VECTORAL PUBLISHING

Date of publication 25/05/2023, date of current version 25/07/2023.

VECTORAL manuscript Identifier 08.02-06/VECTORAL.2023.PUB

CHALLENGES ASSOCIATED WITH THE DEPLOYMENT OF SOFTWARE OVER-THE-AIR (SOTA) UPDATES IN THE AUTOMOTIVE INDUSTRY

RAHUL EKATPURE[1]
[1]Independent researcher

Corresponding author: Ekatpure, R.

ABSTRACT The rise of connected and autonomous vehicles has made Software Over-the-Air (SOTA) updates essential for enhancing vehicle performance, security, and functionality, enabling remote software improvements and fixes without physical vehicle access. However, deploying SOTA updates poses significant challenges that must be addressed to ensure vehicle safety, security, and functionality. This paper explores these challenges in the automotive industry. This study finds that the key challenges include maintaining security and privacy, where vulnerabilities in the SOTA process can expose vehicles to cyber threats, potentially compromising safety and sensitive user data. Network and connectivity issues present another hurdle, as intermittent or poor connectivity can disrupt updates, leading to incomplete installations and software corruption. Compatibility and integration issues between new updates and existing vehicle hardware and software can result in system instability and functionality loss. User acceptance of SOTA updates is also a concern, as skepticism and perceived inconvenience may lead to reluctance in accepting updates, delaying essential improvements. Additionally, compliance with diverse regulatory standards across different regions complicates the deployment process, requiring persistent alignment with legal requirements to avoid penalties. The paper suggests the need for an approach that involves cybersecurity, software development, network engineering, and regulatory compliance to effectively address these challenges and ensure successful SOTA update implementation in vehicles.

INDEX TERMS automotive software, cybersecurity, network connectivity, regulatory compliance, SOTA updates

I. INTRODUCTION

Over-the-Air (OTA) programming encompasses a range of automated processes for updating firmware, software, and encryption keys within various systems Villegas et al., 2019. Key categories within OTA programming include Software Over-the-Air (SOTA), which pertains to the distribution and installation of software updates across devices. Firmware Over-the-Air (FOTA) focuses on updating firmware, typically involving lower-level control systems within electronic devices. Over-the-Air Service Provisioning (OTASP) facilitates the initial setup or modification of service configurations on devices, often in telecommunications contexts Andrade et al., 2017. Over-the-Air Provisioning (OTAP) involves the deployment of configuration settings, such as network or application configurations, to devices. Over-the-Air Parameter Administration (OTAPA) deals with the remote management and adjustment of device parameters , enabling real-time changes to operational settings without direct physical intervention. These OTA mechanisms collectively enhance the flexibility, security, and functionality of electronic systems by enabling remote updates and configurations.

Software Over-the-Air (SOTA) updates have become prevalent in the automotive industry, with leading vehicle manufacturers regularly deploying these updates to enhance infotainment and navigation systems. SOTA technology extends beyond entertainment systems, offering the capability to update software that manages a vehicle's physical components or electronic signal processing systems. There is widespread adoption of SOTA, Firmware Over-the-Air (FOTA) updates are less commonly implemented on a large scale. Pioneers in this area, such as Tesla and NIO, are among the few automotive manufacturers to have successfully deployed FOTA updates extensively. The broader im-

OTA Category	Description
Software Over-the-Air (SOTA)	Distribution and installation of software updates across devices.
Firmware Over-the-Air (FOTA)	Updating of firmware, usually for lower-level control systems within electronic devices.
Over-the-Air Service Provisioning (OTASP)	Initial setup or modification of service configurations, commonly in telecommunications.
Over-the-Air Provisioning (OTAP)	Deployment of configuration settings, such as network or application configurations.
Over-the-Air Parameter Administration (OTAPA)	Remote management and adjustment of device parameters, enabling real-time changes to operational settings.

TABLE 1. Key Categories in Over-the-Air (OTA) Programming

Aspect	Description
SOTA Capabilities	Delivers updates across various vehicle systems, including software for managing physical components and electronic signal processing systems.
User Interface Enhancements	SOTA updates can enhance infotainment screens, instrument clusters, and other user interfaces with improved functionality and new features.
Local Updates	Traditional method requiring physical presence at service centers where technicians update software using specialized tools through an OBD connection.
Remote Updates	Enables updates without visiting service centers, enhancing convenience by remotely maintaining and enhancing vehicle performance and user experience.

TABLE 2. SOTA Technology Capabilities and Software Update Methods in the Automotive Sector

plementation of FOTA is constrained by its more demanding requirements, which include significant computing power, faster mobile connectivity, and enhanced security measures. These stringent prerequisites for FOTA reflect the complexities involved in updating firmware, which directly interacts with the hardware components of vehicles and necessitates robust mechanisms to ensure reliability and safety Blázquez et al., 2021 Chandra et al., 2016. Software-Over-the-Air (SOTA) technology enables the remote downloading and updating of vehicle software from cloud-based servers via Wi-Fi or mobile networks. This process can occur directly to the vehicle or through the owner's device, which then transfers the update, typically via Bluetooth. SOTA updates are designed to be efficient, often utilizing 'delta' files rather than complete software installations. A delta file contains only the changes needed, minimizing download time and reducing distribution costs for manufacturers. This contrasts with a 'full image' file that would require significantly more bandwidth and storage. Consequently, this method optimizes the update process, allowing for rapid deployment and minimizing the disruption to vehicle operations.

SOTA technology can deliver updates across a range of vehicle systems. These include software that manages physical components, such as the braking or steering systems, as well as electronic signal processing systems that handle functions like adaptive cruise control or collision avoidance. Additionally, SOTA updates can enhance user interfaces, including infotainment screens and instrument clusters, providing users with improved functionality and new features without requiring a visit to a service center. This capability underscores the versatility of SOTA in maintaining and enhancing vehicle performance and user experience. Software updates in the automotive sector are essential for maintaining and enhancing vehicle functionality. These updates can be classified into two primary types: Local and Remote updates. The local update method, a traditional approach, requires vehicle owners to visit service centers or dealerships where technicians use specialized tools to update the software through an On-Board Diagnostics (OBD) connection. This process is typically labor-intensive, time-consuming, and necessitates physical presence at a service facility, which can be inconvenient for users and costly for manufacturers due to the logistics and manpower involved.

In contrast, Remote or Over-the-Air (OTA) updates leverage wireless communication to send software updates directly to vehicles while they are in operation Doddapaneni et al., 2017. This modern technique allows updates to be performed seamlessly without requiring a visit to a service center, aligning with Original Equipment Manufacturers' (OEMs) goals to minimize disruption to drivers. OEMs aim to ensure that OTA updates do not render vehicles unusable or pose risks to their functionality during the update process. This approach not only enhances user convenience by reducing downtime but also aligns with contemporary expectations for continuous, hassle-free vehicle maintenance El Jaouhari and Bouvet, 2022 Frey et al., 2021.

