Conf42.com DevOps 2025

Securing Embedded Systems in IoT: A Practical DevOps Approach

Name: Victor Nosakhare Oriakhi

Agenda

Introduction to IoT and Embedded Systems

Security Challenges in Embedded Systems

The Role of DevOps in Securing Embedded Systems

Securing the Development Lifecycle

Securing Post-Deployment IoT Devices

Balancing Innovation and Security

Best Practices for Securing Embedded IoT Systems

Key Takeaways

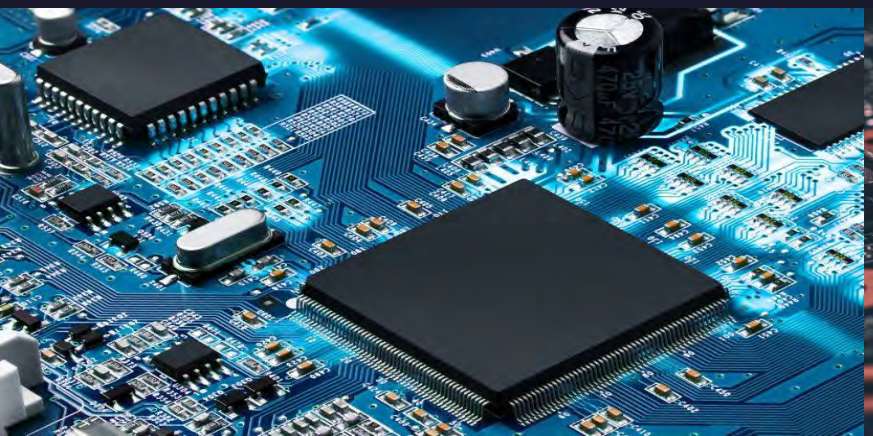
Introduction

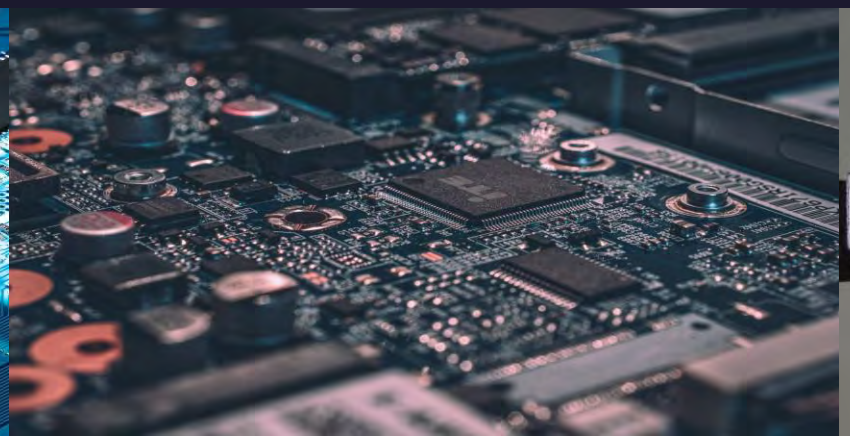
Overview of Embedded Devices and The internet of Things:

Embedded devices form the core of modern robotics and IoT applications, working silently to perform specific tasks within larger systems.

With the rise of connected technology, embedded devices have become integral to fields like healthcare, manufacturing, and smart infrastructure.

The IOT refers to a network of interconnected physical devices embedded with sensors, software and other technologies that enable them to collect, exchange and act on data over the internet





Practical Examples:

Smart Devices (e.g., thermostats, wearables)
Robotics (e.g., autonomous drones, medical robots)
Industrial Automation (e.g., factory sensors, process control systems)

IoT Growth: Over 25 billion connected devices expected by 2030.

Security Challenges in Embedded Systems

Hardware-Level Risks:
Attackers may exploit physical access to devices, conducting attacks like side-channel and tampering attacks, where signals or emissions reveal sensitive information.

Firmware and Software Vulnerabilities:
Outdated firmware and software with hardcoded credentials or weak encryption are easily exploited. Firmware updates, if not secure, can also introduce vulnerabilities.

Communication Protocol Weaknesses:
IoT devices often rely on common protocols (e.g., Wi-Fi, Bluetooth), which can be intercepted if not encrypted, leaving devices open to data leaks and remote manipulation.

