UTE FORENSIA THOT

F1.1.5. Plan de Mantenimiento y Sostenibilidad

THOT

Periodo de Informe 30/09/2025 a 28/02/2026

Fecha: 27/02/2026

Versión: 2. 0

Información de control del documento

Descripción	Valor
Título del Documento:	F1.1.5. Plan de Mantenimiento y Sostenibilidad
Nombre del Proyecto:	THOT
Autor del documento:	Sergio Zaera Mata, Sergio Queraltó Pereira, Jaime Castro Cernadas
Propietario del Proyecto:	CDTI
Director del Proyecto:	Roberto Gómez-Espinosa
Versión Doc.:	2.0
Confidencialidad:	Alta
Fecha:	27/02/2026

Aprobación y revisión del documento:

NOTA: Se requieren todas las aprobaciones. Se deben mantener registros de cada aprobación. Todos los revisores de la lista se consideran necesarios a menos que se indique explícitamente como Opcionales.

Nombre	Rol	Acción	Fecha
Sergio Zaera Mata	Jefe de Proyecto	Revisa	26/01/2026

Historial de documentos:

El Autor del Documento está autorizado a hacer los siguientes tipos de cambios al documento sin requerir que el documento sea aprobado nuevamente:

- *Editorial, formateo y ortografía.*
- *Aclaración.*

Para solicitar un cambio en este documento, póngase en contacto con el Autor o el Propietario del Documento.

Las modificaciones de este documento se resumen en la siguiente tabla en orden cronológico inverso (primero la última versión).

Revisión	Fecha	Creada por	Breve descripción de los cambios
0.0	07/10/25	Sergio Zaera Mata	Preparación ToC
0.1	03/11/25	Sergio Queraltó, Jaime Castro	Contribuciones técnicas iniciales
0.2	28/11/25	UTE ForensIA (Todos)	Revisión & Contribuciones adicionales
1.0	05/12/25	Sergio Zaera Mata	1º Borrador
1.1	16/01/26	UTE ForensIA (Todos)	Contribuciones técnicas
1.2	21/01/26	Sergio Queraltó, Jaime Castro	Revisión & Consolidación
2.0	26/01/26	Sergio Zaera Mata	2º Borrador

Cofinanciado por la Unión Europea
Fondos Europeos

ADVERTENCIA DE CONFIDENCIALIDAD Y RESPONSABILIDAD LEGAL

Este documento contiene información confidencial y secretos empresariales propiedad de la UTE FORENSIA THOT, protegidos por la Ley 1/2019 de Secretos Empresariales, el artículo 13 de la Ley de Contratos del Sector Público (LCSP) y la Directiva (UE) 2016/943 sobre protección de know-how.

Se entrega exclusivamente para la finalidad prevista en el procedimiento administrativo o contractual.

Queda terminantemente prohibida su reproducción, divulgación, cesión o uso por terceros sin autorización expresa y por escrito.

El incumplimiento de estas obligaciones puede constituir:

- Infracción contractual, con las consecuencias previstas en la LCSP.
- Responsabilidad civil y penal, conforme a la Ley 1/2019 y al Código Penal (arts. 278 y ss.).
- Acciones judiciales inmediatas, incluyendo reclamación de daños y perjuicios y medidas cautelares.

Si usted no es el destinatario autorizado, debe comunicarlo de inmediato y proceder a la eliminación del documento. Cualquier uso indebido será perseguido con el máximo rigor legal".

Cofinanciado por la Unión Europea
Fondos Europeos

TABLA DE CONTENIDOS

ÍNDICE DE FIGURAS

ÍNDICE DE TABLAS

1. INTRODUCCIÓN

1.1. Resumen ejecutivo

El presente documento define las prestaciones funcionales y técnicas de la plataforma THOT con el objetivo de establecer un marco técnico sólido que guíe el diseño, la implantación y la evolución de una solución integral de inteligencia forense policial. THOT se concibe como una plataforma interoperable, segura y avanzada, orientada a transformar los procesos de Policía Científica mediante la integración de datos, la automatización de flujos de trabajo, el uso responsable de inteligencia artificial y la explotación de información científica y operativa en tiempo real.

En **Fase I,** el proyecto se centra en asentar los fundamentos funcionales y tecnológicos que permitan avanzar hacia capacidades innovadoras y verificables, asegurando un salto significativo respecto al estado del arte. En este marco, la definición rigurosa de prestaciones es clave para garantizar que la plataforma responde a necesidades reales de los usuarios finales, cumple las exigencias normativas y habilita un desarrollo escalable e interoperable alineado con los niveles de madurez tecnológica requeridos.

En este contexto, el documento **F1.1.5 Plan de Mantenimiento y Sostenibilidad (PMS)** establece el marco operativo inicial para garantizar la continuidad, seguridad, evolución controlada y sostenibilidad técnica y económica de THOT (Lote 1). El plan define cómo se previenen degradaciones, cómo se responde a incidencias y cómo se introducen mejoras sin comprometer la disponibilidad del servicio ni la trazabilidad exigida en un entorno forense.

La estrategia de **mantenimiento** de THOT se basa en un enfoque evidence-driven, de forma que cualquier actuación **preventiva, correctiva o evolutiva** queda sustentada por evidencias verificables obtenidas de la **observabilidad** (métricas, logs y trazas), el sistema de ticketing, el control de versiones y procedimientos operativos (runbooks y checklists), aplicándose tanto a los microservicios de la plataforma como a su infraestructura de ejecución. Por último, incorpora un enfoque de **sostenibilidad técnica y económica** que permite estimar y auditar el OPEX a partir de datos de operación y explotación.

Además, el plan incorpora la interoperabilidad con el Lote 2 como capacidad crítica, gobernando y validando contratos de integración, autenticación segura y consistencia extremo a extremo, sin asumir la operación interna del Lote 2; ante degradaciones externas, las actuaciones se restringen a lo observable desde la plataforma central, manteniendo coordinación y trazabilidad mediante evidencias asociadas.

1.2. Propósito del documento

El propósito de este documento es definir, con rigor operativo y trazabilidad, el plan inicial de mantenimiento y sostenibilidad de THOT (Lote 1), de forma que:

- Se disponga de un marco operativo ejecutable para mantenimiento preventivo, correctivo y evolutivo, incluyendo cadencias, procedimientos verificables, evidencias mínimas y criterios de aceptación/rollback, alineados con la operación continua de una plataforma central.
- Queden definidos los roles, responsabilidades y circuitos de escalado (L1/L2/L3 y roles transversales como Seguridad/PKI y Cadena de Custodia), asegurando que cada intervención tiene propietario, criterio de decisión y trazabilidad en ticketing/CMDB.
- Se establezcan estrategias de actualización y control de configuración (GitOps/IaC, CI/CD, registro de artefactos, políticas de admisión, despliegues progresivos), orientadas a minimizar riesgo, evitar deriva, y permitir reversión rápida ante degradación.

- Se habilite una base auditable para la sostenibilidad técnica y económica, apoyada en medición real (incidencias, MTTR, consumo, crecimiento de datos, coste operativo de herramientas y soporte) y en el control de obsolescencia (EOL/EOS) y dependencia tecnológica.

1.3. Alcance del documento

El alcance del Plan de Mantenimiento y Sostenibilidad de THOT (Lote 1) cubre la operación de la plataforma central en entorno CPD/nube privada, incluyendo infraestructura, orquestación, servicios base y microservicios de negocio, así como los puntos de integración relevantes. En particular, el plan aplica a:

- **Plataforma de ejecución e infraestructura central**: el plan cubre la operación del clúster Kubernetes (plano de control y nodos de trabajo), la conectividad y segmentación de red (incluyendo el componente de red del clúster cuando aplique), los elementos de entrada (Ingress y/o API Gateway), los mecanismos de persistencia (volúmenes y almacenamiento de objetos), así como el control de capacidad, la alta disponibilidad y la recuperación ante fallos de los componentes críticos.
- **Servicios centrales de THOT**: el alcance incluye la persistencia y el gobierno del dato, la mensajería/eventos y los pipelines de procesamiento, los servicios de IA/ML y su despliegue/serving cuando aplique, la coordinación/orquestación, los servicios de alertas y la interfaz de usuario central, junto con las dependencias transversales necesarias para su operación.
- **Seguridad y trazabilidad operativa**: gestión de identidades y permisos, auditoría de operaciones, gestión de secretos y certificados, control de cambios, evidencias vinculadas a ticketing y CMDB, y mecanismos de verificación asociados.
- **Continuidad del servicio**: políticas y ejecución de backup, restauración verificada, DR, y pruebas periódicas con evidencias (runbooks y actas), con preservación de los registros necesarios para auditoría y, cuando aplique, de los elementos asociados a Cadena de Custodia.
- **Interoperabilidad con Lote 2 y sistemas externos**: gobierno de contratos, validación de mensajes/eventos, autenticación mutua y cifrado, compatibilidad entre versiones, observabilidad por integración y verificación de extremo a extremo cuando se introduzcan cambios que puedan impactar a consumidores.