The implementation of software OTA updates offers several significant advantages for OEMs. First, it provides a fast and cost-effective means of updating vehicle software without necessitating physical recalls. This efficiency translates into substantial cost savings and operational benefits, as it eliminates the logistical complexities associated with traditional updates Halder et al., 2020 Blázquez et al., 2021. Additionally, OTA updates enable manufacturers to swiftly

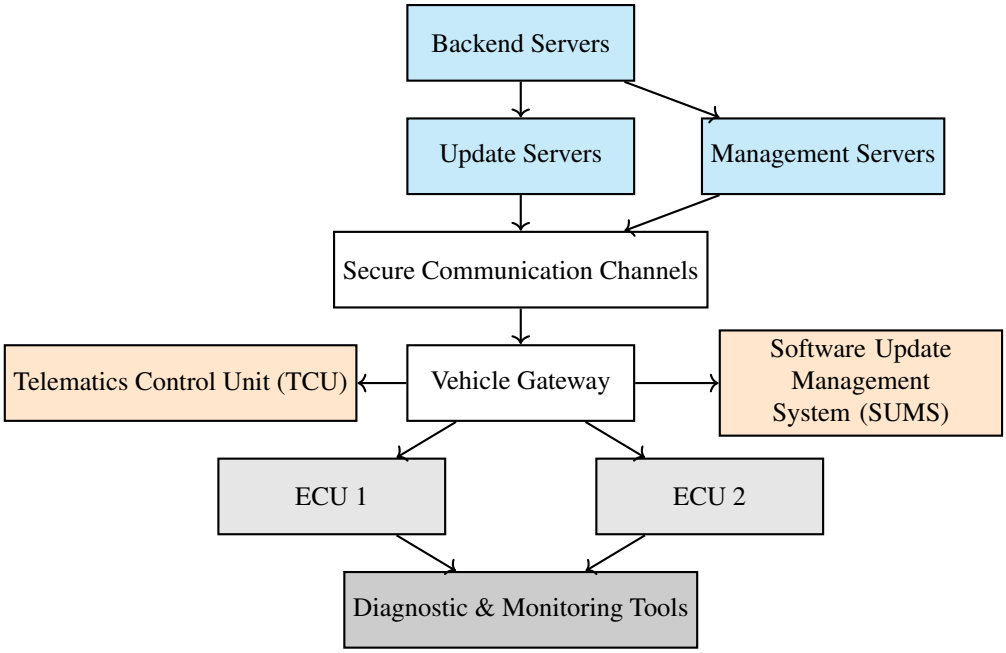

FIGURE 1. Architecture of Software Over-the-Air (SOTA) for Automotive

address software bugs and security vulnerabilities, thereby enhancing vehicle safety and reliability. This capability is crucial in a rapidly evolving digital landscape where quick responses to emerging threats are imperative.

OTA updates present OEMs with opportunities to generate new revenue streams by introducing new features and functionalities after a vehicle has been deployed Hess et al., 2020 Malik et al., 2022. For instance, manufacturers can roll out premium features, software enhancements, or performance upgrades that can be purchased by the vehicle owners, thus adding value beyond the initial sale. This potential for ongoing revenue generation through post-deployment updates represents a strategic advantage, allowing OEMs to leverage software as a service model in the automotive industry Shavit et al., 2007 Sery et al., 2021.

As seen in figure 1, Over-the-Air (SOTA) systems for automotive applications incorporate several architecture to facilitate the efficient delivery and update of software to vehicles. This architecture encompasses several critical components. Backend Servers, including Update Servers and Management Servers, form the core infrastructure, responsible for software storage, update management, and distribution. Update Servers specifically host software updates, patches, and new releases, ensuring their effective distribution to vehicles. Management Servers handle administrative functions, such as user and vehicle management, update scheduling, and the collection of diagnostic data, providing OEMs with interfaces to manage software and its deployment. Secure Communication Channels, utilizing cellular networks, Wi-Fi, or dedicated automotive communication protocols, enable safe transmission of updates from the backend servers to vehicles, employing encryption and authentication mechanisms to maintain security. The Vehicle Gateway serves as the intermediary between the vehicle's internal network and external communication channels, managing incoming updates and distributing them appropriately across vehicle subsystems. Electronic Control Units (ECUs), which control various vehicle functions, receive and integrate new software updates independently. The Telematics Control Unit (TCU), a specialized ECU, manages the vehicle-backend server communication, overseeing the download and installation of updates. Within the vehicle, the Software Update Management System (SUMS) orchestrates the update process, including verification, installation, and rollback if necessary, ensuring operational continuity. Diagnostic and Monitoring Tools are employed to collect data throughout and following the update process, verifying software functionality and identifying potential issues Andrade et al., 2017.

II. SECURITY AND PRIVACY RISKS

The deployment of Software Over-the-Air (SOTA) updates in the automotive industry has transformed how vehicle software is maintained and enhanced. However, the increasing reliance on SOTA introduces significant security and privacy risks that must be addressed to ensure the safe and reliable operation of vehicles . The primary challenge in implementing SOTA updates is ensuring the security of the update process against potential cyber threats. As vehicles become more connected and reliant on digital systems, they present attractive targets for cyber attackers. The SOTA process involves the transmission of software updates from cloud-based servers to vehicles over various communication channels, which inherently exposes the system to risks of interception, unauthorized access, and manipulation Bulmus

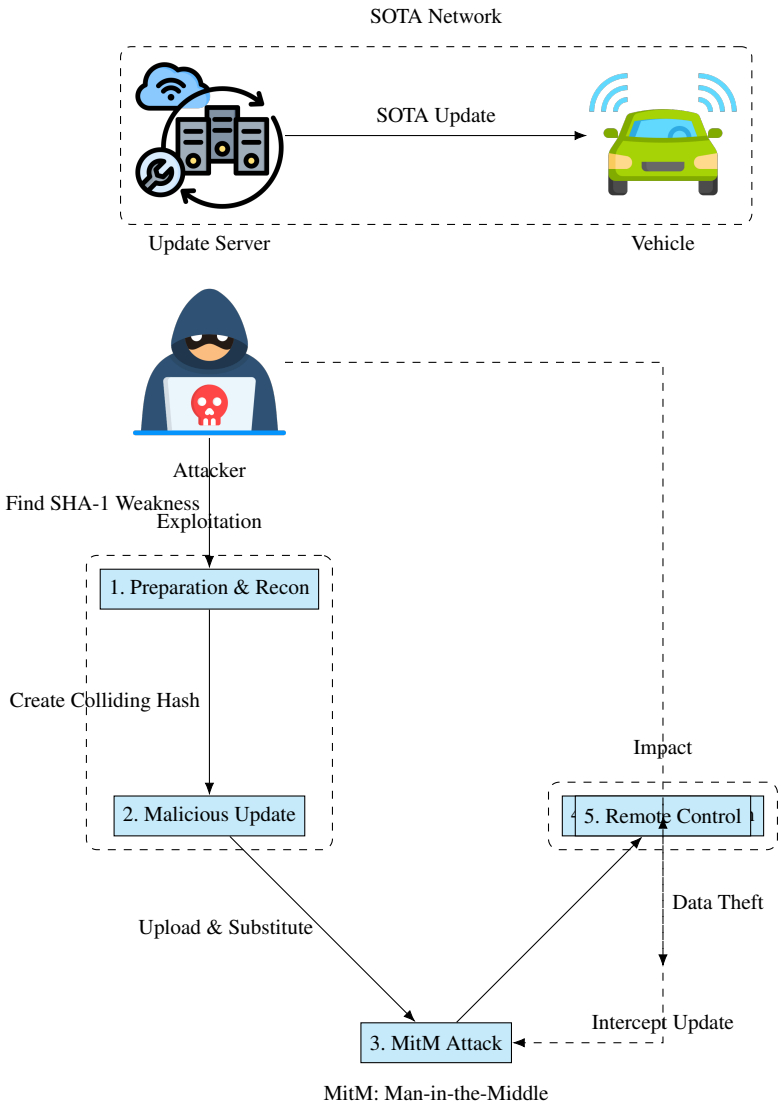

FIGURE 2. Diagram of Cyber Attack on Vehicle through SOTA Vulnerability

et al., 2017.

One of the critical security challenges is ensuring that the update mechanism itself is secure and cannot be exploited. This involves safeguarding the entire update lifecycle, from the generation and packaging of software updates to their transmission, installation, and verification on the vehicle. Any vulnerability in this chain can be exploited by attackers to inject malicious code, gain unauthorized control over vehicle systems, or disrupt vehicle operations.

