Práctica 1

Nombres de los autores

September 2025

1 Resumen

El Trabajo de Fin de Máster propone el diseño, implementación y evaluación de una plataforma de actualización remota (OTA) segura y escalable para dispositivos IoT. La motivación principal radica en la elevada cantidad de dispositivos que sufren por vulnerabilidades y la falta de actualizaciones para solucionarlas, lo que crea vectores de ataque a gran escala. Esta plataforma permite la distribución de actualizaciones firmadas y cifradas a flotas de dispositivos para garantizar la autenticidad, integridad y confidencialidad del software desplegado.

El sistema está pensado para todo tipo de escenarios: desde dispositivos sencillos con pocos recursos hasta situaciones críticas que necesitan protegerse de futuros ordenadores cuánticos. Por eso, soporta dos tipos de criptografía: algoritmos post-cuánticos y algoritmos ligeros (lightweight cryptography), ideales para equipos con poca memoria y batería.

El trabajo incluye un análisis del estado de la práctica sobre actualizaciones OTA en IoT y de las necesidades de seguridad actuales, mostrando la importancia de incorporar criptografía ligera para dispositivos con recursos limitados y la opción de soluciones resistentes a la computación cuántica cuando el contexto lo requiera. Además, se desarrolla un prototipo de plataforma, desplegable en la nube y diseñado para escalar horizontalmente.

Palabras clave: OTA, IoT, actualizaciones seguras, post-cuántico, criptografía ligera, firma digital, cifrado.

Impacto en los Objetivos de Desarrollo Sostenible (ODS): xxx,

Abstract

This Master's thesis proposes the design, implementation, and evaluation of a secure and scalable over-the-air (OTA) update platform for IoT devices. The main motivation is the large number of devices that suffer from vulnerabilities and are not updated, creating large-scale attack vectors. The platform enables distributing signed and encrypted updates to device fleets to ensure the authenticity, integrity, and confidentiality of deployed software.

To accommodate both resource-constrained devices and scenarios requiring resilience against future quantum attacks, the system supports two families of cryptographic algorithms: post-quantum resistant encryption and signing algorithms, and lightweight cryptography algorithms optimized for devices with limited memory, CPU, and energy.

The work includes an analysis of current OTA practices and security needs, highlighting the importance of incorporating lightweight cryptography for constrained devices and offering post-quantum options when the context requires it. Additionally, a cloud-deployable prototype platform is developed, designed for horizontal scalability.

Keywords: OTA; IoT; secure updates; post-quantum; lightweight cryptography; digital signatures; encryption.

Laburpena

Master Amaierako Lan honek IoT gailuetarako OTA (over-the-air) eguneratze-plataforma segurua eta eskalagarri baten diseinua, inplementazioa eta ebaluazioa proposatzen du. Motibazio nagusia da ahultasunak dituzten eta eguneratzerik jasotzen ez duten gailu ugariena, eta horrek erasotze-bideak sortzen ditu eskala handian. Plataforma honek eguneratze sinatuak eta zifratutakoak banatzeko aukera ematen du, sistema batean instalatutako softwarearen jatorria, osotasuna eta konfidentzialtasuna bermatzeko.

Baliabide mugatuak dituzten gailuetarako eta etorkizuneko eraso kuantikoen aurrean erresistentzia behar duten egoeretarako egokituz, sistemak bi kriptografia-familiari laguntza eskaintzen die: kuantumaren aurkako erresistentzia duten zifratze eta sinadura algoritmoak, eta memoria, CPU eta energia murrizketak dituzten gailuentzako optimizatutako 'lightweight' kriptografia algoritmoak.

Lanak OTA eguneratzeei buruzko egungo praktiken analisia eta segurtasun-beharren azterketa barne hartzen ditu, baliabide mugatutako gailuentzako kriptografia arina sartzeko garrantzia nabarmentzen du eta beharrezkoa denean kuantumaren aurkako soluzioak aukeratzearen garrantziaz mintzo da. Gainera, hodeian martxan jar daitekeen eta horizontalki eskalatzeko diseinatutako plataforma prototipo bat garatu.

Gako-hitzak: OTA; IoT; eguneratze seguruak; post-kuantikoa; kriptografia arina; sinadura digitala; zifratzea.