Este alcance está orientado a operar un sistema central en régimen continuo, con foco en estabilidad, seguridad, auditabilidad y evolución controlada, y sirve como referencia formal para la ejecución y verificación del mantenimiento durante el periodo del informe.

1.4. Estructura del documento

El Plan de Mantenimiento y Sostenibilidad de THOT (Lote 1) se organiza en doce secciones que siguen la siguiente lógica:

- **Sección 1. Introducción**: presenta el propósito del plan, su alcance y la guía de lectura, situando el PMS como marco operativo para continuidad, seguridad, trazabilidad y evolución controlada de THOT.
- **Sección 2. Trazabilidad de requisitos**: vincula los requisitos aplicables con este documento.
- **Sección 3. Recursos y responsabilidades**: define el modelo organizativo de operación (propietarios por servicio/proceso, niveles L1/L2/L3 y roles transversales) y las herramientas mínimas para operar de forma homogénea y auditable.
- **Sección 4. Plan de mantenimiento**: describe el mantenimiento preventivo, correctivo y evolutivo, incluyendo procedimientos verificables, criterios de aceptación/rollback, y el esquema de monitorización, alertas, ticketing y escalado.

- **Sección 5. Ciclo de vida y mantenimiento de la infraestructura central**: cubre inventario/CMDB, operación de nube privada y clúster Kubernetes, obsolescencia (EOL/EOS), servicios base, continuidad (backup/DR) y seguridad operativa con control de versiones.
- **Sección 6. Estrategias de actualización**: establece el control del estado (GitOps/IaC), cadena de suministro, despliegue progresivo y reversión, Policy-as-Code, coordinación de cambios con interoperabilidad (Lote 2) y métricas de eficacia del proceso de actualización.
- **Sección 7. Mantenimiento de modelos de IA**: define inventario/registro de modelos, ciclo de vida MLOps, disparadores de actualización, validación previa, despliegue/rollback y observabilidad específica de IA con gobierno del cambio y evidencias.
- **Sección 8. Gestión de certificados y PKIs**: describe la arquitectura PKI, la automatización de certificados en Kubernetes y la protección de claves/secretos, incluyendo rotación y respuesta ante incidentes con trazabilidad.
- **Sección 9. SLAs y KPIs**: fija (con valores objetivo preliminares) los SLAs operativos y los KPIs técnicos, de UX/formación, ambientales y específicos de Cadena de Custodia, junto con el método de medición y revisión.
- **Sección 10. Estrategia de escalabilidad**: detalla el enfoque de escalado y escenarios de crecimiento, con riesgos y mitigaciones basadas en evidencia operacional.
- **Sección 11. Sostenibilidad técnica y económica**: define el marco CAPEX/OPEX, drivers medibles, modelo de costes recurrentes, obsolescencia y resultados (plantillas/escenarios) para sostener el servicio a largo plazo.
- **Sección 12. Anexos**: concentra material de detalle (matrices, catálogos, plantillas y cuadros de mando) para evitar cargar el cuerpo principal y facilitar auditoría.

Cofinanciado por la Unión Europea
Fondos Europeos

2. Trazabilidad de requisitos

ID de requisito	Resumen del Requisito	Por qué el PMS lo cubre
CAL-L1-2	Trazabilidad y soporte a auditorías operativas y forenses	El PMS define que toda intervención deje evidencias verificables y correlacionables, incluyendo restauraciones y cambios, para sostener auditoría técnica y validez operativa.
CAL-L1-3	Mejora continua y acciones correctivas y preventivas	El PMS integra alertas, incidentes y análisis de causa raíz con acciones correctivas y preventivas, ajuste de umbrales y runbooks para reducir recurrencia y OPEX.
COMUN-2	Gobierno de contratos y versionado de interfaces	El PMS contempla cambios controlados de APIs y eventos con compatibilidad hacia atrás, validación extremo a extremo y rollback para evitar roturas a Lote 2 y sistemas externos.
COMUN-5	Etiquetado compatible y cadena de custodia con registros sincronizados	El PMS incluye operación forense segura en THOT, preservando integridad y trazabilidad y verificando consistencia de datos recibidos desde Lote 2 sin asumir su operación interna.
COMUN-6	Seguridad y privacidad de los datos	El PMS asegura que operación, cambios y recuperaciones preservan cifrado, controles de acceso, auditoría y continuidad, evitando degradación de la seguridad por mantenimiento.
COMUN-7	Monitorización y registro de errores	El PMS define observabilidad completa con severidades, correlación y escalado, y exige evidencias reproducibles para diagnóstico y cumplimiento de objetivos operativos.
HW-L1-11	Trazabilidad robusta para integridad y reproducibilidad	El PMS establece versionado de despliegues y artefactos, identificación y correlación de eventos, y evidencias de operación para reconstruir estado, cambios y resultados.
HW-L1-14	Gestión integral de riesgos operativos y técnicos	El PMS incorpora evaluación de riesgos vinculada a mantenimiento, continuidad, seguridad y obsolescencia, con controles y revisiones para reducir impacto y downtime.
HW-L1-3	Alta disponibilidad, escalabilidad y recuperación ante fallos	El PMS define cómo se opera THOT en clúster para mantener continuidad, incluyendo prácticas de alta disponibilidad, escalado controlado y procedimientos de recuperación y verificación con evidencias.
HW-L1-4	Modularidad y extensibilidad para evolucionar el sistema	El PMS establece mantenimiento evolutivo y control de versiones de microservicios y plataforma, con despliegues progresivos y rollback para introducir mejoras sin romper la operación.

Cofinanciado por la Unión Europea
Fondos Europeos

HW-L1-6	Resiliencia operativa y tolerancia a fallos en pipelines	El PMS cubre operación, monitorización y respuesta ante fallos de procesamiento mediante observabilidad, contención, reintentos y recuperación trazable para sostener la continuidad del servicio.
HW-L1-7	Monitorización y gestión de fallos de la plataforma	El PMS define observabilidad central, catálogo de alertas, correlación con ticketing y escalado L1/L2/L3 para detectar, diagnosticar y resolver fallos de forma verificable.
HW-L1-8	Operación y mantenimiento de infraestructura y servicios base	El PMS incluye el ciclo de vida operativo de la infraestructura central y la plataforma base que soportan THOT, con cambios controlados, verificación postcambio y trazabilidad en CMDB y auditoría.
HW-L1-9	Ciclo de vida del dato, retención y destrucción verificable	El PMS cubre gobierno del dato y continuidad, definiendo políticas de retención, backups y procedimientos de borrado controlado con registros auditables.
INT-17	Identificación temprana de incidencias mediante alertas	El PMS establece alertas por dominios críticos de THOT y umbrales de severidad para anticipar degradaciones y activar acciones preventivas antes de afectar a usuarios o a la interoperabilidad.
INT-18	Integración de alertas con ticketing y escalado operativo	El PMS conecta alertas con gestión de incidencias y cambios, definiendo triage, escalado y cierres con RCA para reducir MTTR y asegurar trazabilidad y mejora continua.
MARCO-5	SLA de interoperabilidad	El PMS define monitorización y operación de la interoperabilidad desde THOT, con métricas y alertas para latencia, errores y disponibilidad, y escalado cuando se degrada.
OBL-SEG-1	Seguridad durante todo el ciclo de vida conforme a políticas	El PMS gobierna vulnerabilidades, parches y cambios con evaluación de impacto, mitigación y evidencias, manteniendo cumplimiento operativo sostenido en el tiempo.
OBL-SEG-2	Gestión de vulnerabilidades y respuesta operativa ante riesgos	El PMS define revisiones periódicas, priorización por severidad y mecanismos de remediación o mitigación con control de cambios y verificación, evitando gestión reactiva.
OBL-SEG-3	Soporte a auditorías de seguridad por organismos competentes	El PMS estructura evidencias exportables de operación, cambios, accesos y seguridad para facilitar auditorías, reproducibilidad y verificación independiente.
OBL-SEG-4	Auditoría completa de operaciones y seguridad en datos y comunicaciones	El PMS exige trazabilidad end to end con retención gobernada y correlación con CMDB y tickets, reforzando control de accesos, comunicaciones seguras y registro de actividad.