Moreover, the integration of SOTA with the vehicle's network architecture presents additional security challenges. Modern vehicles are equipped with numerous Electronic Control Units (ECUs) that control various functions, including safety-critical systems like braking and steering. Ensuring that the SOTA updates do not introduce vulnerabilities into these systems is crucial, as any compromise could have severe safety implications. Attackers could potentially exploit these vulnerabilities to manipulate vehicle behavior or disable critical functions, leading to accidents or other dangerous situations.

The impact of failing to secure SOTA updates can be profound, with implications for both vehicle safety and user privacy. Unauthorized access to vehicle systems through a compromised SOTA process can lead to a range of negative outcomes, including accidents, theft, and data breaches Doddapaneni et al., 2017.

Firstly, if an attacker gains control over the vehicle's systems, they could manipulate the vehicle's operation, leading to potential accidents. For instance, an attacker could disable the braking system, alter steering inputs, or interfere with the engine control, resulting in hazardous driving conditions. This poses a significant risk to the safety of vehicle occupants and other road users.

Secondly, cyber attackers could exploit vulnerabilities in the SOTA process to steal vehicles. By disabling security features or bypassing immobilizers through compromised

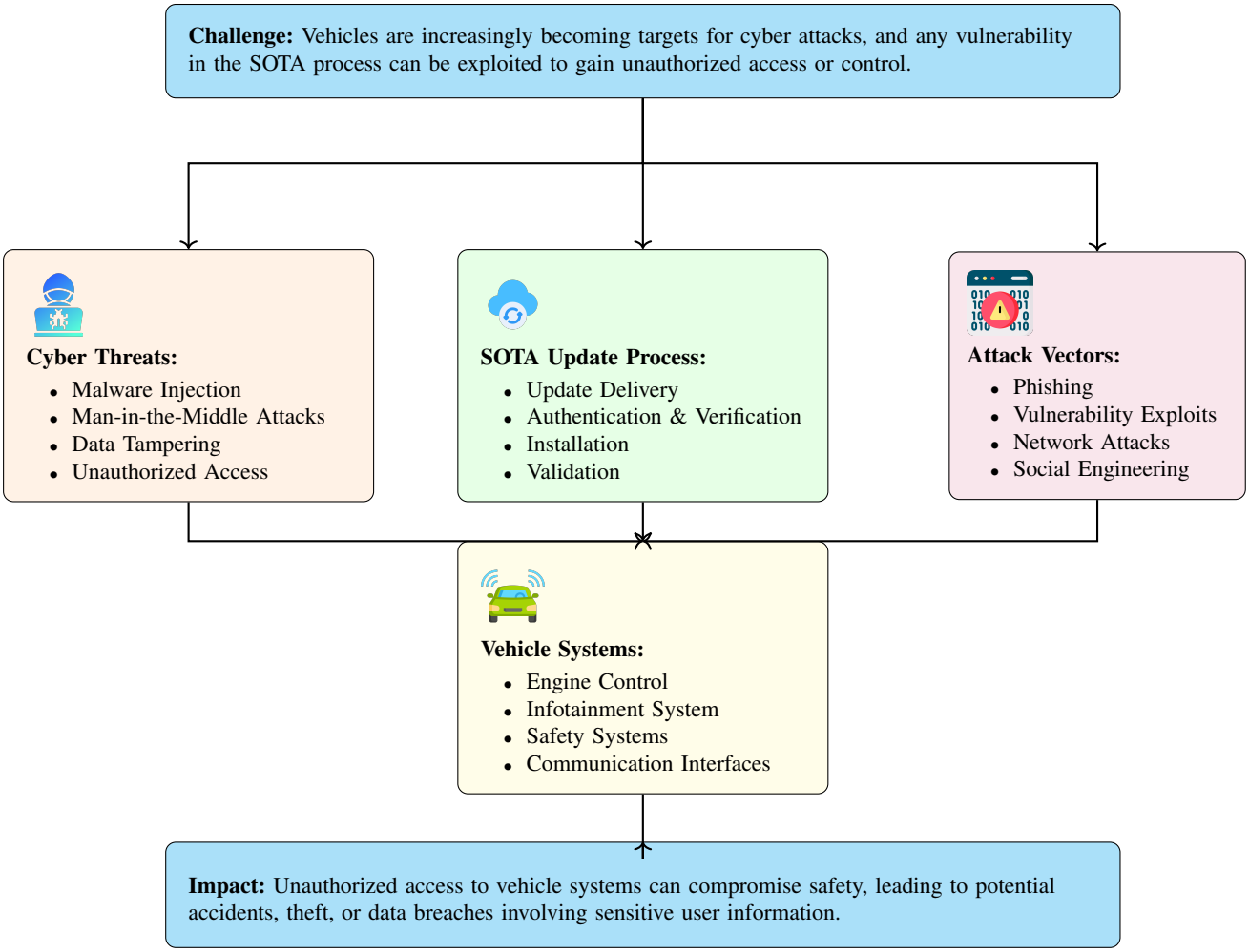

FIGURE 3. Diagram of Securing SOTA Updates: Challenges, Cyber Threats, SOTA Process, Attack Vectors, and Impacts

updates, attackers can gain unauthorized access to the vehicle, enabling theft. This not only results in financial losses for the vehicle owner but also undermines the trust in SOTA-enabled systems.

Thirdly, data breaches resulting from insecure SOTA processes can compromise sensitive user information. Modern vehicles collect and store a vast amount of data related to vehicle performance, location, user preferences, and personal details. If attackers access this data through a compromised update process, it can lead to privacy violations and identity theft. The exposure of such sensitive information can have long-term repercussions for users, including financial loss and damage to personal reputation Doddapaneni et al., 2017.

To mitigate the security and privacy risks associated with SOTA updates, several strategies can be implemented. These strategies focus on enhancing the security of the update process and protecting the vehicle's systems from potential threats.

1. Robust Encryption: Encrypting software updates is essential to protect the integrity and confidentiality of the data during transmission. Encryption ensures that even if the update package is intercepted, it cannot be read or altered by unauthorized parties. Implementing strong encryption algorithms, such as Advanced Encryption Standard (AES), can provide a high level of security for the update packages.

2. Authentication Protocols: Authenticating the source and integrity of the update package is crucial to prevent unauthorized updates. This can be achieved through digital signatures and certificates that verify the identity of the server sending the update and ensure that the update has not been tampered with. Public Key Infrastructure (PKI) can be employed to manage digital certificates and authenticate update sources.

3. Secure Communication Channels: Utilizing secure communication channels, such as Transport Layer Security (TLS), for transmitting updates helps protect against interception and man-in-the-middle attacks. TLS encrypts the data being transmitted and ensures that the communication is only accessible to the intended recipient, thereby preventing eavesdropping and data tampering El Jaouhari and Bouvet, 2022.

4. Regular Security Audits: Conducting regular security audits of the SOTA process and associated systems is essen-

Security Strategy	Deatails
Robust Encryption	Encrypting software updates to protect data integrity and confidentiality during transmission, using strong algorithms like AES.
Authentication Protocols	Using digital signatures and certificates to verify the source and integrity of the update package, preventing unauthorized updates.
Secure Communication Channels	Utilizing secure channels such as TLS for transmitting updates to prevent interception and man-in-the-middle attacks.
Regular Security Audits	Conducting audits including penetration testing and code reviews to identify and address vulnerabilities in the SOTA process.
Update Verification and Rollback Mechanisms	Verifying update integrity and compatibility before installation and enabling rollback to a previous version if issues arise.
Secure Boot and Runtime Protections	Ensuring only authenticated software runs on the system, with runtime protections to monitor and respond to suspicious activities.