2 Lista de acrónimos

AEAD Authenticated Encryption with Associated Data (Cifrado Autenticado con Datos Asociados)

AES Advanced Encryption Standard (Estándar de Cifrado Avanzado)

API Application Programming Interface (Interfaz de Programación de Aplicaciones)

ASCON Algoritmo de criptografía ligera estandarizado por NIST

CPU Central Processing Unit (Unidad Central de Procesamiento)

CRA Cyber Resilience Act (Ley de Resiliencia Cibernética de la UE)

DDoS Distributed Denial of Service (Denegación de Servicio Distribuida)

DFU Device Firmware Update (Actualización de Firmware del Dispositivo)

ECDSA Elliptic Curve Digital Signature Algorithm (Algoritmo de Firma Digital de Curva Elíptica)

ICS Industrial Control Systems (Sistemas de Control Industrial)

IoT Internet of Things (Internet de las Cosas)

LWC Lightweight Cryptography (Criptografía Ligera)

MLDSA Module-Lattice-Based Digital Signature Algorithm (Algoritmo de Firma Digital basado en Redes de Módulos, antes Dilithium)

NIST National Institute of Standards and Technology (Instituto Nacional de Estándares y Tecnología de EE.UU.)

ODS Objetivos de Desarrollo Sostenible

OTA Over-The-Air (Actualización remota por aire)

PC Personal Computer (Ordenador Personal)

PQC Post-Quantum Cryptography (Criptografía Post-Cuántica)

RAM Random Access Memory (Memoria de Acceso Aleatorio)

RSA Rivest–Shamir–Adleman (Algoritmo criptográfico de clave pública)

SDK Software Development Kit (Kit de Desarrollo de Software)

SPHINCS+ Esquema de firma digital post-cuántico basado en funciones hash

TFM Trabajo de Fin de Máster

TUTK ThroughTek Kalay (Plataforma IoT)

UE Unión Europea

3 Introducción

El Internet de las Cosas (IoT) ha transformado radicalmente la forma en que los dispositivos interactúan con el entorno físico y digital. Sin embargo, este crecimiento exponencial ha expuesto una brecha crítica: la falta de mecanismos robustos para mantener la seguridad de estos dispositivos a lo largo de su ciclo de vida operativo. Esta introducción presenta el contexto del problema, examina el estado actual de la investigación y la práctica industrial, y define los objetivos de este Trabajo de Fin de Máster.

3.1 Problemática

El ecosistema IoT ha experimentado un crecimiento sin precedentes, con miles de millones de dispositivos conectados desplegados en entornos industriales, domésticos, sanitarios y de infraestructuras críticas. A pesar de este despliegue masivo, la seguridad no ha evolucionado al mismo ritmo. Una cantidad alarmante de dispositivos permanece sin actualizar debido a la ausencia de mecanismos efectivos de actualización remota, convirtiéndose en vectores de ataque a gran escala.

Las actualizaciones OTA (Over-The-Air) inseguras, o directamente inexistentes, son uno de los mayores riesgos en el IoT. Datos de la industria [1] dicen que cerca del 20 % de las organizaciones han sufrido ataques que venían de dispositivos IoT comprometidos, y que la mayoría de las empresas tienen un riesgo importante por culpa de aparatos sin parchear.

Este problema se complica por dos retos técnicos básicos:

1. **Recursos limitados:** Los dispositivos IoT son muy variados, desde chips muy simples de 8 bits con poquísima memoria hasta sistemas más potentes. Esta variedad hace que no se puedan usar las mismas soluciones de seguridad que usamos en ordenadores normales.

2. **La amenaza cuántica:** Los algoritmos usados actualmente para firmar actualizaciones podrían ser vulnerables a los ordenadores cuánticos del futuro. Como los dispositivos IoT suelen durar muchos años, es buena idea pensar en esto desde el principio para los casos donde la seguridad a largo plazo sea clave.

Si a esto se le suma la cantidad de dispositivos que hay, las actualizaciones poco seguras y las amenazas futuras, tenemos un escenario de riesgo importante. Por eso hace falta buscar soluciones nuevas, aprovechando lo que dice la investigación y la experiencia de la industria.

3.2 Motivación

Lo interesante, y lo que motiva este trabajo, es que ambas perspectivas cuentan historias diferentes pero complementarias. La academia ha demostrado que es posible construir sistemas de actualización seguros, eficientes y adaptados a dispositivos con recursos limitados. La industria, sin embargo, muestra que estas soluciones no se han adoptado de forma generalizada: los frameworks más usados siguen dependiendo de criptografía tradicional, ninguno integra algoritmos ligeros como ASCON ni esquemas post-cuánticos como MLDSA.