SEC-L1-1	Estrategia integral de seguridad, control de acceso y auditoría	El PMS incorpora hardening, control de accesos y gestión del cambio segura para que la operación y las actualizaciones mantengan la postura de seguridad y la auditabilidad.
SEC-L1-2	Protección e integridad de datos, cifrado y continuidad	El PMS incluye cifrado en tránsito y en reposo, control de secretos y backup y restauración verificada para preservar integridad y disponibilidad de datos operativos y forenses.
SEC-L1-3	Integración con identidad corporativa y control granular	El PMS contempla la operación segura de autenticación y autorización con roles y trazabilidad, incluyendo revisiones y evidencias de accesos y acciones administrativas.
SEC-L1-4	Comunicaciones seguras entre módulos e integraciones	El PMS cubre PKI y TLS mTLS para tráfico interno y externo, incluyendo rotación y revocación alineadas con cambios para evitar interrupciones y reducir riesgo.

***Tabla 1 Matriz de trazabilidad de requisitos del pliego aplicables al PMS del Lote 1 (THOT**. Resumen del requisito y justificación de su cobertura en el plan de mantenimiento y sostenibilidad técnica.*

3. Recursos y responsabilidades

Este apartado define el modelo de responsabilidad operativa para el mantenimiento de la plataforma central THOT (Lote 1), asignando propietarios de servicio y propietarios de proceso a lo largo del ciclo de vida en operación. En particular, se cubren las funciones de despliegue y operación de servicios, gestión de cambios, monitorización centralizada, gestión de incidencias y actualización o retirada controlada de componentes.

El objetivo es asegurar: (i) la continuidad operativa de la plataforma central; (ii) la coherencia técnica con la arquitectura de microservicios de THOT (espacio de datos, procesamiento, IA/ML, coordinación y gestión, alertas, comunicaciones y capa de interfaz de usuario); y (iii) el control de seguridad, versionado y trazabilidad, incluyendo la preservación de la Cadena de Custodia y la interoperabilidad con el Lote 2 y con sistemas policiales externos.

Para garantizar una atención consistente y escalable, el mantenimiento se organiza en tres niveles de soporte. El nivel L1 (Operación) se responsabiliza del triaje inicial, la ejecución de procedimientos operativos (runbooks), la verificación de síntomas, la aplicación de medidas de contención y la recuperación básica del servicio. El nivel L2 (Especialistas por módulo) asume el diagnóstico avanzado, el análisis de logs, métricas y trazas, la aplicación de ajustes de configuración y la coordinación técnica entre módulos cuando el impacto trasciende un único componente. El nivel L3 (Ingeniería) aborda correcciones estructurales, cambios evolutivos, hotfixes y rollbacks complejos, así como el gobierno del ciclo de vida de modelos de IA cuando aplique (versionado, actualización controlada y validación técnica).

De forma complementaria, se establecen roles transversales que se activan bajo condiciones específicas. El rol de Seguridad/PKI gestiona credenciales y certificados, valida cambios que afecten a mecanismos de autenticación o cifrado y participa en la respuesta ante incidentes de seguridad relacionados con acceso o identidad. El rol de Responsable de Cadena de Custodia interviene cuando exista riesgo de impacto forense, asegurando validaciones de integridad, trazabilidad y consistencia de los registros asociados a evidencias.

Como elemento de gobierno operativo, el sistema de ticketing actúa como repositorio único de evidencias y como mecanismo de trazabilidad del mantenimiento: toda incidencia y toda intervención preventiva o evolutiva debe quedar registrada, vinculando el componente afectado (servicio/módulo/versión), la evidencia técnica asociada (correlación con telemetría, logs y trazas), las acciones ejecutadas y el resultado verificado. Este registro constituye la base para auditoría, mejora continua y análisis de recurrencia.

Bajo este marco organizativo, THOT se concibe para operar de forma exclusiva en un entorno de CPD / nube privada, bajo un modelo de conectividad permanente. Este supuesto condiciona los recursos mínimos de infraestructura y las capacidades operativas que deben estar disponibles desde el inicio para sostener el servicio en régimen continuo. Por ello, el alcance de recursos se estructura en las siguientes dimensiones:

Dimensión	Alcance operativo
Infraestructura (HW/CPD)	Servidores físicos, cabinas/almacenamiento físico, red física (switching), firewalls/balanceadores físicos si aplica, alimentación y climatización (si se incluye en el contrato del CPD), firmware/BIOS, repuestos y garantías. Gestión de capacidad y sustitución por fallo/obsolescencia.

Cofinanciado por la Unión Europea
Fondos Europeos

Infraestructura (SW/plataforma base)	Virtualización, contenedorización y orquestación, clústeres de ejecución, storage lógico (SSD/NVMe para BD y servicios transaccionales; object storage para data lake), networking lógico (segmentación, políticas, balanceo). Hardening y configuración base.
Servicios centrales	Microservicios desplegados como contenedores: Espacio de datos (PostgreSQL, MongoDB, Ceph, Neo4j), Procesamiento (pipelines Apache Airflow/Dagster, Kafka), IA/ML (modelos Herta, LLMs/SLMs, RAG), Coordinación/gestión (BPMN, optimización), Alertas, CoC, Comunicaciones (MQTT, gRPC, WebRTC, WebSockets), UI (frontend centralizado).
Monitorización, diagnóstico y alertas	Observabilidad centralizada de métricas, logs y trazas. Stack con OpenTelemetry, Alloy, Prometheus, Loki, Tempo y Langfuse. Dashboards en tiempo real con Grafana. Alertas por severidad con escalado a soporte L1/L2/L3.
Gestión de incidencias	Sistema de ticketing como repositorio único de evidencias operativas; cada ticket vincula componente afectado (servicio, nodo, versión), telemetría asociada, acciones ejecutadas y resultado verificado. Clasificación por severidad y registro de RCA para incidencias críticas.
Evidencias y trazabilidad	Evidencias vinculadas al ticket con referencias a métricas/logs/trazas, versiones de contenedores/configuración y hashes de artefactos desplegados. Auditoría completa de cambios en plataforma central.
Cadena de Custodia y validez forense	Registro de eventos de CoC en componente central; auditoría de operaciones sobre evidencias; verificación de integridad y sellado temporal. CoC disponible para auditoría y trazabilidad judicial.
Cambios y actualizaciones	Despliegue controlado mediante CI/CD y GitOps, registros de contenedores, estrategias blue/green o canary, verificación post-cambio con pruebas de regresión y rollback si se detecta degradación. Coordinación con Seguridad/PKI cuando afecte credenciales o mTLS.

***Tabla 2. Dimensiones de alcance operativo del PMS del Lote 1 (THOT**). Alcance por dimensión y elementos incluidos en la operación y mantenimiento de la plataforma central para asegurar continuidad, seguridad, trazabilidad y control de cambios.*

3.1. Responsabilidades por Módulo y por Servicio

En primer lugar, se establece la propiedad operativa (service ownership) de los componentes principales de THOT (Lote 1), asignando un responsable por servicio como punto único de referencia para la gestión técnica, la continuidad operativa y la evolución controlada de cada módulo. Para cada servicio se delimitan responsabilidades mínimas (mantenimiento preventivo y correctivo, control de versiones, seguridad y trazabilidad, y, cuando corresponda, calidad del dato) así como el conjunto mínimo de evidencias que deben generarse y registrarse para permitir verificación en auditoría, incluyendo los requisitos de CoC cuando resulten aplicables. Esto queda indicado en la **Tabla 3**.

