonstrated disruptive impact
4. Real-world relevance
5. Quality of proposed countermeasures
6. Clarity of explanation and presentation

This challenge demonstrates that secure messages do not always guarantee safe behavior. By exposing gaps in today's systems, students help inform stronger, more resilient V2X deployments of the future.

Progression Overview

Learn → Build → Explore → Experiment → Select → Demonstrate → Improve

Student Team Milestones

This milestone roadmap guides teams from initial learning to a successful live demonstration. Each milestone represents measurable progress toward a final disruptive misbehavior showcase.

1. Learn the Foundations Objective:

- Understand how V2X systems are designed to behave. Study key SAE J2735 message (BSM, SPaT, MAP, TIM, PSM)
- Understand IEEE 1609.2 message signing and trust validation
- Review how vehicles, RSUs, and network V2X interact
- Identify where behavioral assumptions exist beyond cryptographic trust

Outcome: Team can clearly explain how a valid V2X message becomes "trusted" by a receiving system.

2. Build the Test Environment Objective:

- Establish a working V2X experimentation platform.
- Configure message encoders/decoders or simulators
- Set up vehicle, RSU, or cloud test components
- Generate and successfully decode at least one standards-compliant message

Outcome: Team can generate, transmit, and observe valid J2735 messages

3. Explore Disruption Ideas Objective:

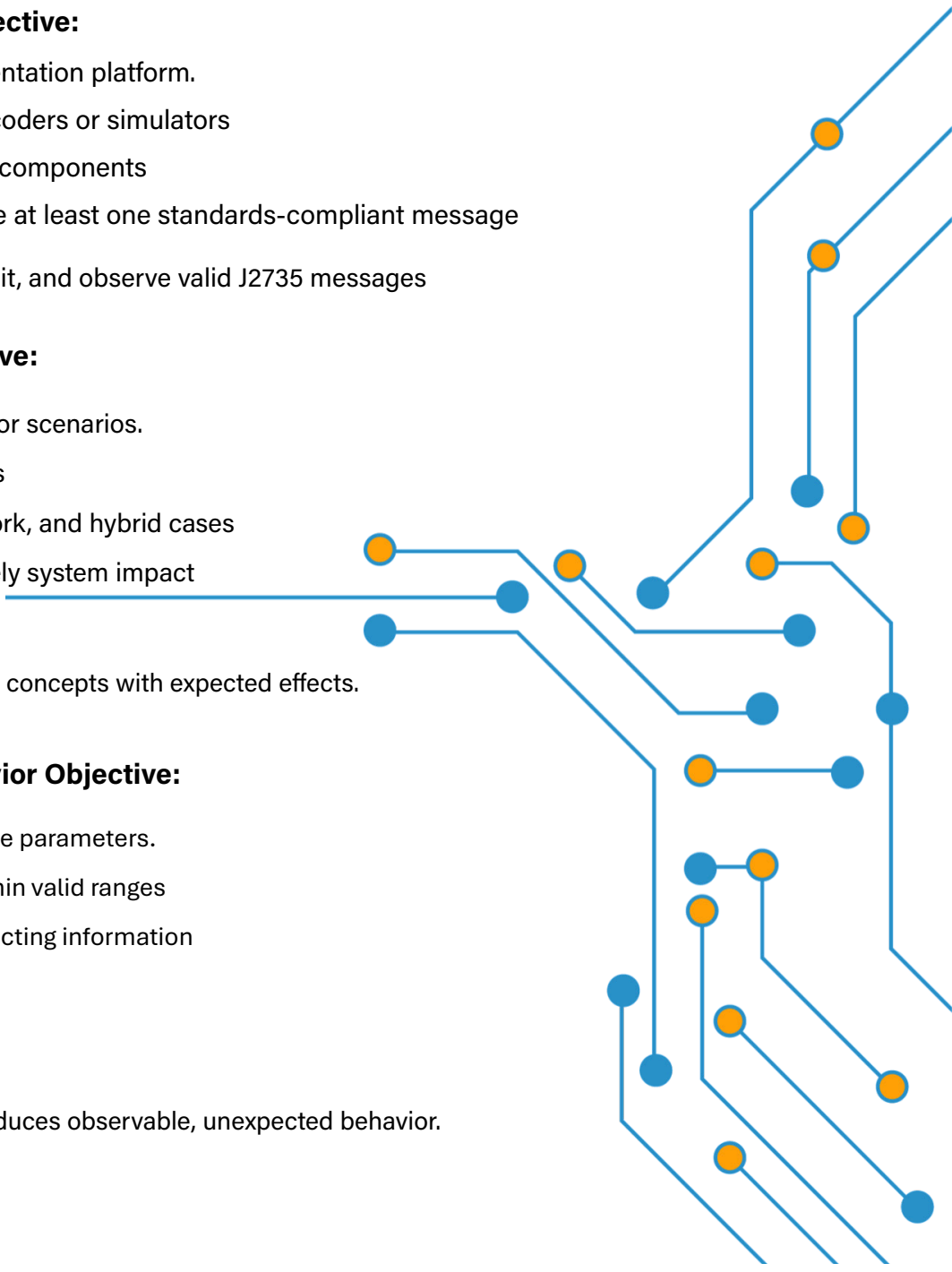
- Identify multiple potential misbehavior scenarios.
- Brainstorm at least 5 candidate ideas
- Consider vehicle, intersection, network, and hybrid cases
- Evaluate technical feasibility and likely system impact
- Narrow to 2–3 promising concepts

Outcome: Shortlist of viable disruption concepts with expected effects.

4. Experiment with Message Behavior Objective:

- Test controlled variations in message parameters.
- Modify selected message fields within valid ranges
- Test timing inconsistencies or conflicting information
- Observe how the system responds
- Document measurable impacts

Outcome: At least one candidate produces observable, unexpected behavior.



Student Team Milestones - Continued

5. Select & Define the Final Scenario Objective:

- Commit to the strongest demonstration concept
- Select the most impactful and repeatable misbehavior
- Define: Normal baseline behavior, Misbehavior conditions
    - Observable disruption
    - Create a simple system diagram

Outcome: Shortlist of viable disruption concepts with expected effects.

6. Engineer the Demonstration Objective:

Develop a reliable and compelling Plugfest demo.

- Script the demonstration sequence:
    - Normal operation
    - Activate misbehavior
    - Show a disruptive result
- Capture logs, visual alerts, or system responses
- Ensure repeatability

Outcome: Reliable, repeatable disruption demonstration.

7. Design Countermeasures Objective:

Propose detection or prevention strategies.

- Identify why the system accepted the misbehavior

    Propose:

    - Plausibility checks
    - Cross-message validation
    - Multi-source verification
- Policy or architectural improvements
- Evaluate effectiveness and tradeoffs

Outcome: Technically credible mitigation strategy.

8. Prepare Final Plugfest Presentation Objective:

Deliver a professional, concise showcase.

- 5–10 minute demonstration
- Clear explanation of:
    - Misbehavior concept
    - Observed impact
    - Proposed countermeasure
    - One-page system diagram or slide summary

Outcome: Demo-ready presentation for industry audience.