Para abordar esta problemática de forma rigurosa, este trabajo adopta una doble perspectiva. Por un lado, el **estado del arte académico**: qué dice la investigación sobre las vulnerabilidades reales, qué arquitecturas de actualización segura se han propuesto, y qué avances hay en criptografía ligera y post-cuántica. Por otro lado, el **estado de la práctica**: qué soluciones existen realmente en el mercado (SWUpdate, Mender, hawkBit...), qué limitaciones tienen, y por qué las empresas siguen desplegando dispositivos vulnerables a pesar de que técnicamente existen alternativas mejores.

Ese hueco entre lo que se puede hacer técnicamente y lo que se hace en realidad es donde se centra este TFM. El objetivo es demostrar que se pueden juntar herramientas probadas como SWUpdate con criptografía moderna (ligera y post-cuántica) en una plataforma centralizada que permita gestionar grandes flotas de forma segura.

3.3 Antecedentes

Este trabajo surge de una necesidad identificada durante las prácticas laborales realizadas en Ikerlan, donde se trabajó agregando funcionalidades a Lamassu, una Infraestructura de Clave Pública (PKI) especializada en dispositivos IoT. Lamassu proporciona gestión centralizada de dispositivos IoT y sus certificados digitales, cubriendo el ciclo de vida completo de las identidades criptográficas: desde la provisión inicial de certificados hasta su renovación y revocación de forma segura a escala. Sin embargo, durante este trabajo se identificó una brecha significativa: mientras que Lamassu gestiona exitosamente la identidad criptográfica de los dispositivos, la plataforma carecía de mecanismos integrados para distribuir y actualizar el firmware de forma segura y auditada. Esta carencia motivó la necesidad de desarrollar una solución de actualizaciones OTA que aprovechara la infraestructura de PKI existente para garantizar que cada dispositivo recibiera únicamente actualizaciones auténticas y autorizadas.

Antes de profundizar en el estado del arte, conviene entender cómo hemos llegado hasta aquí. El concepto de actualización remota de software no es nuevo: los sistemas operativos de escritorio llevan décadas recibiendo parches a través de internet, y los smartphones popularizaron las actualizaciones OTA hace más de quince años. Sin embargo, trasladar este modelo al mundo IoT ha resultado ser mucho más complicado de lo que parecía inicialmente.

Este trabajo se centra explícitamente en la distribución de actualizaciones remotas para dispositivos IoT que ejecutan Linux embebido, el entorno mayoritario en dispositivos industriales y comerciales. **?**Linux embebido**?** hace referencia a distribuciones y stacks (Buildroot, Yocto, u-boot, etc.) y a imágenes rootfs y toolchains optimizados para entornos con recursos limitados; no debe confundirse con una instalación de Linux de escritorio o servidor con servicios y recursos plenos. Por tanto, las adaptaciones que este entorno exige son tanto de naturaleza operativa como de límite de recursos.

El problema de fondo es que los dispositivos IoT no son ordenadores ni teléfonos. Un sensor industrial, una bombilla inteligente o un monitor de salud tienen restricciones de hardware muy diferentes: procesadores de bajo consumo, memoria limitada y fuentes de energía restringidas. Aplicar los mismos mecanismos de actualización que usamos en un PC simplemente no funciona, y eso ha llevado a prácticas inseguras: fabricantes que envían actualizaciones sin cifrar, dispositivos que aceptan firmware sin verificar su origen, o productos que salen al mercado sin capacidad de actualización. Estos problemas han facilitado la creación de botnets y ataques DDoS a gran escala en el pasado, por ejemplo, Mirai [2]. Estudios académicos han identificado además la presencia de SDKs y flujos de actualización inseguros que amplían la superficie de ataque a millones de dispositivos [3].

Por otra parte, los dispositivos IoT tienen ciclos de vida muy largos: un router doméstico puede funcionar durante más de diez años y algunos sistemas industriales durante décadas, lo que complica la gestión de vulnerabilidades y actualizaciones en el tiempo [4]. Durante ese plazo, aparecen nuevas vulnerabilidades, se descubren fallos en algoritmos criptográficos y la amenaza de la computación cuántica pone en evidencia la necesidad de pensar en opciones resistentes a largo plazo. Adicionalmente, se identificó la necesidad de un método centralizado de gestión de actualizaciones que permitiera controlar despliegues, auditar cambios y gestionar rollbacks en flotas de dispositivos, un requisito que se vuelve aún más importante con la entrada en vigor de la Cyber Resilience Act (CRA) de la Unión Europea, que impone obligaciones a fabricantes de hardware y software para garantizar que los equipos desplegados sean mantenidos, actualizados y auditables.