Además, se definen las responsabilidades asociadas a los procesos operativos transversales que gobiernan el ciclo de vida de THOT (Lote 1). La asignación se formaliza mediante una matriz mini-RACI (Responsible/Accountable/Consulted/Informed) y establece, para cada proceso, el flujo operativo mínimo, los artefactos y evidencias auditables, y las particularidades de operación en el entorno de plataforma central en CPD/nube privada. Con ello se garantiza la continuidad del servicio, la trazabilidad de las actuaciones (incluyendo los requisitos de Cadena de Custodia cuando correspondan) y la interoperabilidad con el Lote 2 y con sistemas policiales externos. Como se puede ver en la **Tabla 4**.

3.2. Herramientas operativas y plataformas de soporte

Para ejecutar de forma consistente los procesos operativos transversales definidos en los apartados anteriores (gobierno del ciclo de vida, operación continua, gestión de cambios, respuesta a incidencias, preservación de evidencias y continuidad del servicio), THOT requiere un conjunto mínimo de herramientas de soporte. Estas herramientas constituyen la base de la operación en entorno CPD/nube privada y permiten: (i) mantener inventariado y controlado el estado de la plataforma (servicios, versiones, configuraciones y dependencias); (ii) desplegar y actualizar componentes de forma repetible y auditada; (iii) observar el comportamiento extremo a extremo mediante telemetría centralizada; (iv) registrar toda intervención como evidencia verificable (incidencias, cambios, mantenimientos y retiradas); y (v) sostener la continuidad mediante backup y recuperación ante desastres. Todo esto queda resumido en la **Tabla 5**.

Cofinanciado por la Unión Europea
Fondos Europeos

Módulo / Servicio	Responsable (rol)	Tecnologías asociadas	Responsabilidades operativas mínimas (y evidencias)
Espacio de datos	A designar (Responsable Técnico de Persistencia y Gobierno del Dato)	PostgreSQL; MongoDB; InmuDB (CoC), Ceph(S3); Neo4j; Qdrant/Milvus, KurrentDB	**Responsabilidades:** (i) ejecución de backups y restauraciones con verificación; (ii) control de capacidad y crecimiento (almacenamiento e índices); (iii) gobierno del dato (esquemas, contratos, validaciones); (iv) aseguramiento de calidad del dato (completitud, consistencia, tiempos de ingesta); (v) seguridad (cifrado en reposo, RBAC y auditoría). **Evidencias:** informes de backup/restore y pruebas de restauración, dashboards de ocupación/capacidad, métricas de calidad, logs de auditoría y cambios, inventario y revisión periódica de permisos.
Procesamiento de datos	A designar (Responsable Técnico de Pipelines y Transformación de Datos)	Apache Airflow/Dagster; Apache Kafka/NATS; Redis; pipelines de transformación/enriquecimiento; integración con APIs	**Responsabilidades:** (i) operación y salud de pipelines (éxito/fallo, latencia, volumen); (ii) resiliencia (reintentos, idempotencia y recuperación); (iii) versionado y despliegue controlado de flujos; (iv) monitorización y alertado específicos; (v) coordinación de cambios de esquemas con Data Space e interoperabilidad. **Evidencias:** dashboards de ejecución, logs/trazas de jobs, histórico de alertas y resolución, registro de versiones y changelog de pipelines, evidencias de pruebas de regresión de flujos críticos.
Inteligencia Artificial y Machine Learning (IA/ML)	A designar (Responsable Técnico de Modelos IA y MLOps)	Reconocimiento facial; LLM/SLM; RAG; grafos/correlación; XAI; model registry; serving de inferencia; monitorización de drift, Nvidia Triton/vLLM	**Responsabilidades:** (i) registro y versionado de modelos, datasets y métricas; (ii) validación previa, despliegue controlado y mecanismos de rollback; (iii) monitorización de rendimiento y deriva (drift), criterios de reentrenamiento y retirada; (iv) explicabilidad, trazabilidad de inferencias y control de sesgos cuando aplique; (v) control de acceso y auditoría de cambios. **Evidencias:** entradas en model registry, informes de validación (offline/online), evidencias de despliegue/rollback, alertas de drift y acciones ejecutadas, informes XAI y auditorías de acceso/cambios.
Coordinación y gestión	A designar (Responsable Técnico de Orquestación de Procesos y BPMN)	Motor BPMN; reglas de asignación; integración con UI; APIs de coordinación	**Responsabilidades:** (i) gobierno, versionado y publicación de procesos; (ii) revisión de flujos y optimización (tiempos, cuellos de botella); (iii) alineación con procedimientos operativos; (iv) métricas de carga, backlog y cumplimiento de SLA internos; (v) integración con alertas y UI para seguimiento de casos y tareas. **Evidencias:** repositorio y versiones BPMN, dashboards de ejecución de procesos, métricas de asignación, actas o informes de revisión periódica y acciones correctivas.
Gestión de alertas	A designar (Responsable Técnico de Alertas Tempranas)	Motor de reglas; correlación de eventos; notificaciones (UI y otros canales habilitados); integración con flujos	**Responsabilidades:** (i) definición y mantenimiento del catálogo de alertas (umbrales, severidad, destinatarios, acciones); (ii) revisión periódica y ajuste por falsos positivos/negativos; (iii) operación del servicio de alertado (latencia, disponibilidad, entrega); (iv) integración con ticketing

			y con flujos de trabajo. **Evidencias:** catálogo versionado, histórico de alertas y acciones, informes de efectividad, registros de integración con ticketing y métricas de entrega/latencia.
Cadena de Custodia centralizada (CoC Central)	A designar (Responsable de CoC y Trazabilidad Forense)	Sellado temporal; firma digital; auditoría de eventos; integración con InmuDB	**Responsabilidades:** (i) verificación periódica de integridad; (ii) backups y restauraciones con verificación; (iii) validación de eventos CoC (campos, estados, reglas); (iv) auditorías periódicas de trazabilidad; (v) consolidación y coherencia de eventos procedentes del Lote 2; (vi) control de acceso y auditoría de consultas. **Evidencias:** informes de integridad/verificación, logs de backup/restore y pruebas de restauración, auditorías CoC, métricas de sincronización y consistencia, logs de accesos/consultas.
Comunicaciones e interoperabilidad	A designar (Responsable Técnico de Integraciones y APIs)	APIs REST; OpenAPI/AsyncAPI; gRPC; WebSockets; WebRTC; MQTT; Kafka; mTLS; PKI/certificados	**Responsabilidades:** (i) gobierno y versionado de contratos de integración; (ii) monitorización por integración (latencia, errores, disponibilidad); (iii) pruebas recurrentes de integración y compatibilidad entre versiones; (iv) seguridad de comunicaciones (mTLS) y coordinación del ciclo de vida de certificados con Seguridad/PKI. **Evidencias:** repositorio de contratos y versiones, dashboards por integración, resultados de pruebas automáticas y reportes, inventario de endpoints y dependencias, evidencias de rotación/renovación de certificados y logs asociados.
Interfaz de usuario (UI Central)	A designar (Responsable Técnico de Frontend y Experiencia de Usuario)	Frontend (React/Vue); APIs REST/WebSockets; dashboards; RBAC; accesibilidad (WCAG); multiidioma	**Responsabilidades:** (i) versionado y despliegue controlado del frontend; (ii) pruebas de regresión de funciones críticas; (iii) monitorización de rendimiento y experiencia de usuario; (iv) cumplimiento de accesibilidad y gestión multiidioma; (v) seguridad (integración con autenticación/autorización y auditoría de permisos). **Evidencias:** repositorio y changelog, resultados de pruebas, métricas de rendimiento/errores, evidencias de revisión WCAG, logs de autorización/permisos y revisiones de roles.
Infraestructura (Infrastructure & Platform)	A designar (Responsable Técnico de Infraestructura CPD/Cloud)	Servidores físicos/virtuales; Kubernetes; storage (SSD/NVMe, S3); networking (firewalls/balanceo); virtualización; backup/DR	**Responsabilidades:** (i) capacidad y rendimiento (CPU/RAM/storage/red) con umbrales y alertas; (ii) continuidad (redundancia, failover y recuperación); (iii) parches y actualizaciones planificadas de plataforma; (iv) seguridad de plataforma (segmentación, hardening y control de accesos). **Evidencias:** dashboards de capacidad/rendimiento, informes de pruebas de continuidad/recuperación, calendario y evidencias de parches, auditorías de seguridad de plataforma y logs de accesos.
Formación inmersiva y evaluación cognitiva (Training & Assessment)	A designar (Responsable de Formación Técnica y XR)	XR multiusuario; escenarios; motor neuroergonómico; contenidos formativos; simulaciones	**Responsabilidades:** (i) actualización y versionado de contenidos y escenarios; (ii) planificación y ejecución de campañas de formación/evaluación por rol; (iii) registro de lecciones aprendidas e incidencias recurrentes para mejora continua; (iv) operación del entorno XR (disponibilidad, latencia y estabilidad). **Evidencias:** repositorio de contenidos y versiones, calendario de campañas y resultados, registro de lecciones aprendidas, dashboards del entorno XR y métricas de uso/estabilidad.

