Lab Assignment 6: Research Writing on Mobile and Embedded Device Security

Instructions

Choose any THREE of the prompts below. Write 2–3 well-developed paragraphs per prompt. You must cite at least one credible source per prompt (e.g., news articles, cybersecurity reports, peer-reviewed journals, government regulations, or official tech documentation).

Research Writing Prompts (Choose 3)

1. **Mobile Malware in the Wild**
   Research a recent case of mobile malware (e.g., spyware, malicious apps, or trojans). How did it spread? What was its impact? What measures were taken to contain or remove it?
2. **BYOD Risks and Policy Design**
   Examine the risks associated with Bring Your Own Device (BYOD) in workplace environments. What should a secure BYOD policy include? Provide examples from a real organization, if possible.
3. **Sideloading and App Store Alternatives**
   Explore the growing trend of sideloading apps. What platforms allow it? What are the associated risks, and what protections can users or organizations implement?
4. **Embedded Systems in Critical Infrastructure**
   Investigate how embedded systems like SCADA are used in sectors like energy or transportation. What are the security challenges in managing these systems? Cite real-world examples of past attacks.
5. **IoT Devices and Home Network Security**
   Review common vulnerabilities in consumer IoT devices (e.g., smart cameras, thermostats, or voice assistants). How can consumers secure their home networks? Include any vendor recommendations.
6. **Mobile Device Management (MDM) Solutions**
   Compare two MDM solutions in terms of features, ease of use, and effectiveness. Which one would you recommend for a mid-sized business and why?
7. **Cryptographic Challenges in IoT Devices**
   Explore the problem of implementing strong cryptography in low-power IoT devices. What are the trade-offs, and what new technologies are helping to bridge the gap?
8. **Secure Coding Practices for Embedded Applications**
   Identify secure coding techniques specifically used for embedded or mobile applications. How do these practices reduce vulnerabilities like buffer overflows or improper memory handling?