Universidad Tecnológica Nacional - Facultad Regional Córdoba
Cátedra de Ingeniería y Calidad de Software
Poster Científico

Plan de despliegue de firmware para red de sensores IoT en plantas industriales

Autores: Grupo 12 - ISW 4k3 - 2025

Barrionuevo Daniel - Castro Martin - Chaile Emmanuel - Cornejo Francisco - Freytes Agustin - Höhlke Augusto - Liendo Juan - Silvestri Brian - Virinni Bruno

Introducción

En plantas industriales distribuidas, los sensores IoT deben actualizar su firmware de forma remota, segura y sin depender de conectividad estable. Nuestro proyecto propone un plan de despliegue que minimiza riesgos, reduce costos operativos y evita "brickear" dispositivos críticos.

En algunas fábricas los sensores IoT tienen conectividad limitada: algunos se conectan una vez al día y otros solo a redes locales. Una actualización fallida puede inutilizar un dispositivo y frenar mediciones críticas.

Desafío

El desafío es diseñar un despliegue seguro, resistente a interrupciones y operativo incluso con cortes de red, con versionado estricto, rollback, verificación criptográfica y una estrategia por lotes para minimizar riesgos.

Estrategia

Los distintos componentes (backend, consola y firmware) requieren estrategias propias. En los sensores de campo se usa actualización incremental, que reduce el tamaño del paquete y el tiempo de transferencia. **Energy-aware Incremental OTA Update for Flash-based Batteryless IoT Devices** respalda este enfoque por su eficiencia y confiabilidad. En conjunto, permite un despliegue más seguro y adecuado a las condiciones industriales reales.

Automatización, CI/CD

El proceso se apoya en una pipeline DevOps: GitHub Actions construye y prueba automáticamente el firmware, aplicando control de configuración y versionado semántico.

El despliegue utiliza comunicación segura (TLS), firma digital del firmware y verificación antes de instalar. Los dispositivos usan dos particiones (A/B) para asegurar rollback automático si la actualización falla.

El monitoreo con Grafana permite detectar errores tempranos y validar el comportamiento en campo, reduciendo tiempos de respuesta y evitando desplazamientos técnicos innecesarios.

Plan de release

1. Preparación del artefacto:
- Construcción automática mediante CI/CD.
- Asignación de versión (1.3.0) y firma digital.
- Generación del paquete incremental desde 1.2.0.

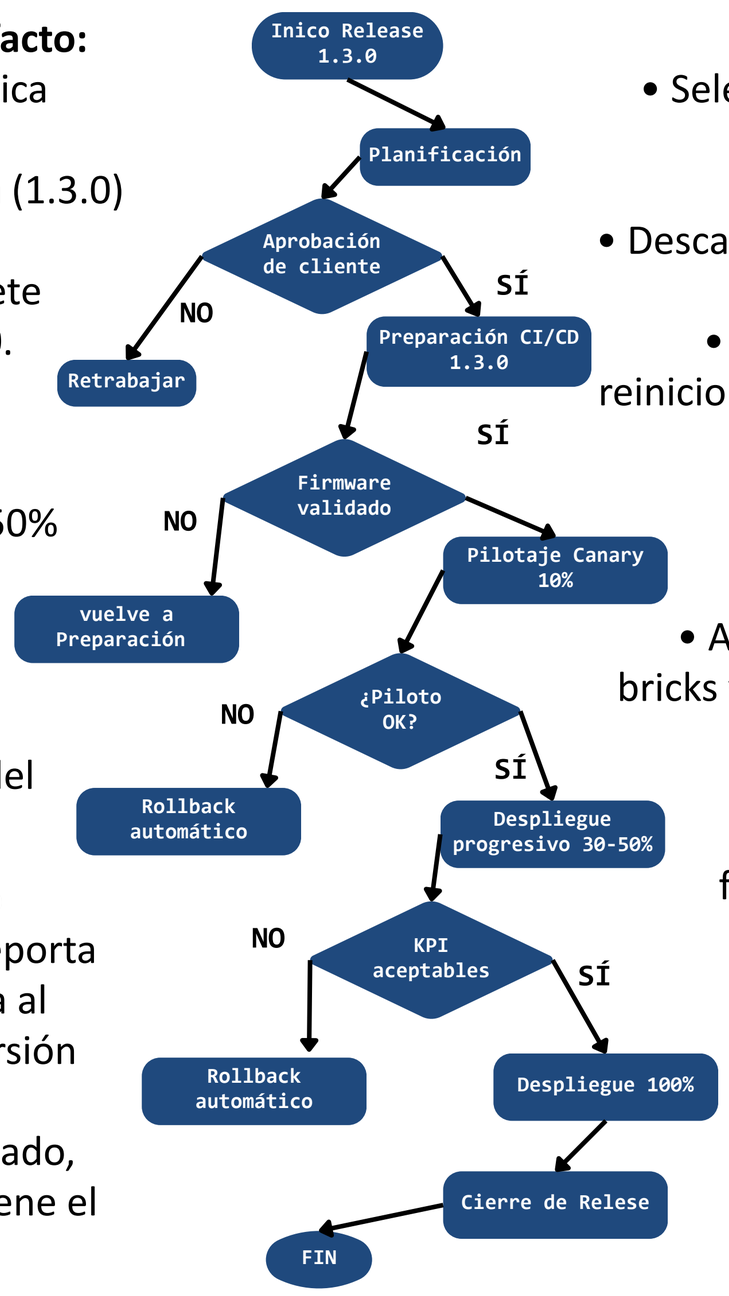
2. Pilotaje inicial
- Selección de un lote reducido de sensores con buena conectividad.
- Descarga del delta y verificación de integridad.
- Instalación en partición B, reinicio y validación durante 24 h.

3. Escalado progresivo
- Si no se detectan errores, despliegue al 50% restante.
- Nueva ventana de observación.
- Finalmente, actualización al 100% del parque.

4. Seguridad y resiliencia
- Identity per device y autenticación estricta.
- Arquitectura A/B que evita bricks y habilita rollback seguro.

5. Rollback automático
- Si el dispositivo no reporta "update success" o falla al arrancar, vuelve a la versión 1.2.0.
- El backend marca estado, registra métricas y detiene el despliegue.

6. Monitoreo y cierre
- Validación de telemetría, funcionamiento y consumo energético.
- Registro de versiones instaladas por planta.
- Documentación de resultados y lecciones aprendidas.



Conclusión

En síntesis, nuestro plan de despliegue reduce el riesgo de fallas en firmware, evita costosos desplazamientos técnicos y mejora la confiabilidad del sistema.

Además, integra principios de DevOps, seguridad y automatización adaptados a entornos industriales con baja conectividad.

Bibliografía

Bakhshi, T. (2024). "A Review of IoT Firmware Vulnerabilities and Auditing". Sensors, 24(2), 708.

Wei W., Banerjee J., Islam S., Pan C., & Xie M. (2024). Energy-aware Incremental OTA Update for Flash-based Batteryless IoT Devices.

"Over-the-Air Software Updates in the Internet of Things: An Overview of Key Principles". (2020). ResearchGate.

Info Adicional

Para acceder a la información completa de la investigación por favor escanear el código QR