3.4 Estado del arte

Esta sección analiza sistemáticamente la literatura académica y las soluciones tecnológicas existentes para identificar las contribuciones científicas, las limitaciones de las implementaciones actuales, y las brechas que justifican la propuesta de este trabajo. El análisis se estructura en cuatro partes: evidencia empírica sobre vulnerabilidades, plataformas OTA existentes, familias de algoritmos criptográficos emergentes, y análisis crítico de las limitaciones identificadas.

3.4.1 Panorama de vulnerabilidades en dispositivos IoT

Es evidente que la seguridad en IoT es un problema grave, tanto si miramos informes de la industria como estudios académicos. Todos coinciden en que hacen falta soluciones sólidas para actualizar el firmware.

Evidencia desde la industria:

Los informes de la industria pintan un panorama preocupante. Se estima que más de 5.600 millones de dispositivos IoT serán vulnerables en los próximos años, sobre todo con la llegada del 5G y el crecimiento masivo de dispositivos conectados [5].

Reportes de seguridad [1] evidencian que aproximadamente el 20 % de las organizaciones han detectado ataques basados en dispositivos IoT en sus infraestructuras, y que la gran mayoría de entornos corporativos presentan exposición significativa a riesgos derivados de dispositivos comprometidos o sin actualizar. Esta situación refleja una superficie de ataque en constante expansión, donde la falta de procedimientos de mantenimiento y actualización constituye una debilidad estructural.

Los informes de seguridad destacan consistentemente que los *mecanismos de actualización inseguros* y el *firmware/software obsoleto* se encuentran entre las 10 principales vulnerabilidades de IoT [6]. La falta de aplicación de parches y la antigüedad del código base de muchos dispositivos representa un punto de entrada preferente para los atacantes, siendo a menudo el eslabón más débil de la cadena de seguridad.

Un ejemplo particularmente preocupante de vulnerabilidades en componentes de terceros se reveló en mayo de 2024, cuando se identificaron fallos críticos en la plataforma IoT ThroughTek Kalay (TUTK), afectando a más de cien millones de dispositivos a nivel global, incluyendo cámaras de vigilancia y sistemas de seguridad. La explotación en cadena de estas vulnerabilidades permite comprometer completamente el dispositivo, subrayando cómo un fallo en un componente de terceros puede tener un impacto masivo en todo el ecosistema de productos [7].

Adicionalmente, el firmware desactualizado continúa siendo el principal motor de las botnets IoT, como Mirai, que explotan credenciales por defecto o vulnerabilidades conocidas para reclutar dispositivos y lanzar ataques de denegación de servicio distribuido (DDoS) masivos. Incidentes recientes han documentado ataques DDoS sin precedentes impulsados por routers y dispositivos IoT comprometidos [2], demostrando que el problema no solo persiste, sino que se amplifica con el crecimiento del ecosistema.

Evidencia desde la literatura académica:

Desde una perspectiva más técnica, el estudio **?**AoT: Attack on Things**?** de Ibrahim et al. [3] ofrece datos empíricos alarmantes sobre la prevalencia de SDKs vulnerables en el ecosistema de actualizaciones OTA. Los autores analizaron 23 dispositivos IoT comerciales y sus aplicaciones móviles asociadas, identificando seis SDKs de actualización de firmware (DFU, Device Firmware Update) que presentan vulnerabilidades críticas. Mediante un análisis automatizado a gran escala, el estudio reveló que 1,356 aplicaciones disponibles en Google Play Store dependen de estos SDKs vulnerables y que, en conjunto, estas aplicaciones gestionan al menos 61 modelos de dispositivos IoT ampliamente distribuidos. Esta cadena de dependencias inseguras subraya la magnitud del problema: millones de dispositivos en el campo son potencialmente explotables a través de vectores de actualización comprometidos.

Investigaciones recientes sobre IoT industrial [**IIoTSecurity:2023**] confirman estos problemas: firmware que raramente se actualiza tras el despliegue, mecanismos de actualización inseguros, y dispositivos con ciclos de vida de décadas sin mantenimiento adecuado. Todo esto subraya la urgencia de plataformas automatizadas de actualización segura que puedan operar de forma continua a lo largo de toda la vida útil del dispositivo.

La evidencia convergente desde múltiples fuentes establece que las vulnerabilidades en mecanismos de actualización no son casos aislados, sino un problema sistémico que afecta a millones de dispositivos desplegados. Esta constatación motiva el análisis de las soluciones tecnológicas existentes para determinar si abordan adecuadamente estos desafíos.