Seguridad Transversal y Control de Acceso	A designar (Responsable Técnico de Seguridad y Ciberseguridad)	OpenZiti (Zero Trust); HashiCorp Vault (Secretos); Keycloak (IAM/SSO); Istio (Service Mesh); Cert-manager; Falco (Runtime); Trivy (Escaneo); Kyverno (Políticas)	**Responsabilidades**: (i) operación y mantenimiento de la red Zero Trust y Service Mesh (políticas de acceso y mTLS); (ii) gestión del ciclo de vida de secretos, claves y certificados (rotación y custodia); (iii) administración de identidad (SSO, MFA, federación); (iv) monitorización de seguridad en tiempo real (amenazas en runtime) y escaneo continuo de vulnerabilidades; (v) gobierno de políticas de seguridad como código. **Evidencias**: logs de acceso y auditoría de Vault/Keycloak, informes de vulnerabilidades de imágenes (Trivy), alertas de seguridad (Falco), estado de controladores OpenZiti, registros de rotación de credenciales y reportes de cumplimiento de políticas (Kyverno).

Tabla 3. Dimensiones de recursos y capacidades operativas mínimas de THOT

Proceso	R (Responsible)	A (Accountable)	C (Consulted)	I (Informed)	Flujo operativo (resumen)	Evidencias mínimas
Despliegue de servicios (alta de microservicio)	Responsable Técnico del servicio (Service Owner)	Dirección del Servicio (Lote 1)	Infraestructura/Plataforma; Seguridad/PKI; Interoperabilidad (si afecta contratos); CoC (si gestiona evidencias)	Operación (L1/L2)	Alta del servicio y dependencias; asignación de recursos y políticas; aplicación de baseline; provisión de credenciales/certificados; despliegue del artefacto aprobado; validación técnica; alta en observabilidad y catálogo de alertas; paso a estado operativo.	CMDB/inventario de servicios; baseline aprobada; inventario de credenciales/certificados; registro CI/CD (artefacto y hash); checklist de aceptación; evidencia de alta en observabilidad y runbook inicial.
Cambios y actualizaciones de servicios	Responsable Técnico del servicio afectado	Responsable de Gestión de Cambios	Infraestructura/Plataforma; Seguridad/PKI; Interoperabilidad (si afecta contratos); CoC (si afecta evidencias)	Operación (L1/L2/L3); Usuarios internos afectados (si procede)	Ticket de cambio; análisis de impacto; aprobación; pruebas; despliegue progresivo; verificación; rollback/mitigación; actualización documental y cierre.	Ticket de cambio completo; resultados de pruebas; registro de despliegue (versión/estrategia); métricas pre/post; evidencia de rollback (si aplica); CMDB y changelog actualizados.
Actualización de modelos de IA	Responsable Técnico de Modelos IA y MLOps	Dirección del Servicio (Lote 1)	Seguridad/PKI; CoC (si aplica); Infraestructura/Plataforma; Interoperabilidad (si aplica)	Operación (L1/L2/L3); Servicios	Registro de versión; validación y no degradación; canary y expansión gradual; monitorización intensiva;	Model registry; informe de validación/benchmark; evidencias de despliegue por fases; dashboards/alertas post-cambio;

				consumidores (si procede)	rollback si degradación; criterios de retirada; cierre.	acciones; criterios de retirada; changelog del modelo.
Monitorización operativa y alertas	Operación (L1)	Dirección del Servicio (Lote 1)	Service Owners; Infraestructura/Plataforma; Seguridad/PKI	Soporte L2/L3; Gestión de Cambios (si deriva en cambio)	Supervisión continua; correlación telemetría; clasificación por severidad; apertura de incidencias; runbooks; escalado; análisis de tendencias y preventivos.	Catálogo de alertas versionado; dashboards por servicio; runbooks; tickets vinculados; informes de tendencias y acciones recomendadas.
Respuesta a incidencias operativas	Operación L1 (triaje); L2 (diagnóstico); L3 (resolución)	Dirección del Servicio (Lote 1)	Service Owner afectado; Seguridad/PKI; CoC (si aplica); Infraestructura/Plataforma	Usuarios internos afectados y solicitante (si procede)	Triage; contención; escalado; diagnóstico; corrección; verificación; RCA; postmortem si crítica.	Ticket con severidad y timeline; evidencias (métricas/logs/trazas); verificación; RCA; postmortem y plan preventivo (si aplica).
Retirada o evolución de servicios	Responsable Técnico del servicio (Service Owner)	Dirección del Servicio (Lote 1)	Interoperabilidad; Seguridad/PKI; CoC; Infraestructura/Plataforma	Operación (L1/L2/L3); Usuarios afectados	Análisis de impacto; plan de migración/desmantelamiento; comunicación; preservación de datos/evidencias; revocación accesos; actualización CMDB; validación final; lecciones aprendidas.	Plan aprobado; comunicaciones; evidencias de preservación/integridad; revocaciones/rotación; CMDB "retirado"; documentación EOL y lecciones aprendidas.
Alta/provisión de hardware y recursos CPD	Infraestructura/Plataforma (CPD)	Responsable de Infraestructura CPD/Cloud	Seguridad/PKI (si hay HSM/PKI/appliances o segmentación); Operación; Dirección del Servicio (para priorización)	Service Owners; Gestión de Cambios	Solicitud/capacidad; aprobación; provisión o ampliación (servidores, almacenamiento, red, balanceadores/firewalls si aplica); inventariado; hardening y baseline; pruebas de aceptación (rendimiento/conectividad); puesta en servicio; actualización de CMDB y monitorización.	Inventario/CMDB de activo (serial, ubicación, rol); baseline de configuración y hardening; evidencias de pruebas de aceptación; registro de parches/firmware inicial; alta en monitorización; acta de puesta en servicio.
Mantenimiento de hardware/firmware y plataforma base *(recomendado)*	Infraestructura/Plataforma (CPD)	Responsable de Infraestructura CPD/Cloud	Seguridad/PKI; Operación; Gestión de Cambios	Dirección del Servicio; Service Owners	Planificación de ventana; evaluación de impacto; actualización de firmware/BIOS/hipervisor y componentes base; verificación de	Ticket de cambio; calendario de mantenimiento; versiones antes/después; resultados de pruebas; métricas pre/post; incidencias asociadas y cierre.

