4월 28, 2025 Library

ISO 21434: Connected Car OTA Security

The Era of Connected Cars and the Importance of OTA, and Security Threats

Hello, Engineers! This is Hermes Solution.

Today, cars are rapidly transforming from mere means of transportation into ‘smart devices on wheels,’ intricately connected by sophisticated software and networks. This transformation is accelerating even further as we enter the era of the Software-Defined Vehicle (SDV).

One of the core drivers of this connected car revolution is the OTA (Over-the-Air) update technology. Through OTA, manufacturers can keep vehicle software up-to-date without requiring a service center visit, add new features, and most importantly, quickly resolve security vulnerabilities.

However, as vehicle connectivity strengthens and OTA utilization increases, potential security threats that attackers can exploit also grow. If attackers inject malicious updates or intercept and manipulate the update process, it can go beyond vehicle malfunction and pose a serious threat to passenger safety. Malicious update injection, update tampering, Man-in-the-Middle (MitM) attacks, Denial-of-Service (DoS) attacks, authentication bypass, supply chain attacks, and exploitation of API vulnerabilities are emerging as major security threats targeting connected cars. These attacks can lead to devastating consequences such as data breaches, immense financial loss, and damage to brand image.

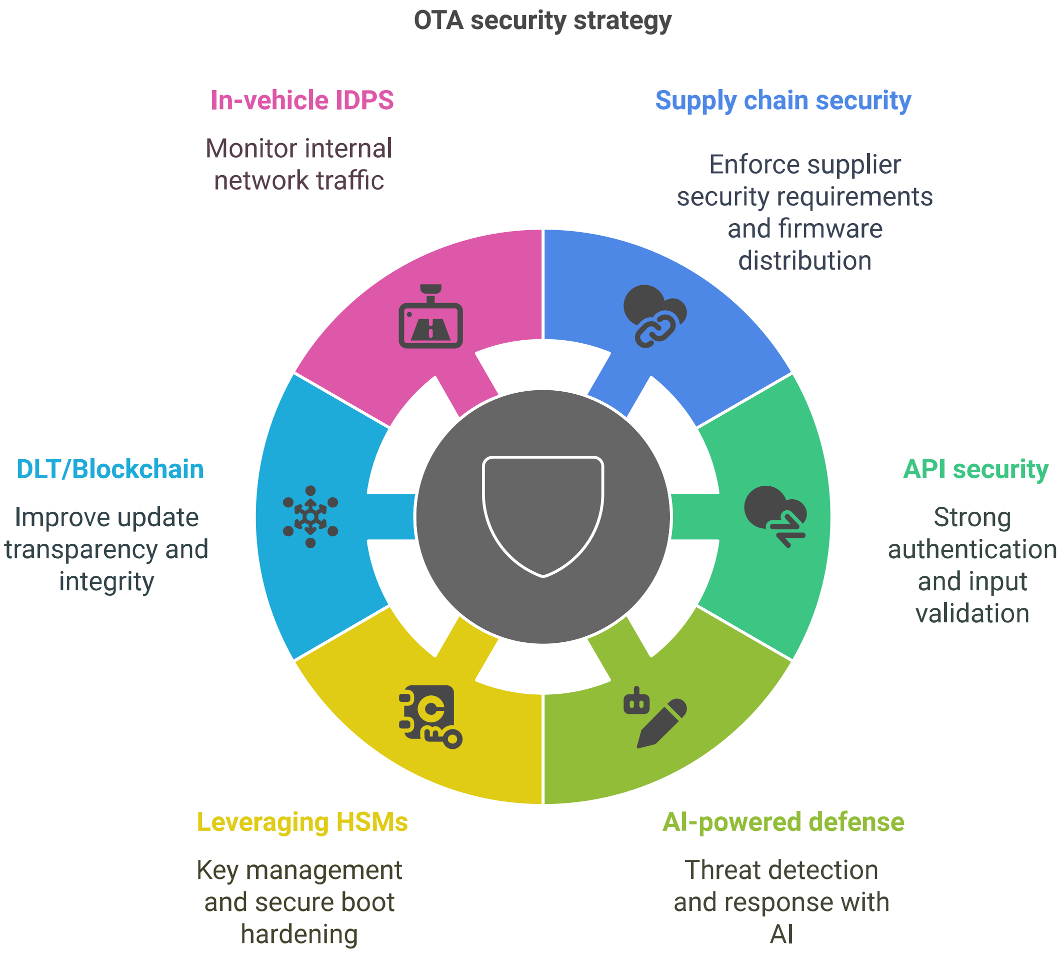
Against this backdrop, international standards like ISO/SAE 21434 for automotive cybersecurity and international regulations based on it, such as UN R155/R156, provide essential guidelines for building a safe and reliable OTA update environment. In this blog post, we aim to delve deeply into how to establish and implement a secure OTA update security strategy from the

21434 standard. Now, OTA security is no longer a technical option, but an essential strategy for

l compliance in the connected car business.