TABLE 3. Strategies for Mitigating Security and Privacy Risks in SOTA Updates

tial to identify and address potential vulnerabilities. Security audits can include penetration testing, code reviews, and vulnerability assessments to evaluate the security posture of the update mechanism and associated infrastructure. These audits help in proactively identifying weaknesses and implementing corrective measures before they can be exploited by attackers Ghosal et al., 2022.

5. Update Verification and Rollback Mechanisms: Implementing mechanisms to verify the integrity and compatibility of updates before installation can prevent the deployment of malicious or corrupted updates. Additionally, having rollback mechanisms in place allows the system to revert to a previous stable version if an update fails or introduces issues, thereby minimizing the risk of operational disruption.

6. Secure Boot and Runtime Protections: Integrating secure boot mechanisms ensures that only authenticated and authorized software can be executed on the vehicle's systems. Secure boot checks the integrity of the firmware and software during the boot process, preventing the execution of unauthorized code. Runtime protections, such as integrity monitoring and anomaly detection, can help detect and respond to suspicious activities during the operation of the vehicle.

III. NETWORK AND CONNECTIVITY CONSTRAINTS

The deployment of Software Over-the-Air (SOTA) updates in the automotive industry is fundamentally dependent on reliable network connectivity to ensure that software updates are delivered seamlessly and effectively. Vehicles, by nature of their mobility, encounter a diverse array of environments where network connectivity can vary significantly due to factors such as geographic location, network coverage, or interference. This variability in connectivity presents substantial challenges for maintaining the continuity and integrity of the SOTA process. Geographic location plays a critical role in connectivity quality; while urban areas typically enjoy robust network infrastructure with comprehensive cellular and Wi-Fi coverage, rural and remote regions often experience weak or no network signals. Additionally, environmental conditions such as mountainous terrain, dense forests, or tunnels can obstruct signal transmission, further complicating the connectivity landscape for vehicles in motion.

Algorithm 1 Chunked Data Transfer Algorithm

Input: Update package P, Network conditions N, Available resources R
Output: Updated system
`InitializeUpdateProcess(`P, N, R`)` **begin**
  Identify and prepare the update package P Determine optimal chunk size S based on N and R
end
`ChunkData(`P, S`)` **begin**
  Divide the update package P into smaller chunks C_i Assign unique identifiers ID_i to each chunk C_i
end
`TransmitChunks(`C_i`)` **begin**
  foreach *chunk* C_i **do**
    Send C_i sequentially or concurrently based on N Implement acknowledgment mechanism for C_i **if** *acknowledgment received for* C_i **then**
      Mark C_i as successfully transmitted
    end
  end
end
`HandleDisruptions(`C_i`)` **begin**
  Monitor transmission for disruptions or failures **if** *disruption or failure detected* **then**
    Retransmit only the affected chunk(s) C_i
  end
end
`ReassembleData(`C_i`)` **begin**
  Verify integrity and completeness of all received chunks C_i Reassemble chunks C_i into the original update package P Apply the update to the system
end

Network coverage itself is highly variable, with cellular networks offering extensive reach but subject to issues such as signal interference and varying load conditions, which can affect the stability and speed of data transmission. Wi-

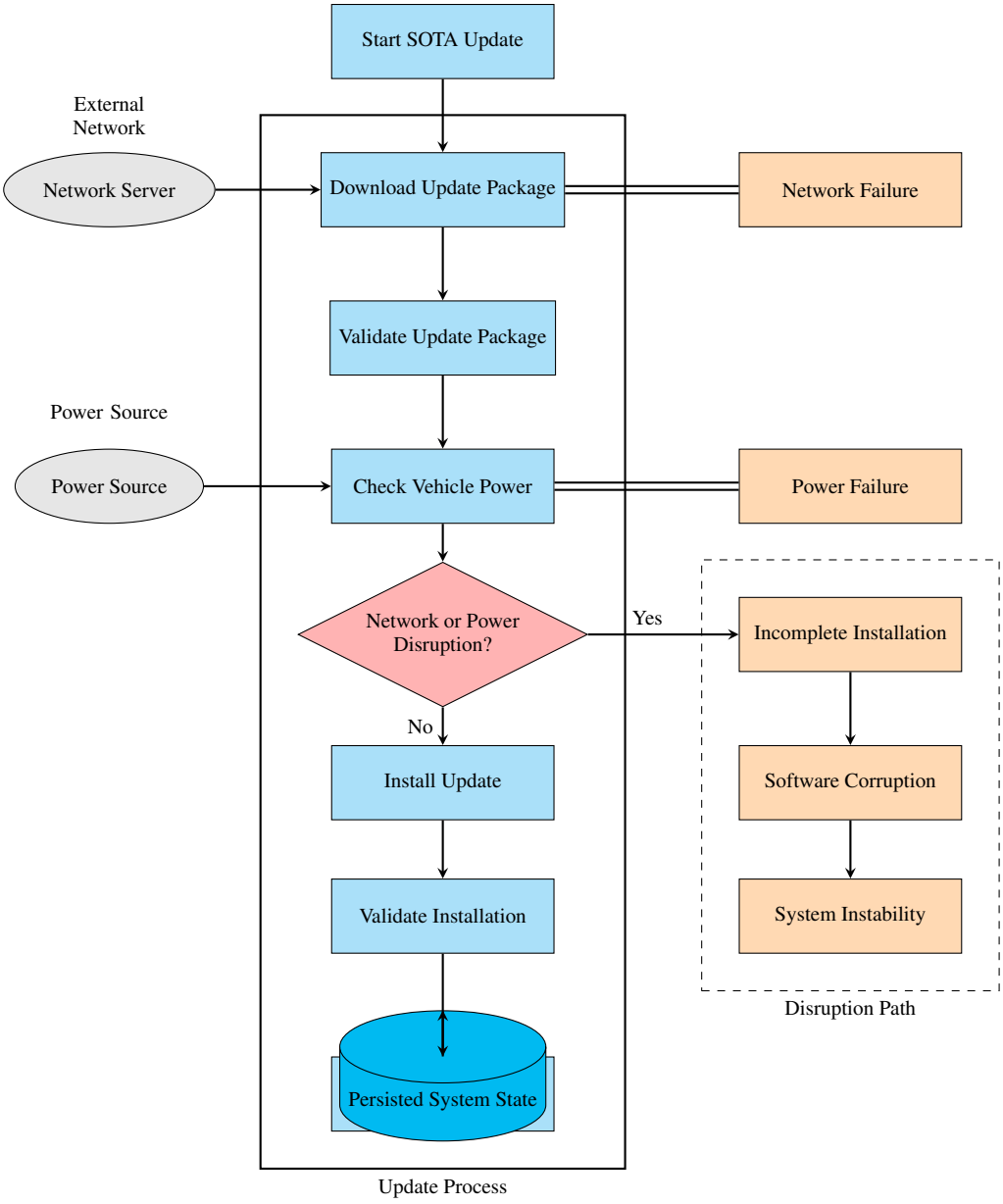

FIGURE 4. SOTA Update Process for Autonomous Vehicles with Potential Disruptions

Fi networks, though capable of providing high bandwidth, are generally limited to specific locations such as residential areas or commercial establishments and are not available during travel. Network interference from other wireless devices, physical obstructions, and atmospheric conditions can also degrade signal quality, leading to fluctuations in connection stability and bandwidth. These connectivity challenges necessitate the development of strategies to ensure that SOTA updates can be reliably delivered despite these varying conditions Guissouma et al., 2019.

The impacts of connectivity disruptions during SOTA updates are significant and, affecting both vehicle functionality and user safety. Connectivity interruptions can lead to incomplete installations where the update process is halted before it can be fully completed, leaving the vehicle's software in an inconsistent or partially updated state. This incomplete installation can disable critical functions or cause the vehicle's systems to behave unpredictably. For instance, an interrupted update to software controlling the braking system could impair braking performance, creating serious safety risks Halder et al., 2020. Additionally, interrupted transmissions can corrupt the software package, resulting in the installation of erroneous or incomplete code. Such corruption can lead to system instability, causing the vehicle's electronic systems to malfunction or crash, impacting everything from infotainment systems to critical operational components. These potential disruptions underline the importance of robust mechanisms to manage connectivity-related issues and ensure the

reliability of SOTA updates Halder et al., 2020.