3.4.2 Plataformas OTA existentes: análisis comparativo

En respuesta a las necesidades documentadas de actualización segura, la comunidad de código abierto y diversos proveedores comerciales han desarrollado plataformas especializadas. Sin embargo, el análisis comparativo revela limitaciones significativas en relación con los requisitos identificados.

SWUpdate [8] es un framework de código abierto para actualizaciones en sistemas Linux embebidos, con soporte para actualizaciones atómicas (esquemas A/B) y firmas RSA/ECDSA, pero sin gestión centralizada de flotas. En resumen, es un software capaz de instalar las actualizaciones de forma segura en el dispositivo, pero no proporciona una solución completa para la gestión de actualizaciones a gran escala. La gran ventaja de SWUpdate es su adaptabilidad gracias a los modulos y su customización.

Eclipse hawkBit [9] ofrece gestión backend de actualizaciones OTA a escala empresarial con rollout progresivo, aunque la seguridad criptográfica depende de la implementación del cliente.

Mender [10] integra cliente y servidor para actualizaciones OTA con gestión de flotas, pero utiliza exclusivamente algoritmos tradicionales (RSA y ECC) sin soporte para criptografía ligera ni post-cuántica.

Balena [11] proporciona gestión de flotas basada en contenedores Docker, introduciendo overhead significativo que lo hace inadecuado para dispositivos muy restringidos.

extbfRAUC [12] es similar a SWUpdate en funcionalidades de actualización atómica, pero tampoco incluye familias de algoritmos criptográficos avanzadas.

La Tabla 1 resume las características principales de estas plataformas.

Cuadro 1: Comparativa de plataformas OTA existentes

Plataforma	**Firmas**	**Cifrado**	**LWC**	**PQC**	**Gestión flota**
SWUpdate	Sí	Sí	No	No	Externa
hawkBit	Delegada	Delegado	No	No	Sí
Mender	Sí	Sí	No	No	Sí
Balena	Sí	Sí	No	No	Sí
RAUC	Sí	Sí	No	No	Externa

LWC: Lightweight Cryptography; PQC: Post-Quantum Cryptography

Como se observa, ninguna de las soluciones existentes ofrece soporte simultáneo para criptografía ligera y post-cuántica. Esta carencia representa una limitación significativa dado el crecimiento de dispositivos con recursos extremadamente limitados y la amenaza emergente de la computación cuántica. El presente trabajo se construye sobre SWUpdate como agente de actualización en el dispositivo, aprovechando su madurez y robustez probada, pero extendiéndolo gracias a su personalización y facil agregación de modulos con una capa de gestión centralizada que incorpora las familias de algoritmos criptográficos avanzadas ausentes en las soluciones actuales.

Este análisis revela una brecha clara: mientras existen frameworks robustos para gestión de actualizaciones y organismos de estandarización que han definido familias de algoritmos criptográficos avanzadas, ninguna solución integra ambas. La siguiente subsección examina el estado de estas familias de algoritmos criptográficos emergentes.

3.4.3 Necesidad de algoritmos criptográficos eficientes

La enorme variedad de dispositivos IoT, desde chips minúsculos hasta sistemas potentes, hace que necesitemos diferentes tipos de criptografía. Para las actualizaciones OTA, hay dos familias clave: la criptografía ligera para los dispositivos pequeños, y la post-cuántica para protegernos del futuro.

Criptografía ligera: ASCON y el estándar NIST LWC.

El NIST ha elegido ASCON [13] como el estándar para criptografía ligera. Está hecho a medida para dispositivos con pocos recursos y ofrece cifrado autenticado (AEAD), lo que garantiza que los datos son confidenciales y no han sido modificados.

Para las actualizaciones OTA, ASCON es ideal: es hasta cinco veces más eficiente que AES si no tienes hardware dedicado, consume muy poca memoria y protege tanto el firmware como sus metadatos, evitando ataques de repetición o modificación.

Criptografía post-cuántica: preparación ante amenazas futuras.

Los ordenadores cuánticos son una amenaza real a medio plazo para los algoritmos que usamos hoy (RSA, ECDSA). Como los dispositivos IoT duran muchos años, es de sentido común pensar en esto desde ya.

El NIST ha seleccionado tres esquemas de firma post-cuántica: MLDSA (buen equilibrio), Falcon (firmas pequeñas) y SPHINCS+ (muy seguro teóricamente). Las firmas digitales son clave para asegurar que el firmware es auténtico, incluso frente a ataques cuánticos.