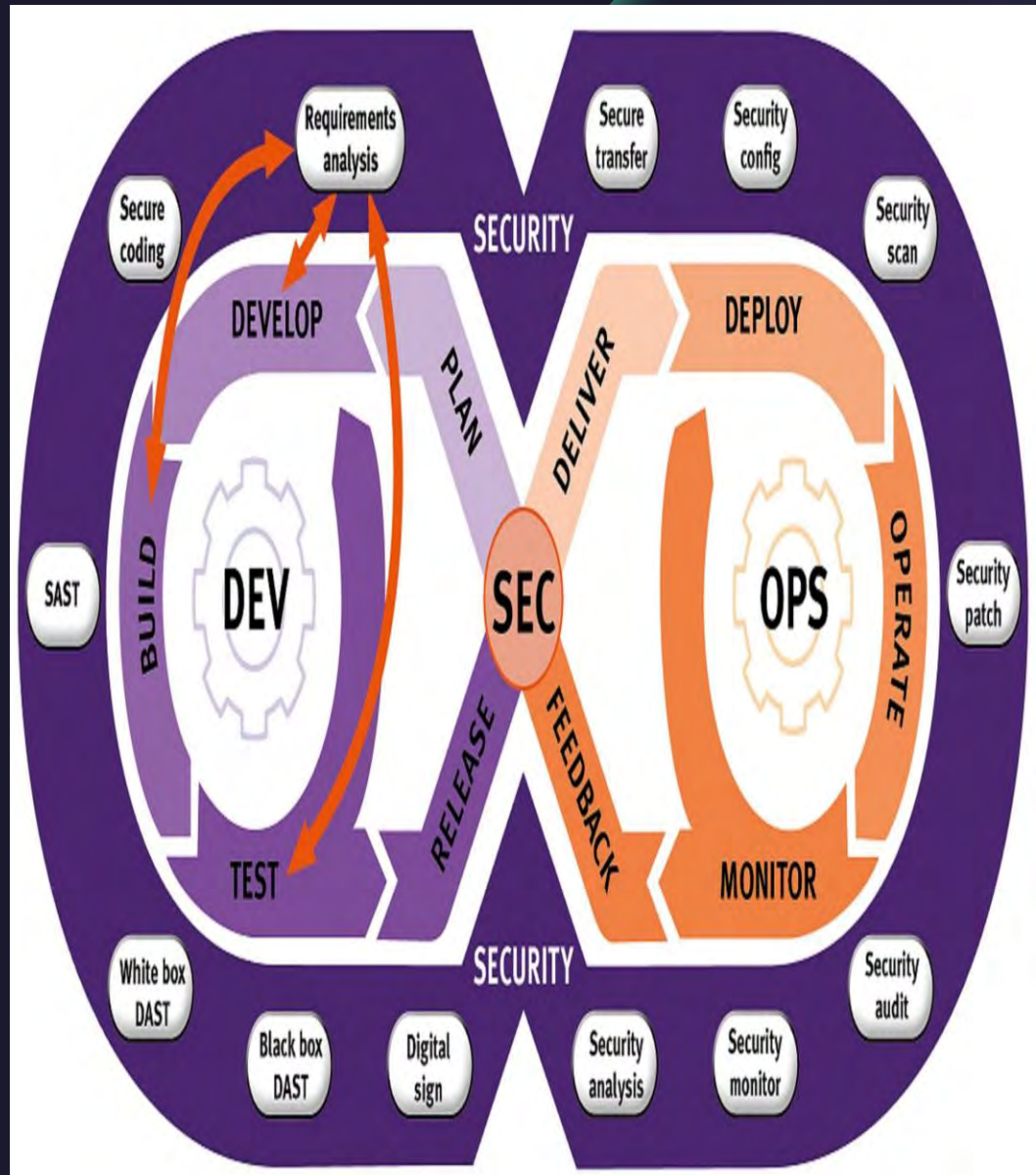
The Role of DevOps in Securing Embedded Systems

What is DevOps?: DevOps is the integration of development (Dev) and operations (Ops) through automation, continuous integration, and continuous delivery.

DevOps for Embedded Systems: Applying DevOps principles to embedded systems development to improve security, agility, and automation.

Benefits:

- **Continuous Monitoring**: Real-time detection of security vulnerabilities.
- **Automated Security Testing**: Ensuring security validation is part of the CI/CD pipeline.
- **Faster Issue Resolution**: Addressing security flaws quickly through automated processes.



Securing the Development Lifecycle

Security by Design: Implement security from the design phase, not as an afterthought.

- **Use of Encryption**: Secure communication using encryption protocols (e.g., TLS).
- **Secure Boot**: Ensuring devices start with trusted software only.
- **Trusted Hardware**: Using Trusted Platform Modules (TPMs) to protect hardware integrity.

CI/CD Pipelines for Hardware:

Automating hardware and software tests for vulnerabilities.

Using penetration testing and vulnerability scanning tools.

Automated Security Validation: Ensure security is validated in every phase of the development process.

Securing Post-Deployment IoT Devices

Ongoing Monitoring: Set up continuous monitoring of deployed devices for unusual activities.

- **Log and Alert Systems**: Implement logging and automated alerts for potential security breaches.

Patch Management:

- **OTA (Over-the-Air) Updates**: Regular firmware and software updates to patch vulnerabilities.
- **Remote Security Patching**: Secure mechanisms for updating devices without physical intervention.

Long-Term Security: Regularly assess and update device security throughout its lifecycle.

Balancing Innovation and Security

Innovation Without Compromise:

- Security practices should be integrated in a way that does not hinder the performance or innovation of IoT devices.
- Security and innovation can coexist if built into the development lifecycle from the beginning.

Security as an Enabler:

- Devices with strong security features are more likely to be trusted and adopted.
- Secure systems open opportunities for new applications and markets, increasing device scalability.

Best Practices for Securing Embedded IoT Systems

Secure Development Lifecycles (SDLC): Ensure security is embedded at each phase of the development cycle.

- Conduct regular threat modeling and risk assessments.

Use Secure Communication Protocols: Encrypt communication using HTTPS, TLS, or other secure protocols.

Automate Security Testing: Integrate security testing into your CI/CD pipeline for faster vulnerability detection.

Incident Response Plans: Develop clear, actionable protocols for responding to security incidents.

Key Takeaways

Embed Security Early: Integrate security in the design and development phases to mitigate risks early.

CI/CD for Hardware: Implement continuous integration and continuous delivery practices for hardware security.

Monitor and Patch: Regularly monitor and patch devices after deployment to maintain security.

Security and Innovation: Secure systems enable innovation, driving greater user trust and device scalability.

CONCLUSION

IoT and embedded systems—we can create systems that are both innovative and resilient. Prioritizing security builds trust, drives scalability, and ensures a safer future for everyone. are revolutionizing our world, but they bring critical security challenges. By integrating security into the development process with DevOps principles—like secure design, continuous monitoring, and regular updates

Thank you very much!