# **Milestone 02: Literature & Technology Review**

**Literature Review**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Aspect | Security Analysis of Android Automotive    Research Paper 1 | Vulnerability Analysis of Android Auto Infotainment Apps  Research Paper 2 | Penetration Testing of an In-Vehicle Infotainment System  Research Paper 3 | Vulnerability Analysis of an Automotive Infotainment System's Wi-Fi Capabilities  Research Paper 4 | Identifying potential vulnerabilities in a generic Android based infotainment systems Emulated within Android Studio |
| Objective | Security analysis of Android Automotive platform | Analyze vulnerabilities in Android Auto apps | Penetration testing of in-vehicle infotainment systems | To analyze and identify vulnerabilities related to Wi-Fi capabilities in automotive infotainment systems. The focus is on understanding how wireless communication can be exploited by attackers to gain unauthorized access or disrupt system functionality | Identifying potential vulnerabilities in Android-based infotainment systems |
| Methodology | Examination of system architecture and countermeasures | Static and dynamic analysis of app behaviors | Penetration testing and security evaluation | The paper employs a combination of static and dynamic analysis techniques to examine the Wi-Fi functionalities within automotive infotainment systems. This includes:   * Reviewing the system architecture and Wi-Fi implementation. * Conducting penetration tests and simulated attacks on the Wi-Fi interfaces. * Analyzing data packets and communication protocols to identify potential vulnerabilities. | Emulating system in Android Studio, using ZAP and Nessus |
| Tools Used | System architecture analysis tools | Static and dynamic analysis tools | Penetration testing tools | The research utilizes various security analysis and testing tools, including:   * Wireshark for packet analysis. * Aircrack-ng for Wi-Fi penetration testing. * Custom scripts and tools for simulating attacks and capturing traffic data | ZAP, Nessus |
| Vulnerabilities Identified | Risks from third-party apps, network vulnerabilities | App vulnerabilities like data leakage, insecure storage | Multiple security issues including data breaches and system compromise | The analysis reveals several key vulnerabilities, including:   * Insecure Wi-Fi configurations (e.g., weak passwords, open networks). * Susceptibility to common Wi-Fi attacks such as Man-in-the-Middle (MitM), DE authentication, and packet injection. * Inadequate encryption and authentication mechanisms. * Potential for unauthorized access and control over infotainment system functionalities | Vulnerabilities found using ZAP and Nessus |
| Severity Assessment | Moderate to high, depending on the app and network integration | Varies; significant for poorly designed apps | High severity due to potential system-wide impacts | The identified vulnerabilities are assessed based on their potential impact on system security and user safety:   * High severity: Vulnerabilities that allow remote code execution or unauthorized control of critical system functions. * Medium severity: Vulnerabilities that enable data interception or manipulation without direct control over the system. * Low severity: Vulnerabilities that pose minimal risk to system operation but could lead to privacy issues. | Severity assessment based on findings |
| Mitigation Strategies | Stronger app vetting, enhanced system defenses | Better app design practices, secure storage solutions | Implementing robust security measures, regular penetration tests | The paper suggests several mitigation strategies to address the identified vulnerabilities:   * Implementing strong encryption and authentication protocols for Wi-Fi connections. * Regularly updating and patching Wi-Fi firmware and software. * Enforcing secure Wi-Fi configurations, including strong passwords and disabling unnecessary network features. * Conducting regular security audits and penetration tests to identify and address new vulnerabilities. | Suggested mitigation strategies |
| Strengths | Comprehensive system-level analysis | Detailed app-level vulnerability insights | Practical approach with real-world penetration testing | ·  Comprehensive analysis of Wi-Fi capabilities and vulnerabilities.  ·  Use of practical penetration testing techniques to demonstrate real-world risks.  ·  Clear identification of high-risk vulnerabilities and actionable mitigation strategies. | Specific focus on Android-based infotainment vulnerabilities |
| Weaknesses | Limited to theoretical analysis, not practical testing | Focused only on app vulnerabilities, not system-wide | Limited scope to a single system | ·  Limited scope focusing only on Wi-Fi vulnerabilities, potentially overlooking other critical aspects of infotainment system security.  ·  Reliance on specific tools and methods which may not cover all potential attack vectors. | Emulated environment may not capture all real-world variables |
| Threats | Increased complexity of in-vehicle networks | Evolving app development practices | Constantly evolving threat landscape | ·  Rapidly evolving attack techniques and new vulnerabilities emerging in Wi-Fi and automotive systems.  ·  Increased complexity of infotainment systems integrating multiple wireless technologies, potentially introducing new security risks. | Emerging new vulnerabilities and attack vectors |
| Challenges | Ensuring comprehensive coverage of all potential threats | Keeping up with new app vulnerabilities | Staying updated with the latest penetration testing tools and techniques | · Keeping up with the fast pace of technological advancements and new security threats in the automotive industry.  · Ensuring comprehensive coverage of all potential vulnerabilities, given the complex and varied nature of infotainment systems.  · Balancing security with usability, ensuring that security measures do not overly complicate user experience or system functionality. | Accurate emulation of real-world conditions and maintaining up-to-date threat knowledge |