Que el NIST haya estandarizado estos algoritmos significa que técnicamente es posible usarlos. Pero integrarlos en plataformas reales va despacio. En este proyecto he elegido MLDSA como esquema principal para las firmas digitales, combinándolo con cifrado ligero.

3.5 Objetivos

A partir del análisis de la problemática y lo comentado previamente, se establecen los siguientes objetivos para este Trabajo de Fin de Máster.

3.5.1 Propósito y alcance

Visto el panorama de vulnerabilidades en el ecosistema IoT, las limitaciones de las plataformas OTA existentes y la brecha entre la investigación y la aplicación, este trabajo propone una solución que combina tecnologías consolidadas con algoritmos criptográficos modernos y necesarios.

La estrategia adoptada consiste en utilizar **SWUpdate** como instalador de actualizaciones en el dispositivo, aprovechando su madurez y robustez probada en entornos de producción industrial, pero extendiendo sus capacidades criptográficas para incluir soporte tanto para **algoritmos de criptografía ligera (LWC)** como para **esquemas post-cuánticos**. Esta extensión se integra con una **plataforma de gestión centralizada de flotas de dispositivos**, diseñada para escalar horizontalmente y proporcionar trazabilidad completa del ciclo de vida de las actualizaciones.

Objetivo general:

Diseñar, implementar y evaluar una plataforma de actualización OTA segura y escalable para dispositivos IoT basada en SWUpdate, extendida con soporte para criptografía ligera (LWC) y post-cuántica, e integrada con una arquitectura de gestión centralizada de flotas, abordando las limitaciones identificadas en soluciones existentes.

Objetivos específicos:

1. **Implementar verificación criptográfica robusta:** Desarrollar mecanismos de firma digital basados en algoritmos tradicionales (RSA/ECDSA) y post-cuánticos (MLDSA) para garantizar la autenticidad e integridad de cada paquete de actualización, mitigando el riesgo de distribución de firmware malicioso documentado en estudios como **?**AoT: Attack on Things**?** [3].

2. **Integrar nuevas opciones criptográficas:** Dar soporte a algoritmos ligeros (ASCON) y post-cuánticos (MLDSA), para cubrir tanto dispositivos muy limitados como aquellos que necesitan protegerse del futuro.

3. **Arquitectura escalable en la nube:** Crear una plataforma de gestión que pueda crecer horizontalmente para administrar muchos dispositivos y tener control total sobre las actualizaciones (saber si han fallado, reintentarlas, etc.).

4. **Integrar frameworks maduros de actualización:** Construir la solución sobre SWUpdate como agente de actualización en el dispositivo, aprovechando su robustez probada en entornos de producción industrial mientras se añade la capa de gestión centralizada y las familias de algoritmos criptográficos avanzadas ausentes en su implementación estándar.

3.6 Planificación

Para la consecución de los objetivos de este Trabajo de Fin de Máster, se ha definido un plan de trabajo dividido en tres fases principales, que abarcan desde la investigación inicial hasta el despliegue y validación de la plataforma.

3.6.1 Fase 1: Investigación

El objetivo de esta fase es establecer las bases teóricas y técnicas del proyecto, analizando el estado del arte y seleccionando las tecnologías más adecuadas. En esencia, se busca analizar los estándares actuales, identificar qué algoritmos son usados comúnmente y evaluar las nuevas alternativas disponibles, tal como se ha comentado en la introducción.

1.1 Revisión de estándares actuales. Se realizará un estudio exhaustivo de las normativas y recomendaciones de seguridad vigentes para dispositivos IoT (como las guías de NIST y ETSI), así como de los mecanismos de actualización OTA utilizados actualmente en la industria.

1.2 Evaluación de algoritmos lightweight y post-cuánticos. Se analizarán y compararán diferentes algoritmos criptográficos, poniendo el foco en aquellos de la familia de criptografía ligera (LWC) para dispositivos con recursos limitados y en los esquemas de criptografía post-cuántica (PQC) para garantizar la seguridad a largo plazo.

1.3 Elección de algoritmos para los escenarios planteados. Basándose en la evaluación anterior, se seleccionarán los algoritmos de firma y cifrado más adecuados para los distintos escenarios de uso definidos, equilibrando seguridad y rendimiento.