To mitigate the network and connectivity constraints associated with SOTA updates, several strategies can be implemented to enhance the reliability and fault tolerance of the update process. Fault-tolerant update mechanisms are essential, with strategies such as chunked data transfer, which breaks the update package into smaller, more manageable chunks that can be reassembled on the vehicle. This approach minimizes data loss and retransmission overhead by only retransmitting affected chunks in the event of a disruption. Checkpointing and resume capabilities further enhance fault tolerance by allowing the system to save progress at specific intervals, enabling updates to resume from the last checkpoint rather than restarting entirely. This reduces the risk of incomplete installations and accelerates the update process. Error correction codes (ECC) can detect and correct transmission errors, ensuring data integrity by identifying and requesting retransmission of corrupted parts Mayilsamy et al., 2018.

Utilizing multiple network types can also bolster connectivity reliability. Cellular networks provide extensive coverage and are well-suited for updates while the vehicle is in motion, leveraging LTE, 5G, or similar technologies to offer high bandwidth and low latency. Wi-Fi networks can be utilized when the vehicle is stationary, providing high data transfer rates ideal for large updates without relying on cellular data. A hybrid network approach, which dynamically switches between cellular and Wi-Fi based on availability and signal quality, ensures that the most suitable network is used, optimizing the update process and mitigating connectivity issues. Intelligent prioritization and scheduling of updates can further minimize the impact of connectivity constraints. Adaptive scheduling, where updates are applied based on predicted connectivity quality—such as when the vehicle is parked in areas with strong signals—reduces the risk of connectivity disruptions and ensures updates occur under optimal conditions. In cases where full connectivity is unavailable, partial updates that address critical issues can be prioritized to ensure vital functions are maintained while waiting for better connectivity for full updates Mehar et al., 2022.

Local storage of updates on the vehicle can also mitigate connectivity issues by allowing for deferred application. Updates can be pre-downloaded and stored locally when connectivity is available, to be applied later during vehicle downtime, ensuring the update process does not rely on continuous connectivity. Background downloading allows updates to be incrementally downloaded without disrupting normal vehicle operations, with full application occurring once the entire package is available, reducing the need for continuous high-bandwidth connectivity. Secure and efficient data transfer protocols are critical for enhancing the reliability of SOTA updates. Secure data transmission using encrypted and authenticated channels, such as TLS, prevents unauthorized access and ensures the integrity of the update package. Data compression techniques can reduce the size of the update package, making it more manageable for transmission over networks with variable bandwidth, speeding up the update process in environments with limited connectivity.

Redundancy and backup systems further ensure the reliability and safety of updates. A dual-system architecture allows one system to remain operational while the other is updated, providing a fallback option in case of issues during the update process. Rollback mechanisms enable the system to revert to a previous stable version if an update fails or introduces problems, ensuring continuous vehicle functionality even if the latest update is not successfully applied. Continuous monitoring and feedback loops are also essential for managing connectivity issues and improving the update process. Real-time connectivity monitoring allows the system to adjust the update process based on current conditions, pausing or delaying updates when connectivity is weak and resuming when conditions improve. Providing feedback to the user about update status and connectivity issues helps manage expectations and ensures users are aware of potential disruptions, enhancing user trust and confidence in the reliability of SOTA updates Steurich et al., 2016.

IV. SOFTWARE COMPATIBILITY AND INTEGRATION ISSUES

Ensuring compatibility between Software Over-the-Air (SOTA) updates and the diverse hardware and software ecosystem of modern vehicles is a challenge. Vehicles today are equipped with a many Electronic Control Units (ECUs), responsible for specific functions such as engine management, braking, and infotainment. These ECUs operate on various operating systems and often interact with third-party applications, creating a multi-software environment. Each component must integrate seamlessly with new software updates to avoid disruptions. This integration complexity is further exacerbated by the range of configurations across different vehicle models and versions, requiring updates to be compatible with a wide array of hardware and software setups. Ensuring that updates harmonize with these diverse configurations without causing conflicts or malfunctions is crucial for maintaining system stability and vehicle functionality Steurich et al., 2016.

Incompatibility between SOTA updates and existing vehicle systems can lead to several severe consequences, including system crashes, loss of functionality, and erratic vehicle behavior. System crashes occur when an update is incompatible with the hardware or software, causing critical vehicle systems to fail. For example, an update that disrupts the ECU managing the engine could lead to engine failure, creating hazardous driving conditions. Additionally, incompatibility can result in the loss of functionality for various vehicle features, such as braking assistance or adaptive cruise control, diminishing vehicle performance and user experience. Erratic behavior is another potential outcome, where systems may not respond as expected or exhibit unpredictable actions due to conflicting updates. These issues highlight the importance of ensuring that updates do not negatively impact the opera-

Mitigation Strategy	Description
Chunked Data Transfer	Breaks the update into smaller chunks for easier transmission and reassembly, reducing data loss and retransmission needs.
Checkpointing and Resume	Saves progress at intervals, allowing updates to resume from the last checkpoint, minimizing incomplete installations.
Error Correction Codes (ECC)	Detects and corrects transmission errors by identifying corrupted parts and requesting retransmission.
Multiple Network Types	Utilizes cellular networks for in-motion updates and Wi-Fi for stationary updates, with hybrid approaches for optimal connectivity.
Adaptive Scheduling	Schedules updates based on predicted connectivity quality, applying updates during optimal conditions to minimize disruptions.
Local Storage of Updates	Pre-downloads and stores updates locally for deferred application, allowing updates without continuous connectivity.
Background Downloading	Incrementally downloads updates without interrupting normal operations, applying the update once fully downloaded.
Secure Data Transfer Protocols	Uses encrypted and authenticated channels, such as TLS, for secure data transmission and integrity of update packages.
Data Compression Techniques	Reduces the size of update packages to facilitate transmission over networks with variable bandwidth.
Redundancy and Backup Systems	Employs dual-system architectures and rollback mechanisms for operational fallback and recovery in case of update failures.
Continuous Monitoring	Adjusts the update process based on real-time connectivity conditions, with feedback loops to inform users of update status and issues.

TABLE 4. Strategies for Mitigating Network and Connectivity Constraints in SOTA Updates

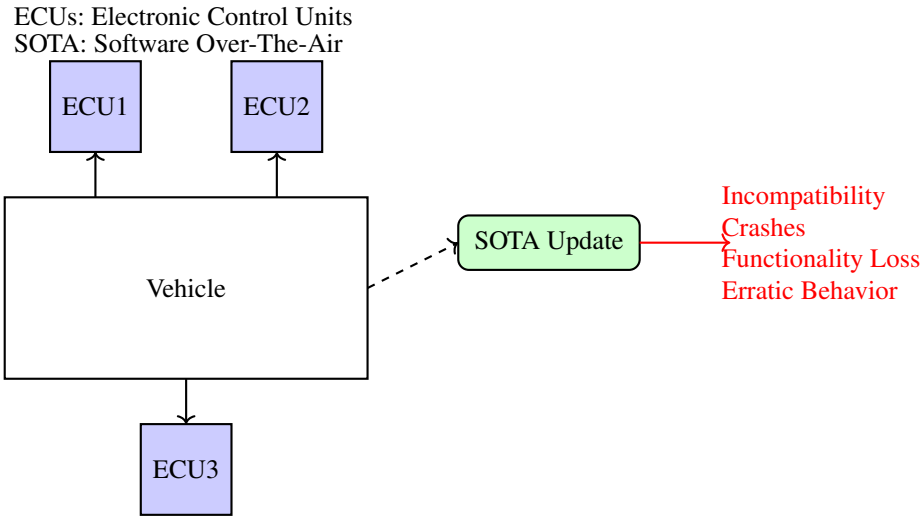

FIGURE 5. SOTA updates must be compatible with the existing hardware and software ecosystem of the vehicle. This includes diverse electronic control units (ECUs). Incompatibility can lead to system crashes, functionality loss, or erratic vehicle behavior.

tion of the vehicle's core systems or third-party applications integrated within the vehicle's software architecture.