1. **References:**  
   Research Paper1: Security Analysis of Android Automotive  
   Link: [(PDF) Security Analysis of Android Automotive (researchgate.net)](https://www.researchgate.net/publication/340632296_Security_Analysis_of_Android_Automotive)
2. Research Paper2: Penetration Testing of an In-Vehicle Infotainment System  
   Link: [Penetration Testing of an In-Vehicle Infotainment System (diva-portal.org)](https://www.diva-portal.org/smash/get/diva2:1708534/FULLTEXT01.pdf)
3. Research Paper3: Vulnerability Analysis of Android Auto Infotainment Apps  
   Link: [Vulnerability analysis of Android auto infotainment apps | Proceedings of the 15th ACM International Conference on Computing Frontiers](https://dl.acm.org/doi/abs/10.1145/3203217.3203278)
4. Research Paper4: Vulnerability Analysis of an Automotive Infotainment System's Wi-Fi Capabilities  
   Link: [Vulnerability Analysis of an Automotive Infotainment System's WIFI Capability | IEEE Conference Publication | IEEE Xplore](https://ieeexplore.ieee.org/abstract/document/8673049)

**Technology Review**

Technology Review: Identifying Potential Vulnerabilities in a generic Android based Infotainment Systems Emulated within Android Studio (Beyond ZAP and Nessus) While ZAP and Nessus are powerful tools for assessing car infotainment security, incorporating alternative technologies can provide a more comprehensive and targeted approach. Here's a review of four such alternatives:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ASPECT | CAN Bus Analyzer | Wire shark | Security Onion Live | Mobile Device Management (MDM) Tools with Security Features |
| Technology Overview | The Controller Area Network (CAN) bus is a communication protocol commonly used in car electronics, including infotainment systems. A CAN bus analyzer allows you to monitor and analyze traffic on the CAN bus | Wire shark is a free and open-source network protocol analyzer that can capture and analyze network traffic across various interfaces, including Wi-Fi connections commonly used in car infotainment systems. | Security Onion Live is a free and open-source Linux distribution pre-configured with a suite of security monitoring tools. It offers capabilities for network intrusion detection, packet capture, and security logging analysis. | MDM tools are used for managing and securing mobile devices within an organization. Some MDM solutions offer security features like application whitelisting, data encryption, and remote wipe capabilities, which can be extended to car infotainment systems running on Android |
| Security Testing Capabilities | Helps identify suspicious activity on the CAN bus, potentially revealing attempts to manipulate vehicle controls or extract sensitive data.  Can be used to analyze the security of CAN bus communication protocols and identify vulnerabilities like weak encryption or lack of authentication. | Useful for analyzing network traffic between the car infotainment system and external devices or networks (e.g., smartphones connecting via Wi-Fi). Can be used to identify vulnerabilities related to insecure wireless connections, data transmission protocols, or unauthorized access attempts | Provides a comprehensive environment for monitoring network traffic generated by the car infotainment system, potentially revealing malicious activity or unauthorized access attempts. Can be used to correlate data from various sources (ZAP, Wireshark) to create a holistic view of the security posture. | MDM can be used to enforce security policies on the car infotainment system, mitigating risks associated with unauthorized app installations or data leakage. Can provide centralized monitoring and management of security configurations within the car infotainment system. infotainment system. |
| Strengths | Provides deep insights into communication between car components, crucial for assessing infotainment system interaction with vehicle control systems. Useful for uncovering attacks targeting CAN bus vulnerabilities, which are becoming increasingly prevalent. | Versatile tool for analyzing network traffic across various protocols, not limited to the car infotainment system. Offers deep packet inspection capabilities for detailed analysis of network communication content. | Offers a centralized platform for managing various security tools, streamlining the analysis process. Provides valuable insights into overall network security beyond just the car infotainment system. | Offers proactive security measures beyond vulnerability assessment, improving the overall security posture of the car infotainment system. Useful for managing security across a fleet of vehicles with car infotainment systems. |
| Weaknesses | Requires specialized hardware and software to capture and analyze CAN bus traffic. May require a deeper understanding of CAN bus protocols for effective interpretation of captured data. | May require advanced network analysis skills to effectively interpret captured data and identify security issues. Might not be as efficient as a dedicated CAN bus analyzer for car-specific communication protocols | Setting up and configuring Security Onion Live requires some technical expertise in security tools and Linux administration. May be resource-intensive for older computers, impacting performance during testing | Weaknesses: Implementing MDM requires collaboration with the car manufacturer to ensure compatibility with the infotainment system. May not be a readily available option for individual researchers or small teams due to potential licensing costs |