Cofinanciado por la Unión Europea
Fondos Europeos

					continuidad; rollback/mitigación; cierre con evidencias.	
Retirada/sustitución de hardware (obsolescencia o fallo)	Infraestructura/Plataforma (CPD)	Responsable de Infraestructura CPD/Cloud	Dirección del Servicio; Seguridad/PKI (si hay credenciales/certs); Operación; Compras/Administración (si aplica)	Service Owners; Gestión de Cambios	Detección de obsolescencia/fallo; análisis de impacto; plan de sustitución; migración de cargas/datos; borrado seguro y baja; revocación de accesos/credenciales asociadas; actualización de inventario; cierre y lecciones aprendidas.	Informe de impacto y plan; evidencias de migración/validación; certificado o evidencia de borrado seguro; registros de baja en CMDB; evidencias de revocación de accesos; acta de retirada/sustitución y actualización de capacidad.
Seguimiento OPEX y sostenibilidad económica	Operación / Service Management	Dirección del Servicio (Lote 1)	Infraestructura/Plataforma; Service Owners; Compras/Administración (si aplica)	Comité de Dirección	Captura periódica de costes; consolidación trazable; tendencias y proyección; optimizaciones; informe al Comité; registro de decisiones.	Cuadro OPEX trazable (métricas/CMDB/tickets/facturas); informes de tendencia/proyección; análisis de sensibilidad; decisiones documentadas.

Tabla 4. Mini-RACI de procesos operativos transversales del ciclo de vida de THOT (Lote 1), *con responsables, flujo mínimo y evidencias auditables para operación, cambios, incidencias y gestión de infraestructura*

Categoría	Herramienta / Plataforma	Propósito	Requisitos mínimos	Requisitos trazados
Inventario y CMDB	CMDB corporativa (Configuration Management Database)	Inventario de servicios, versiones, configuraciones, dependencias e infraestructura (cómputo, almacenamiento y red).	API para consulta/actualización automatizada; integración con CI/CD/GitOps; modelado de relaciones de dependencia; auditoría de cambios; soporte de estados (activo, en cambio, retirado).	Inventario/CMDB; control de configuración; trazabilidad de cambios; gestión de dependencias.
Orquestación de contenedores	Kubernetes (clúster corporativo o dedicado a THOT)	Despliegue, escalado y actualización de microservicios, incluyendo políticas de disponibilidad.	Alta disponibilidad del plano de control; almacenamiento persistente (PV/PVC); red de servicios (CNI); integración con registry; RBAC; soporte de estrategias de despliegue (rolling/blue-green/canary); políticas de recursos.	Arquitectura modular; escalabilidad; alta disponibilidad; control de accesos; operación de microservicios.

Cofinanciado por la Unión Europea
Fondos Europeos

Registry de contenedores	Registry privado (Harbor, ACR, ECR u otra solución corporativa)	Almacenamiento de imágenes con versionado, escaneo y control de integridad.	Escaneo automático; firma de imágenes (cuando se habilite); políticas de retención; RBAC; backup; trazabilidad de origen (repositorio/commit).	Seguridad de la cadena de suministro; integridad de artefactos; control de versiones; auditoría.
CI/CD	Pipeline CI/CD (GitLab CI, Jenkins, Azure DevOps u otra solución corporativa)	Automatización de build, test, despliegue y rollback de servicios y modelos.	Integración con repositorios Git; tests automatizados (unitarios/integración/regresión); despliegue hacia Kubernetes/GitOps; notificaciones; auditoría de ejecuciones; segregación de entornos.	Calidad de despliegues; control de cambios; despliegue seguro; repetibilidad.
GitOps	Herramienta GitOps (ArgoCD)	Gestión declarativa de configuración y sincronización con el estado deseado en Git.	Sincronización automática; detección de drift; control de permisos; rollback; integración con Kubernetes; auditoría de cambios; separación por entornos.	Control de configuración; trazabilidad; consistencia de entornos; rollback y recuperación.
Observabilidad (métricas)	Prometheus + Grafana (o equivalente corporativo)	Recolección, almacenamiento y visualización de métricas de servicios e infraestructura.	Retención mínima 30 días; dashboards por servicio y plataforma; alertas integradas con ticketing; API de consulta; etiquetado por servicio/versión/entorno.	Monitorización; gestión de fallos; diagnóstico; análisis de tendencias.
Observabilidad (logs)	Loki	Agregación, indexación y búsqueda de logs centralizados.	Retención mínima 90 días para operación; retención extendida cuando aplique a auditoría; búsquedas avanzadas; RBAC; integridad y trazabilidad de acceso a logs; extracción por servicio/versión.	Operación y diagnóstico; auditoría; trazabilidad; control de accesos a evidencias operativas.
Observabilidad (trazas)	Tempo	Trazabilidad distribuida y análisis de latencia extremo a extremo.	Instrumentación (SDK/agentes); retención mínima 7 días; grafos de llamadas; análisis por spans; correlación con logs y métricas; etiquetado por servicio/versión.	Diagnóstico distribuido; rendimiento/latencia; dependencias; resolución de incidencias.
Ticketing	Sistema corporativo (Jira, ServiceNow u otro)	Gestión de incidencias, cambios, mantenimiento HW/CPD y retiradas; repositorio de evidencias operativas.	Workflows (incidencias/cambios/mantenimientos/retiradas); campos estructurados (servicio, versión, severidad, impacto); enlaces a métricas/logs/trazas; notificaciones; SLA/OLA; auditoría; reporting.	Gestión de incidencias y cambios; trazabilidad operativa; auditoría; cumplimiento de SLAs.

Runbooks	Repositorio (wiki/Confluence/Markdown en Git)	Procedimientos estándar para recuperación L1 y diagnóstico L2/L3, enlazados a alertas.	Control de versiones; búsqueda; vinculación a alertas; formato estructurado; revisión/aprobación; historial de cambios y caducidad/revisión periódica.	Operación estandarizada; respuesta a incidencias; consistencia; mejora continua.
Model Registry	MLflow Model Registry u otra solución	Registro de versiones de modelos con metadatos (datasets, métricas, responsable, fecha).	Versionado; almacenamiento de artefactos; metadatos extensibles; transiciones (candidate→staging→production→archived); API; integración con CI/CD; trazabilidad de entrenamiento/evaluación.	Gobierno de modelos; MLOps; trazabilidad y auditoría de IA; control de despliegues.
Backup/DR	Soluciones open source o solución corporativa (Veeam, Commvault, Cohesity u otra) + sitio DR	Backup de datos críticos y replicación a nodo/sitio secundario para recuperación ante desastre.	Backup automatizado; políticas de retención; verificación de integridad; pruebas periódicas de restauración; replicación a DR; definición y seguimiento de RTO/RPO.	Continuidad de negocio; recuperación ante desastre; protección de datos; disponibilidad.
PKI / Identidad	PKI corporativa o dedicada	Emisión/renovación/revocación de certificados (mTLS), soporte a firma digital y sellado temporal.	CA raíz e intermedias; HSM cuando aplique; renovación automatizada; CRL/OCSP; auditoría de emisión/revocación; integración con mecanismos de firma/sellado temporal cuando se habiliten.	Autenticación y cifrado; integridad; gestión de credenciales; auditoría de seguridad; comunicaciones seguras.

Tabla 5. Herramientas operativas de soporte para la operación, trazabilidad, seguridad, MLOps y continuidad

4. ESTRATEGIAS DE MANTENIMIENTO

Este apartado establece el marco operativo para mantener la plataforma THOT (Lote 1) estable, segura y disponible tras su despliegue, operando de forma exclusiva en la infraestructura de CPD/nube privada de Policía Científica. El mantenimiento abarca el conjunto completo de la solución, incluyendo tanto la infraestructura que la soporta (hardware y recursos del CPD, almacenamiento, red y plataforma base) como el software desplegado, y contempla asimismo la interoperabilidad entre lotes como capacidad crítica de operación. En consecuencia, el alcance comprende los servicios centrales de THOT desplegados como microservicios contenerizados (Espacio de Datos, Procesamiento de Datos, IA/ML, Coordinación y Gestión, Alertas, Cadena de Custodia centralizada, Comunicaciones e Interfaz de Usuario) junto con la plataforma subyacente necesaria para su ejecución, actualización controlada, monitorización y continuidad de servicio.

Adicionalmente, el mantenimiento contempla la gestión de la capa de integración e interoperabilidad con el Lote 2 y con sistemas policiales externos, actuando THOT como servidor/orquestador de servicios y punto de consolidación. Esta capa incluye la exposición y gobierno de APIs (REST/gRPC), la recepción y validación de datos y eventos (p. ej., MQTT/Kafka), el control de contratos e interfaces, la autenticación mutua y cifrado de comunicaciones (mTLS) y la preservación de la Cadena de Custodia centralizada. La relación con el Lote 2 se limita al ámbito de integración, sin intervenir en la operación interna de sus componentes: se mantiene y valida el cumplimiento de contratos de API, la consistencia de mensajes/eventos, los mecanismos de acuse de recibo (ACK/NACK) cuando proceda, la consolidación de evidencias de CoC y la verificación de integridad y coherencia de los datos recibidos.