OTA Security: Why Essential? New Risks for Connected Cars

OTA is a powerful tool for managing vehicle software, but it simultaneously provides attackers with new attack vectors. It has a broad attack surface, including in-vehicle clients, backend servers, and communication channels.


OTA security strategy
In-vehicle IDPS
Monitor internal network traffic
Supply chain security
Enforce supplier security requirements and firmware distribution
DLT/Blockchain
Improve update transparency and integrity
API security
Strong authentication and input validation
Leveraging HSMs
Key management and secure boot hardening
AI-powered defense
Threat detection and response with AI

Key Threat Scenarios:

- **Malicious Update Injection:** Attackers distribute malware by compromising servers or intercepting communications (e.g., disabling braking systems).
- **Update Tampering:** Altering update content during transmission (e.g., manipulating engine parameters).
- **Man-in-the-Middle (MitM):** Intercepting communications to steal authentication information or deliver counterfeit updates.
- **Denial-of-Service (DoS):** Paralyzing server or vehicle update functions, delaying patches.
- **Authentication/Authorization Bypass:** Forcing the installation of specific vehicle updates through unauthorized access.
- **Supply Chain Attacks:** Infected software from the development/production stage being distributed via OTA.
- **API Vulnerability Exploitation:** Attempting data theft or unauthorized updates by exploiting vulnerabilities in backend/connected service APIs.

These threats can lead to functional safety risks, data breaches, financial loss, recalls, and brand damage. Recent attacks are evolving towards ransomware and data theft for financial gain, extending to the entire automotive ecosystem. The surge in automotive-related security vulnerabilities (CVEs) is linked to the expansion of OTA, highlighting the importance of rapid security patching.

International regulations UN R155 (Cyber Security Management System, CSMS) and UN R156 (Software Update Management System, SUMS) mandate the establishment of CSMS and SUMS, which are essential requirements for type approval in major nd Korea. Secure OTA is central to compliance with these regulations and must satisfy both nts.

Roadmap for Secure OTA: ISO/SAE 21434 Standard

ISO/SAE 21434 is the international standard for 'Cybersecurity Engineering' in road vehicles, providing a framework for identifying, evaluating, and managing risks throughout the vehicle lifecycle.

Key Principles:

- **CSMS:** Establishment of an organizational-level cybersecurity management system (Mandated by UN R155).
- **Risk-Based Approach:** Prioritizing security activities based on Threat Analysis and Risk Assessment (TARA).
- **Security by Design:** Considering security from the initial development stages and performing security activities throughout all phases.
- **Full Lifecycle Management:** Emphasizing continuous security activities in the post-production phase (vulnerability monitoring, updates, incident response).

OTA updates are a core function in the 'Operation and Maintenance' phase, and ISO/SAE 21434 defines specific requirements for this phase. While ISO/SAE 21434 itself is a standard, UN regulations effectively require compliance with ISO/SAE 21434 for type approval, making it a practical path to regulatory compliance.

Building a Robust OTA Ecosystem: ISO/SAE 21434-Based Approach

Establishing a secure OTA strategy involves comprehensive activities from risk analysis to design, implementation, testing, and ongoing management.

1. Starting Point: Performing OTA System TARA

This is a systematic process (ISO/SAE 21434 Clause 15) for identifying, evaluating, and determining the risk level of potential threats and vulnerabilities in the OTA system.

- **Asset Identification:** Clearly defining assets to protect (client, firmware, keys, servers, etc.).
- **Threat Scenario Identification:** Deriving threat situations that could harm assets (malicious injection, tampering, etc.).
- **Vulnerability Analysis:** Analyzing weaknesses in design, implementation, and operation.
- **Impact Assessment:** Evaluating the severity of damage if a threat succeeds (safety, financial, operational, privacy).
- **Risk Level Determination:** Synthesizing likelihood and impact to prioritize. TARA results serve as the basis for deriving security goals and requirements.

2. Building the Defense Line: Defining Core OTA Security Requirements

Concrete security requirements are defined based on TARA and reflected in the system design.

- **Integrity and Authentication:** Preventing update tampering and confirming trusted origin (Hash, Digital Signature, PKI).
- **Confidentiality:** Protecting update content from unauthorized access (Strong encryption).
- **Secure Communication:** Protecting vehicle-to-server communication (TLS, Mutual Authentication).
- **Vehicle/Server Authentication and Authorization:** Verifying the identity of the target vehicle and server, confirming update suitability (Certificates, HSM, DID/VC).
- **Secure Storage:** Safely storing sensitive information (Secure memory, HSM).
- **Secure Update Process:** Verification before installation, secure installation and recovery (Signature/hash verification, Secure Boot, Rollback).