To mitigate these software compatibility and integration issues, testing on various configurations is essential. Testing updates across multiple vehicle models and hardware setups helps identify potential compatibility problems before deployment. This includes configuration testing to ensure that updates work with different hardware and software combinations, integration testing to verify that all components function together without conflicts, and regression testing to ensure that new updates do not disrupt existing functionalities. Maintaining backward compatibility is another crucial strategy, which involves ensuring that updates are compatible with older versions of hardware and software. This practice allows vehicles with different generations of components to receive updates without issues, extending the useful life of vehicle components and ensuring a consistent user experience. Implementing compatibility layers within the software architecture can also help by adapting communication protocols and data formats, facilitating the integration

Consequence	Description
System Crashes	Incompatibility can cause critical system failures, leading to vehicle shutdowns or inability to start.
Loss of Functionality	Key vehicle functions may become inoperative or unreliable, impacting systems such as braking, steering, or infotainment.
Erratic Behavior	Vehicle systems may exhibit unpredictable behavior, such as incorrect sensor readings or erroneous system responses, compromising safety.
Degraded Performance	Vehicle performance can degrade due to conflicts between the updated software and existing systems, affecting overall vehicle efficiency.
Increased Maintenance Costs	Incompatibilities can lead to frequent repairs and updates, increasing maintenance costs and vehicle downtime.
User Frustration	Repeated issues from updates can lead to user dissatisfaction and loss of trust in the vehicle's technology and reliability.
Safety Risks	Incompatibility issues can pose safety risks by impairing critical systems, potentially leading to hazardous situations.

TABLE 5. Consequences of Incompatibility Between SOTA Updates and Existing Vehicle Systems

of new updates with legacy systems.

Adopting a modular software architecture also enhances the integration of SOTA updates by allowing independent updates to different components without affecting the entire system. Isolating software into distinct modules enables updates to be applied to specific components without disrupting other parts of the system. This modular approach ensures that communication and data exchange between modules remain consistent, even when individual components are updated. Dynamic linking techniques allow software modules to be integrated at runtime, providing flexibility in how components interact and reducing the need for system-wide updates. Continuous Integration and Deployment (CI/CD) practices also play a vital role in managing compatibility issues. Automated testing within CI/CD frameworks allows for continuous validation of updates against various configurations and scenarios, identifying compatibility issues early in the development cycle and reducing the risk of problems during deployment.

V. USER ACCEPTANCE AND EXPERIENCE

User often exhibit skepticism regarding the impact of these updates, stemming from concerns about potential disruptions, time requirements, and data costs associated with the update process. This skepticism can manifest as reluctance to accept updates, which poses a significant challenge by potentially delaying the deployment of essential software improvements and security patches. Users' concerns are particularly relevant in scenarios where updates might affect critical vehicle functions or require substantial data and time resources.

A negative perception of SOTA updates can have far-reaching consequences. Users' reluctance to accept updates can result in a failure to deploy critical software enhancements or security patches in a timely manner, exposing the vehicle to unresolved software bugs or vulnerabilities. This delay not only undermines the vehicle's performance and safety but also affects the overall reliability and user trust in the SOTA system. Furthermore, users who perceive the update process as cumbersome or intrusive may develop a general resistance to adopting new technologies, impacting the broader acceptance of digital innovations in automotive systems. The reluctance to update can also create a fragmented user base where some vehicles operate on outdated software, complicating support and maintenance efforts for manufacturers.

Mitigating these challenges requires a focus on enhancing user communication, streamlining the update process, and providing flexibility in update scheduling. Clear and transparent communication about the benefits of SOTA updates is essential to build user trust and acceptance. This involves providing detailed information on what each update entails, how it improves vehicle functionality, and addressing any concerns related to security or performance enhancements. Effective communication should highlight the immediate advantages of updates, such as improved system stability, new features, or enhanced security measures, thereby encouraging users to view updates as beneficial rather than disruptive.

Creating user-friendly interfaces for managing SOTA updates can significantly improve user experience and acceptance. Interfaces should be designed to be intuitive, providing users with straightforward options for initiating, scheduling, or postponing updates. Clear prompts and notifications should guide users through the update process, offering reassurance that their vehicle's functionality will not be adversely affected. Additionally, minimizing disruptions during the update process is crucial. Updates should be designed to occur seamlessly in the background, with mechanisms to ensure that vehicle operations are not interrupted. For example, updates can be downloaded incrementally and applied during periods when the vehicle is not in use, such as overnight or during scheduled maintenance, reducing the perceived inconvenience to the user.

Offering flexible scheduling options for updates can further enhance user acceptance by allowing users to choose

Aspect	Negative Perception Consequence	Impact
Software Deployment	Users' reluctance to accept updates	Failure to deploy critical software enhancements or security patches in a timely manner.
Vehicle Performance	Exposure to unresolved software bugs or vulnerabilities	Undermines vehicle performance and safety, affecting overall reliability.
User Trust	Perception of update process as cumbersome or intrusive	General resistance to adopting new technologies, impacting broader acceptance of digital innovations in automotive systems.
Technological Adoption	Reluctance to update software	Creates a fragmented user base, complicating support and maintenance efforts for manufacturers.

TABLE 6. Consequences of Negative Perception of SOTA Updates

times that are most convenient for them. Providing the ability to defer updates or select specific windows for installation ensures that users can integrate updates into their routines without significant disruption. This flexibility can alleviate concerns about update timing and data usage, particularly for users with limited data plans or those who frequently travel. Additionally, providing options for users to connect to Wi-Fi for updates can reduce concerns about data costs, making the update process more accessible and less burdensome.

Incorporating feedback mechanisms into the SOTA system can also play a pivotal role in improving user experience and acceptance. Providing users with channels to offer feedback on their update experience allows manufacturers to identify pain points and areas for improvement. Regularly collecting and analyzing user feedback helps in refining the update process, addressing any recurring issues, and enhancing overall satisfaction.

VI. REGULATORY AND COMPLIANCE REQUIREMENTS

Given the diverse regional and international regulations governing vehicle software and data privacy, these regulations can vary widely across different markets, influencing the methods and protocols for delivering and managing software updates. Manufacturers must fulfill requirements that dictate how updates are performed, data is handled, and user consent is obtained, necessitating a robust approach to ensure compliance across all jurisdictions in which they operate. Failure to comply with these regulations can have severe consequences, including legal penalties, mandatory recalls, or restrictions on vehicle sales in certain regions.

The impact of non-compliance with regulatory standards can be profound, affecting both the operational and financial aspects of automotive manufacturers. Legal penalties may include substantial fines for breaching data privacy laws or failing to adhere to safety standards. For instance, the European Union's General Data Protection Regulation (GDPR) imposes stringent requirements on data handling and user consent, with non-compliance resulting in heavy fines that can amount to a significant percentage of a company's global revenue. In addition to financial penalties, manufacturers may face mandatory recalls of non-compliant vehicles, which can incur considerable costs and damage the brand's reputation. Moreover, non-compliance can lead to restrictions or outright bans on vehicle sales in specific markets, limiting the manufacturer's ability to compete and operate globally. These repercussions underscore the critical importance of integrating regulatory compliance into the development and deployment of SOTA updates.