3.6.2 Fase 2: Diseño e integración de módulos de seguridad para dispositivos IoT

Esta fase constituye el núcleo del desarrollo técnico, donde se diseñan e implementan los componentes de la plataforma. Se incluye el desarrollo del frontend, que es la web que permitirá a los desarrolladores hacer uso de toda la infraestructura.

2.1 Diseño de la arquitectura requerida. Definición de la arquitectura global del sistema, especificando los componentes del backend, la base de datos, la API de comunicación y la estructura del cliente en el dispositivo IoT.

2.2 Integración de algoritmos de cifrado y/o firma. Implementación e integración de las librerías criptográficas seleccionadas en la Fase 1 dentro de los servicios de la plataforma, habilitando las capacidades de firma digital y cifrado de los paquetes de actualización tanto en el backend como en el dispositivo IoT.

2.3 Desarrollo del módulo de distribución segura de actualizaciones. Creación del servicio encargado de gestionar el repositorio de actualizaciones, verificar la autenticidad de las peticiones y distribuir los binarios de forma segura a los dispositivos autorizados.

2.4 Integración de firma y cifrado Backend y Frontend. Desarrollo de la interfaz web (frontend) y su conexión con el backend, permitiendo a los administradores y desarrolladores subir nuevos firmwares, firmarlos criptográficamente y gestionar las campañas de actualización de manera intuitiva.

2.5 Contenerización y despliegue en k8s. Empaquetado de todos los microservicios de la plataforma en contenedores (Docker) y orquestación de su despliegue en un clúster de Kubernetes (k8s) para garantizar la escalabilidad, disponibilidad y facilidad de gestión.

3.6.3 Fase 3: Automatización de pruebas, despliegue y ciclo de vida

La fase final se centra en la calidad, la automatización y la validación del sistema en entornos realistas.

3.1 Creación o adaptación del pipeline CICD. Implementación de flujos de Integración y Despliegue Continuo (CI/CD) para automatizar la construcción de las imágenes, la ejecución de pruebas y el despliegue de nuevas versiones de la plataforma.

3.2 Definición de tests unitarios y e2e. Desarrollo de una batería de pruebas automatizadas que incluya tests unitarios para validar la lógica de los componentes individuales y tests de extremo a extremo (e2e) para verificar el flujo completo de actualización OTA.

3.3 Monitorización y validación para entornos de desarrollo y producción. Puesta en marcha de herramientas de observabilidad y monitorización para supervisar el rendimiento y la salud del sistema, validando su correcto funcionamiento tanto en entornos de prueba como en producción.

La planificación temporal de estas fases y tareas se detalla en el diagrama de Gantt de la Figura 1.