Cuando se produzcan incidencias o degradaciones originadas en el Lote 2, el mantenimiento se aplica con un enfoque evidence-driven, circunscrito a lo observable y verificable desde THOT y a los mecanismos de integración, sustentado en: (i) observabilidad (métricas, logs y trazas) para detectar degradaciones, localizar el punto de fallo y confirmar la recuperación; (ii) un catálogo de alertas con condiciones de disparo, severidad, responsable, acción asociada y objetivo de reacción; (iii) ticketing trazable, de modo que toda actuación (manual o automática) quede registrada con sello temporal, activo afectado, versiones, evidencias técnicas y verificación posterior; y (iv) procedimientos cerrados (runbooks y checklists) para asegurar una ejecución homogénea, repetible y auditable.

La estrategia se articula en tres ejes complementarios: preventivo, correctivo y evolutivo**.** El **mantenimiento preventivo** establece baselines, umbrales y checklists de salud sobre los servicios centrales y la infraestructura CPD/nube (capacidad y saturación de recursos, versiones de contenedores y configuraciones, estado de pipelines y colas, deriva y rendimiento de modelos IA cuando aplique, validez y caducidad de certificados PKI, estado de integraciones e interoperabilidad, y controles de seguridad), con el objetivo de reducir la probabilidad de incidencias que comprometan la disponibilidad de la plataforma central, la calidad del ciclo de inteligencia o la trazabilidad forense.

El **mantenimiento correctivo** sigue un proceso trazable que comienza con la detección (alertas automáticas del stack de observabilidad o reportes de usuario), el triage inicial y la clasificación de severidad y alcance (servicios, integraciones o infraestructura afectada). Incluye medidas de contención proporcionales (pausa controlada de flujos, aislamiento lógico, limitación de tasas), recuperación (reinicio ordenado de servicios, rollback de versiones, reprovisión de configuración, rotación o regeneración de credenciales cuando proceda), verificación post-intervención (confirmación mediante métricas, logs y trazas de estabilidad y ausencia de regresión) y cierre con documentación de causa raíz, acciones preventivas y lecciones aprendidas. En todo caso se priorizan actuaciones reversibles y auditables, preservando continuidad operativa e integridad de la Cadena de Custodia.

El **mantenimiento evolutivo** gestiona cambios orientados a adaptar o mejorar la plataforma central frente a nuevas necesidades funcionales o dependencias técnicas, incorporando nuevas versiones de microservicios, modelos de IA, configuraciones de pipelines, contratos e interfaces de interoperabilidad, actualizaciones de dependencias y mejoras de hardening. Se ejecuta bajo control de cambios, con análisis de impacto (operación, seguridad, interoperabilidad con Lote 2, Cadena de Custodia y OPEX), validación previa en preproducción, despliegue escalonado (piloto y oleadas) con verificación post-cambio y capacidad de rollback automático o inmediato ante degradación detectada.

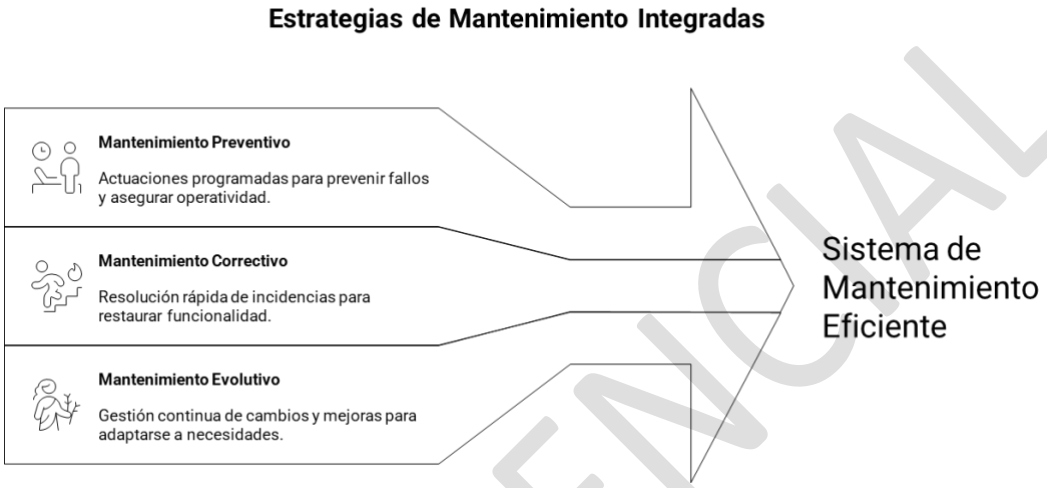

Figura 1. Resumen de las estrategias de mantemiento planteadas para THOT.

4.1. Mantenimiento preventivo

El mantenimiento preventivo de THOT (Lote 1) tiene como finalidad reducir la probabilidad de incidencias y anticipar degradaciones antes de que afecten a la operación del ciclo de inteligencia forense. Se aplica de forma sistemática sobre los servicios centrales, la infraestructura CPD/nube privada que los soporta y la capa de interoperabilidad con el Lote 2 y sistemas externos. El enfoque es evidence-driven: se basa en telemetría (métricas, logs y trazas), baselines de funcionamiento y umbrales de alerta. Toda desviación respecto a esos baselines se registra como evidencia verificable y queda vinculada a una acción preventiva planificada o a un ticket, preservando trazabilidad y facilitando auditoría.

4.1.1. Cadencias operativas y procedimientos verificables

El mantenimiento preventivo se ejecuta mediante cadencias definidas y procedimientos verificables. Cada revisión produce, como salida mínima, un parte de control con marca temporal, componente evaluado (servicio/módulo/activo), resultado (conforme/no conforme), referencia a la evidencia técnica (panel/consulta de métricas, logs o trazas; checklist cuando proceda) y, si aplica, el ticket asociado con severidad y responsable.

a) Revisión diaria (operación)

Orientada a asegurar continuidad operativa y detección temprana de fallos. Se apoya en dashboards y alertas proactivas, y valida:

- **Disponibilidad y salud de servicios críticos**: estado de health-checks, accesibilidad de APIs y endpoints de integración, tasa de errores y latencias de servicio. Se controlan, como mínimo, P95 de latencia y

tasa de errores 5xx para operaciones críticas, con umbrales definidos por servicio (no se fijan valores únicos si no existen SLAs internos ya aprobados).

- **Estabilidad de ejecución y colas**: reinicios anómalos, errores recurrentes y saturación de recursos a nivel de servicio y plataforma (CPU, memoria, almacenamiento e I/O), así como backlogs de procesamiento o mensajería (p. ej., tareas pendientes y retrasos sobre baseline).
- **Interoperabilidad con Lote 2**: disponibilidad del canal de integración, evolución del backlog de recepción, validación de conformidad de mensajes/eventos con la versión de contrato activa y trazabilidad de acuses de recibo cuando estén implementados (ACK/NACK), con alertas por ausencia o retrasos anómalos frente a baseline.
- **Seguridad e identidad**: vigencia de certificados y credenciales, eventos de autenticación fallida o denegaciones anómalas, y señales de comportamiento sospechoso en logs de seguridad, con escalado al rol Seguridad/PKI cuando proceda.
- **CoC centralizada (controles de integridad)**: verificaciones automáticas de consistencia y de la correlación básica entre volumen de evidencias recibidas y eventos de CoC generados/consolidados, detectando divergencias evidentes.