Mitigating the risks associated with regulatory and compliance challenges requires a proactive strategy. One key approach is to stay updated with regulatory changes and emerging standards in the automotive industry. This involves monitoring legislative developments and participating in industry forums and working groups that shape regulatory frameworks. Engaging with regulatory bodies during the development of SOTA systems is also crucial. Establishing open channels of communication with regulators allows manufacturers to seek guidance on compliance issues, clarify ambiguities in regulations, and obtain approvals or certifications for their update mechanisms. This engagement helps ensure that the update process aligns with regulatory expectations and can preempt potential compliance issues before they arise.

Ensuring that SOTA updates meet compliance standards across different jurisdictions involves implementing standardized processes that can be adapted to various regulatory environments. This includes developing a comprehensive compliance framework that incorporates the requirements of different regions into the update lifecycle. Key elements of this framework should include data protection measures that comply with privacy laws such as GDPR, the California Consumer Privacy Act (CCPA), and similar regulations in other markets. These measures should ensure that user data is collected, processed, and stored in accordance with legal standards, with mechanisms for obtaining explicit user consent and providing transparency about data usage. Additionally, safety regulations require that updates do not compromise vehicle safety and must undergo rigorous testing and certification processes to verify compliance. This includes adhering to standards such as the United Nations Economic Commission for Europe (UNECE) WP.29 regulations, which mandate cybersecurity and software update management requirements for automotive systems.

Developing and maintaining detailed documentation and

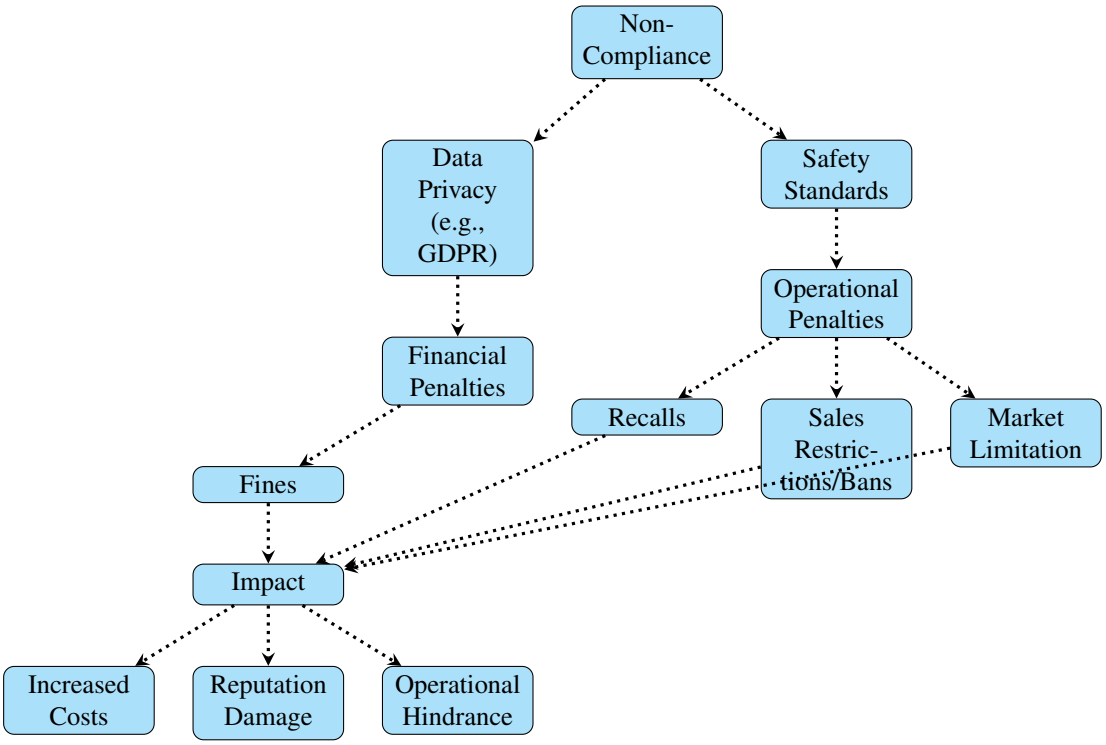

FIGURE 6. Impact of Non-Compliance with Regulatory Standards on Automotive Manufacturers

Key Approach	Description	Impact
Staying Updated with Regulatory Changes	Monitoring legislative developments and participating in industry forums and working groups. Engaging with regulatory bodies during SOTA system development to seek guidance, clarify ambiguities, and obtain approvals.	Ensures alignment with regulatory expectations and preempts potential compliance issues. Facilitates smooth integration of new regulations into update processes.
Implementing Standardized Processes	Developing a compliance framework adaptable to various regulatory environments. Incorporating data protection measures in line with GDPR, CCPA, and other privacy laws. Ensuring updates do not compromise vehicle safety by adhering to standards such as UNECE WP.29.	Streamlines compliance across jurisdictions, reducing legal risks. Enhances data protection and vehicle safety, promoting user trust.
Developing Detailed Documentation and Audit Trails	Capturing the entire update process, including development, testing, deployment, and user notification. Maintaining an audit trail of updates for transparency and traceability, demonstrating compliance to regulatory authorities.	Provides evidence of compliance, facilitates audits and investigations, and supports traceability and accountability in update management.
Standardizing Update Protocols	Adopting uniform practices for update delivery, data handling, and user interaction that meet the highest regulatory standards in any target market. Implementing robust cybersecurity measures, including encryption, authentication, and integrity checks, to protect against cyber threats.	Ensures consistent application of compliance measures across regions. Enhances cybersecurity, reducing the risk of unauthorized access and manipulation of vehicle software.

TABLE 7. Approaches to Mitigating Regulatory and Compliance Risks in SOTA Updates

audit trails for SOTA updates is another essential component of regulatory compliance. Documentation should capture the entire update process, including the development, testing, deployment, and user notification phases. This documentation serves as evidence of compliance and can be reviewed by regulatory authorities in case of audits or investigations. Maintaining an audit trail of updates provides transparency into the changes made to vehicle software, including version histories, update content, and the timing of deployments. This audit trail helps demonstrate that updates are performed in accordance with regulatory requirements and can facilitate traceability in the event of issues or disputes.

Standardizing update protocols to align with regulatory requirements across different regions can also ease the compliance process. Standardization involves adopting uniform practices for update delivery, data handling, and user interaction that meet the highest regulatory standards applicable in any of the target markets. This approach simplifies the update process and ensures that compliance measures are consistently applied, reducing the risk of regional discrepancies. Implementing robust cybersecurity measures is integral to compliance, as regulations increasingly mandate protections against cyber threats in vehicle systems. Ensuring that updates are delivered securely, with encryption, authentication, and integrity checks, mitigates the risk of unauthorized access or manipulation of vehicle software, aligning with regulatory expectations for cybersecurity.

VII. CONCLUSION

The integration of Software Over The Air (SOTA) updates into the automotive sector presents significant advantages, including cost savings from inexpensive over-the-air bug fixes and the ability to enhance vehicle capabilities throughout their lifecycle. Despite these benefits, the application of SOTA to safety-critical automotive functions is fraught with considerable challenges that must be examined and addressed before widespread implementation.

The challenge of securing SOTA updates against cyber threats involves protecting the update process from interception, unauthorized access, and manipulation. The potential impacts of failing to secure SOTA updates include compromising vehicle safety, enabling theft, and exposing sensitive user data.

Network and connectivity constraints present significant challenges to the reliable deployment of SOTA in the automotive industry. These challenges arise from the variability in network quality due to geographic location, coverage, and interference, impacting the continuity and integrity of the update process. The consequences of connectivity disruptions include incomplete installations, software corruption, and system instability, which can compromise vehicle functionality and safety. To address these challenges, an approach that includes fault-tolerant update mechanisms, multi-network utilization, intelligent scheduling, local storage, secure data protocols, redundancy, and continuous monitoring is necessary.