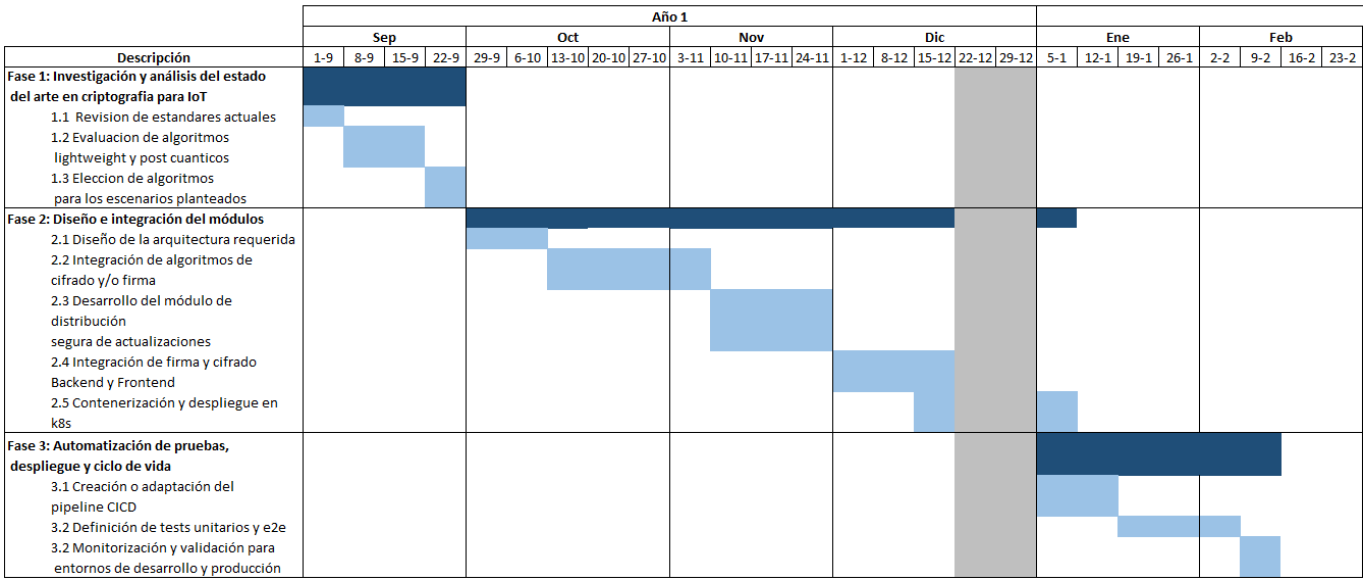

Figura 1: Planificación temporal del proyecto (Diagrama de Gantt)

Referencias

[1] Gartner. *IoT security surveys and reports*. Inf. téc. Consultado en 2023. Gartner, 2023.

[2] Kaspersky Security. *Mirai Botnet and IoT-Based DDoS Attacks: Evolution and Current Threats*. https://www.kaspersky.com/resource-center/threats/mirai-botnet. Análisis de botnets IoT y ataques DDoS masivos impulsados por dispositivos comprometidos. 2025.

[3] Muhammad Ibrahim, Andrea Continella y Antonio Bianchi. **?**AoT - Attack on Things: A security analysis of IoT firmware updates**?** En: *2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)*. 2023, págs. 1047-1064. DOI: 10.1109/EuroSP57164.2023.00065.

[4] Yewande Goodness Hassan et al. **?**Security Challenges in Industrial IoT: Update Mechanisms and Firmware Vulnerabilities**?** En: *International Journal/Conference Name* 04.01 (ene. de 2023). Received: 12-12-2022; Accepted: 19-01-2023, págs. 697-703. ISSN: 2582-7138.

[5] Cisco Systems. *Cisco Annual Internet Report (2018–2023) and IoT Forecast*. Inf. téc. Proyección de más de 29 mil millones de dispositivos IoT conectados para 2025. Cisco Systems, 2025. URL: https://www.cisco.com/c/en/us/solutions/collateral/executive-perspectives/annual-internet-report/white-paper-c11-741490.html.

[6] Fortinet. *Top 10 IoT Vulnerabilities and Security Threats*. https://www.fortinet.com/resources/cyberglossary/iot-security. Informe sobre las principales vulnerabilidades en dispositivos IoT, incluyendo mecanismos de actualización inseguros y firmware obsoleto. 2024.

[7] Bitdefender Labs. *Critical Vulnerabilities in ThroughTek Kalay Platform Affect Over 100 Million IoT Devices*. Inf. téc. Vulnerabilidades críticas en la plataforma TUTK que afectan a más de 100 millones de dispositivos IoT globalmente. Bitdefender, mayo de 2024. URL: https://www.bitdefender.com/blog/labs/notes-on-throughtek-kalay-vulnerabilities-and-their-impact/.

[8] SWUpdate Project. *SWUpdate - Software Update for Embedded Linux Devices*. Framework de código abierto para actualizaciones de firmware en sistemas Linux embebidos con soporte para actualizaciones atómicas A/B y firmas RSA/ECDSA. 2024. URL: https://swupdate.org/.

[9] Eclipse Foundation. *Eclipse hawkBit - IoT Update Management*. Plataforma backend de código abierto para gestión de actualizaciones OTA a escala empresarial con rollout progresivo y gestión de campañas. 2024. URL: https://www.eclipse.org/hawkbit/.

[10] Northern.tech. *Mender - Over-the-Air Software Updates for IoT*. Plataforma OTA que integra cliente y servidor para actualizaciones con gestión de flotas, rollback automático e interfaz gráfica. 2024. URL: https://mender.io/.

[11] Balena Inc. *Balena - IoT Fleet Management Platform*. Plataforma de gestión de flotas IoT basada en contenedores Docker para despliegue y actualización de aplicaciones. 2024. URL: https://www.balena.io/.

[12] RAUC Project. *RAUC - Robust Auto-Update Controller*. Framework de código abierto para actualizaciones seguras de sistemas Linux embebidos con soporte para múltiples esquemas de particionado. 2024. URL: https://rauc.io/.

[13] National Institute of Standards and Technology (NIST). *Lightweight Cryptography: Program Document*. Special Publication (SP) 800-232. U.S. Department of Commerce, 2023. DOI: 10.6028/NIST.SP.800-232.ipd. URL: https://csrc.nist.gov/pubs/sp/800/232/ipd.