3. Fortifying the Vehicle: Secure Implementation Strategy

Defined requirements are effectively implemented in the system.

- **Establishing Trust Anchors (Secure Boot):** Ensuring the integrity of initial boot software to secure the OTA process.
- **Hardware Security Enhancement (HSM):** Providing strong security through dedicated secure hardware for key management and cryptographic operations. Used in Secure Boot, certificate management, etc.

Adhering to secure coding standards, utilizing static/dynamic analysis to prevent vulnerabilities.

olation: Applying multiple layers of security mechanisms, isolating communication between critical systems.

4. Preparing for Failure: Designing a Robust Rollback Mechanism

A function is essential to prevent vehicle bricking in case of update failure and restore a stable state.

- **A/B Partitioning:** Using two partitions, booting from the previous partition on failure (Fast rollback, requires double memory).
- **Backup-Based Caching:** Storing the new update in a separate space, restoring from backup on failure (Hardware flexibility, downtime occurs). Anti-downgrade functionality to prevent malicious rollback to an older version should also be considered.

5. Verifying Strength: Performing Comprehensive Security Testing

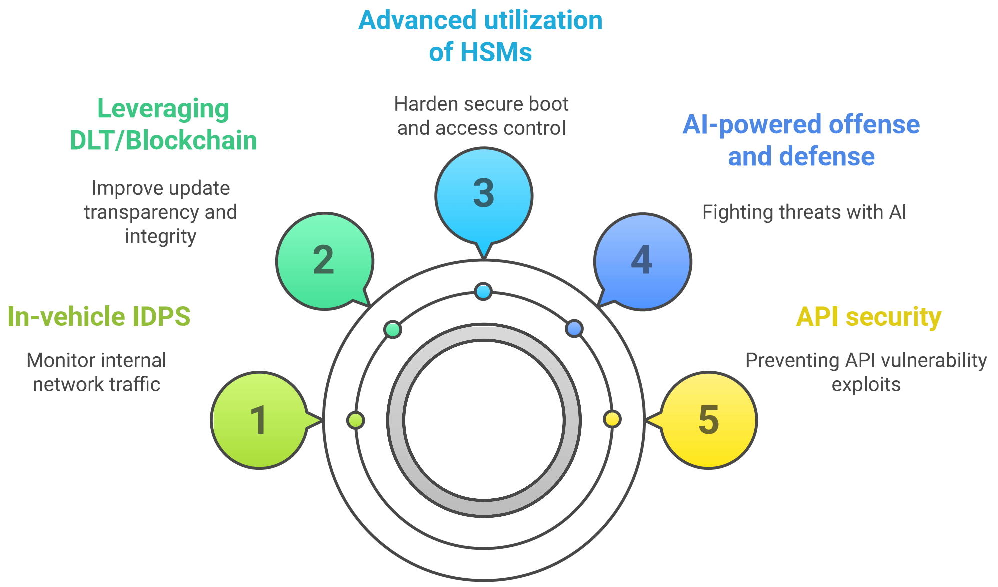
Verification of the implemented system's defenses is essential.

- **Penetration Testing:** Actively searching for vulnerabilities from an attacker's perspective.
- **Fuzz Testing:** Inputting abnormal data to identify vulnerabilities.
- **Code Review & Static/Dynamic Analysis:** Analyzing source code and execution stages for vulnerabilities.
- **Cryptographic Implementation Verification:** Confirming compliance with cryptographic standards and security strength. Robust OTA security is achieved through a multi-layered, holistic approach combining technology, hardware, software, processes, and resilience.

Preparing for the Future: Latest OTA Security Threats and Defense Technologies

The threat

Future OTA security strategy


Advanced utilization of HSMs
Harden secure boot and access control
3
Leveraging DLT/Blockchain
Improve update transparency and integrity
2
AI-powered offense and defense
Fighting threats with AI
4
In-vehicle IDPS
Monitor internal network traffic
1
API security
Preventing API vulnerability exploits
5

- **Supply Chain Security Enhancement:** Countering supply chain vulnerability attacks through supplier security requirements, SBOM utilization, and firmware distribution security enhancement.
- **API Security:** Preventing API vulnerability exploitation through strong authentication/authorization and input validation.
- **AI-Based Attacks and Defense:** Emergence of sophisticated AI-powered attacks, countered by AI-based IDS and other defenses.
- **Advanced HSM Utilization:** Utilizing HSM beyond key management for enforcing secure boot, access control, etc.
- **DLT/Blockchain Utilization:** Researching improvements in update transparency, integrity, and auditability using DLT/blockchain.