Addressing software compatibility and integration issues is also important for the successful deployment of Software Over-the-Air (SOTA) updates in vehicles. The diverse ecosystem of modern vehicle systems necessitates rigorous testing, backward compatibility, and a modular software architecture to ensure seamless integration. The impacts of compatibility issues, including system crashes, functionality loss, and erratic behavior, underscore the need for robust mitigation strategies. Implementing testing, maintaining backward compatibility, adopting modular architectures, and using CI/CD practices, the automotive industry can effectively manage these challenges.

Gaining user trust and ensuring a positive experience with Software Over-the-Air (SOTA) updates is essential for the successful implementation of this technology in automotive systems. Addressing user skepticism through clear communication, user-friendly interfaces, minimal disruption during updates, and flexible scheduling options can significantly enhance user acceptance. By prioritizing user experience and proactively managing concerns related to update impact, time, and data costs, manufacturers can facilitate a smoother transition to SOTA-enabled vehicles and ensure that critical software improvements and security patches are effectively deployed. This user-centric approach not only supports the technical efficacy of SOTA updates but also promotes broader adoption and trust in the digital advancements within the automotive industry.

Adhering to regulatory and compliance requirements for Software Over-the-Air (SOTA) updates is also an essential aspect of modern automotive software management. The impacts of non-compliance, including legal penalties, recalls, and sales restrictions, highlight the critical need for robust compliance measures. The solutions include staying updated with regulatory changes, engaging with regulatory bodies during development, and implementing a standardized compliance framework, manufacturers can effectively manage regulatory risks. Detailed documentation, audit trails, and standardized protocols further support compliance efforts, ensuring that SOTA updates are delivered in a manner that meets legal and safety requirements across all jurisdictions. This study identifies the main challenges in deploying Software Over-the-Air (SOTA) updates for vehicles, including cybersecurity risks, network connectivity issues, and software compatibility concerns. It offers a framework for addressing these challenges to facilitate the reliable implementation of SOTA updates. The study's insights aim to support improved vehicle performance, security, and functionality through effective SOTA deployment.

VECTORAL PUBLISHING POLICY

VECTORAL maintains a strict policy requiring authors to submit only novel, original work that has not been published previously or concurrently submitted for publication elsewhere. When submitting a manuscript, authors must provide a comprehensive disclosure of all prior publications and ongoing submissions. VECTORAL prohibits the publication

of preliminary or incomplete results. It is the responsibility of the submitting author to secure the agreement of all co-authors and obtain any necessary permissions from employers or sponsors prior to article submission. The VECTORAL takes a firm stance against honorary or courtesy authorship and strongly encourages authors to reference only directly relevant previous work. Proper citation practices are a fundamental obligation of the authors. VECTORAL does not publish conference records or proceedings.

VECOTORAL PUBLICATION PRINCIPLES

Authors should consider the following points:

1) To be considered for publication, technical papers must contribute to the advancement of knowledge in their field and acknowledge relevant existing research.
2) The length of a submitted paper should be proportionate to the significance or complexity of the research. For instance, a straightforward extension of previously published work may not warrant publication or could be adequately presented in a concise format.
3) Authors must demonstrate the scientific and technical value of their work to both peer reviewers and editors. The burden of proof is higher when presenting extraordinary or unexpected findings.
4) To facilitate scientific progress through replication, papers submitted for publication must provide sufficient information to enable readers to conduct similar experiments or calculations and reproduce the reported results. While not every detail needs to be disclosed, a paper must contain new, usable, and thoroughly described information.
5) Papers that discuss ongoing research or announce the most recent technical achievements may be suitable for presentation at a professional conference but may not be appropriate for publication.

References

Andrade, C. E., Byers, S. D., Gopalakrishnan, V., Halepovic, E., Majmundar, M., Poole, D. J., Tran, L. K., & Volinsky, C. T. (2017). Managing massive firmware-over-the-air updates for connected cars in cellular networks. *Proceedings of the 2nd ACM international workshop on smart, autonomous, and connected vehicular systems and services*, 65–72.

Blázquez, E., Pastrana, S., Feal, Á., Gamba, J., Kotzias, P., Vallina-Rodriguez, N., & Tapiador, J. (2021). Trouble over-the-air: An analysis of fota apps in the android ecosystem. *2021 IEEE Symposium on Security and Privacy (SP)*, 1606–1622.

Bulmus, A., Freiwald, A., & Wunderlich, C. (2017). *Over the air software update realization within generic modules with microcontrollers using external serial flash* (tech. rep.). SAE Technical Paper.

Chandra, H., Anggadjaja, E., Wijaya, P. S., & Gunawan, E. (2016). Internet of things: Over-the-air (ota) firmware update in lightweight mesh network protocol for smart urban development. *2016 22nd Asia-Pacific Conference on Communications (APCC)*, 115–118.

Doddapaneni, K., Lakkundi, R., Rao, S., Kulkarni, S. G., & Bhat, B. (2017). Secure fota object for iot. *2017 IEEE 42nd Conference on Local Computer Networks Workshops (LCN Workshops)*, 154–159.

El Jaouhari, S., & Bouvet, E. (2022). Secure firmware over-the-air updates for iot: Survey, challenges, and discussions. *Internet of Things*, *18*, 100508.

Frey, M., Bjelaković, I., & Stańczak, S. (2021). Over-the-air computation in correlated channels. *IEEE Transactions on Signal Processing*, *69*, 5739–5755.

Ghosal, A., Halder, S., & Conti, M. (2022). Secure over-the-air software update for connected vehicles. *Computer Networks*, *218*, 109394.

Guissouma, H., Diewald, A., & Sax, E. (2019). A generic system for automotive software over the air (sota) updates allowing efficient variant and release management. *Information Systems Architecture and Technology: Proceedings of 39th International Conference on Information Systems Architecture and Technology–ISAT 2018: Part I*, 78–89.

Halder, S., Ghosal, A., & Conti, M. (2020). Secure over-the-air software updates in connected vehicles: A survey. *Computer Networks*, *178*, 107343.

Hess, T., Bol, D., & Sadre, R. (2020). Ultra-low-power over-the-air-update in secure lorawan networks. *Ecole polytechnique de Louvain, Université catholique de Louvain*.

Malik, A. W., Rahman, A. U., Ahmad, A., & Santos, M. M. D. (2022). Over-the-air software-defined vehicle updates using federated fog environment. *IEEE Transactions on Network and Service Management*, *19*(4), 5078–5089.

Mayilsamy, K., Ramachandran, N., & Raj, V. S. (2018). An integrated approach for data security in vehicle diagnostics over internet protocol and software update over the air. *Computers & Electrical Engineering*, *71*, 578–593.

Mehar, M., Waghole, A., Bharti, A., & Behre, P. (2022). Over the air (ota) update system–a systematic.

Sery, T., Shlezinger, N., Cohen, K., & Eldar, Y. C. (2021). Over-the-air federated learning from heterogeneous data. *IEEE Transactions on Signal Processing*, *69*, 3796–3811.

Shavit, M., Gryc, A., & Miucic, R. (2007). *Firmware update over the air (fota) for automotive industry* (tech. rep.). SAE Technical Paper.

Steurich, B., Scheibert, K., Freiwald, A., & Klimke, M. (2016). *Feasibility study for a secure and seamless integration of over the air software update capability in an advanced board net architecture* (tech. rep.). SAE Technical Paper.

Villegas, M. M., Orellana, C., & Astudillo, H. (2019). A study of over-the-air (ota) update systems for cps

and iot operating systems. *Proceedings of the 13th European Conference on Software Architecture-Volume 2*, 269–272.