Cualquier condición anómala genera ticket preventivo con evidencias correlacionadas (servicio/activo, versión, y referencias a telemetría).

b) Revisión semanal

Orientada a consolidar tendencias y planificar acciones preventivas. Incluye:

- **Capacidad y tendencias de infraestructura**: consumo por servicio y por capa (cómputo, memoria, almacenamiento e I/O), identificación de crecimiento sostenido y previsión de saturación para activar escalado o reequilibrio antes de afectar a la operación.
- **Procesamiento y pipelines**: tasa de éxito/fallo por flujo crítico, latencias P95 frente a baseline, cuellos de botella (tareas lentas, dependencias bloqueantes) y reintentos anómalos.
- **IA/ML (cuando aplique)**: estabilidad de inferencia (errores/timeout), latencias, consumo de recursos y señales de deriva (drift) de datos o rendimiento cuando existan métricas de referencia definidas para cada modelo.
- **Conformidad de interoperabilidad**: tasa y causas de rechazos por validación de esquema/contrato, compatibilidad entre versiones y degradaciones de calidad/consistencia en datos recibidos.
- **Revisión de seguridad operativa**: accesos privilegiados, cambios relevantes de configuración, eventos de seguridad y cumplimiento de baselines de hardening, incluyendo seguimiento de vulnerabilidades críticas y estado de remediación.

Las desviaciones se traducen en acciones planificadas (reconfiguración, ajuste de recursos, corrección de pipeline, actualización controlada, coordinación con Lote 2) registradas en tickets y consolidadas en un informe semanal.

c) Revisión mensual / ventana de mantenimiento

Orientada a control de versiones, deuda técnica y validaciones coordinadas. Se distinguen dos bloques:

- **Actualizaciones controladas**: actualización de componentes de plataforma y servicios (imágenes, configuraciones, dependencias críticas, reglas/umbrales de alertado y, si procede, modelos de IA), con despliegue escalonado y capacidad de rollback.
- **Validación de interoperabilidad extremo a extremo**: ejecución de pruebas coordinadas con Lote 2 cuando existan cambios en contratos, autenticación (mTLS/certificados) o flujos de mensajería; verificación de consistencia de datos, trazabilidad de acuses y consolidación correcta en CoC.

Cofinanciado por la Unión Europea
Fondos Europeos

Las periodicidades y umbrales se ajustan progresivamente con datos reales de operación (tendencias, recurrencia, consumo y patrones de interoperabilidad), dejando registro de cada ajuste y su evidencia justificativa.

4.1.2. Actividades preventivas de resiliencia y seguridad

Además de las revisiones periódicas, se incorporan actividades preventivas específicas para asegurar resiliencia y seguridad:

- **Pruebas de recuperación y disponibilidad**: ejecuciones controladas de recuperación (p. ej., reinicios verificados de servicios críticos y comprobación de dependencia), con verificación basada en telemetría y checklist de aceptación.
- **Rollback verificable**: disponibilidad de versiones inmutables y mecanismos de reversión para servicios y configuraciones; verificación periódica de que los artefactos necesarios para rollback están accesibles y son coherentes (imágenes, manifests/configuración y, si aplica, versiones de modelos).
- **Gestión preventiva de certificados/credenciales**: monitorización de caducidad, renovación planificada, verificación de establecimiento de mTLS y trazabilidad de rotaciones, con escalado al rol Seguridad/PKI ante riesgos de expiración o anomalías.
- **Integridad y seguridad de la cadena de suministro**: escaneo de vulnerabilidades en artefactos, control de procedencia y políticas de despliegue que impidan ejecutar imágenes o configuraciones no autorizadas.
- **Hardening y dependencias**: revisión periódica de configuraciones de seguridad (políticas de red, privilegios de ejecución, secretos) y seguimiento de vulnerabilidades relevantes con plan de remediación.
- **Backup y restauración**: ejecución de backups conforme a política y **pruebas periódicas de restauración** sobre activos críticos (datos y configuración), asegurando evidencias de integridad y tiempos de recuperación coherentes con objetivos operativos.
- **Interoperabilidad preventiva con Lote 2**: pruebas periódicas de contratos/formatos, autenticación y validación de consistencia, incluyendo verificación de consolidación en CoC cuando aplique.
- **Revisión de salud HW/CPD y degradación**: comprobación periódica de alarmas de hardware, componentes en estado degradado y verificación de redundancia efectiva (paths de almacenamiento, enlaces, fuentes), con acciones correctivas planificadas.
- **Ciclo de firmware/EOL**: seguimiento trimestral de firmware/BIOS/controladoras y de fin de soporte/garantía, alineado con ventanas de mantenimiento y con registro de riesgo y plan de sustitución.

4.2. Mantenimiento correctivo

El mantenimiento correctivo de THOT (Lote 1) define un ciclo operativo **cerrado, trazable y orientado a recuperar el servicio** ante incidencias que afecten a la plataforma central. El proceso está diseñado para ser **forensic-safe**: prioriza el diagnóstico remoto y las acciones reversibles, preserva la integridad y trazabilidad de la información, y activa verificaciones específicas cuando exista impacto potencial sobre la Cadena de Custodia, evidencias o metadatos. Todas las etapas se sustentan en evidencias verificables mediante observabilidad (métricas, logs y trazas), ticketing, control de cambios y runbooks.

El flujo de trabajo del mantenimiento correctivo es el siguiente:

- **Incidencia detectada.** La incidencia se detecta por alertas automáticas del stack de observabilidad o por reporte de usuarios/operadores. La detección debe quedar respaldada por evidencia inicial verificable (alerta, panel o consulta reproducible de métrica/log/traza), con marca temporal y

componente afectado. Se consideran señales típicas: indisponibilidad de API/UI, degradación sostenida de latencia, incremento anómalo de errores 5xx/timeout, reinicios repetidos, degradación de colas/backlogs de procesamiento o mensajería, fallos de persistencia (timeouts o errores de base de datos), degradación de almacenamiento (latencia/I/O), fallos de validación de mensajes/contratos en integraciones, y fallos de autenticación/cifrado (mTLS, expiración de certificados, errores de handshake).

- **Registrar incidencia.** Se crea un ticket manual o automático desde el sistema de alertas. El registro incluye, como mínimo: identificador único, fecha/hora, origen (alerta/usuario), servicio o módulo afectado (y dependencias presumibles), entorno (producción/preproducción), severidad inicial, impacto operativo, y referencias a la evidencia técnica inicial (panel/alerta, consulta de métricas, logs y trazas). Deben consignarse las versiones relevantes (imagen/artefacto desplegado, configuración asociada y, cuando aplique, versión de contrato de integración o versión de modelo IA).
- **Clasificar y acotar severidad.** Tras un triaje rápido se confirma alcance y severidad, considerando: impacto en disponibilidad o degradación funcional, afectación a servicios críticos, riesgo de propagación por dependencias, impacto en integraciones con Lote 2 y sistemas externos, riesgo de seguridad (acceso, identidad, anomalías), y riesgo forense (posible afectación de evidencias o CoC). Esta clasificación fija objetivos de reacción y resolución, activa el circuito de escalado (L1/L2/L3, Service Owner, Infraestructura, Seguridad/PKI y Responsable de CoC cuando proceda) y queda reflejada en el ticket.
- **Diagnóstico basado en evidencia.** El diagnóstico se realiza correlacionando métricas, logs y trazas a través de la cadena de dependencias, evitando acciones que puedan alterar indebidamente evidencias o registros. Se verifican, como mínimo: estado de servicios y pods, latencias y errores por endpoint, saturación y límites de recursos, salud de colas y pipelines, conectividad y errores en integraciones, estado de certificados/credenciales, y métricas de almacenamiento/bases de datos. Cuando la evidencia no sea suficiente, se habilitan trazas adicionales o logs de nivel controlado durante una ventana acotada, dejando constancia en el ticket y evitando exposición de datos sensibles.
- **Contención (medidas reversibles).** Antes de aplicar cambios, se priorizan medidas de contención para estabilizar la plataforma: limitación de tasa (rate limiting), circuit breakers, aislamiento lógico de un flujo degradado, pausa controlada de ingestas o pipelines no críticos, desactivación temporal de funcionalidades no esenciales, o reducción controlada de carga. Estas acciones deben quedar registradas con justificación, alcance y verificación posterior.
- **Recuperación (restauración del servicio).** Se ejecutan acciones de recuperación escalonadas y auditables: reinicio ordenado de servicios afectados, reprogramación o relanzamiento de pipelines, limpieza controlada de colas con garantías de no pérdida (según política), reprovisión de configuración declarativa, rollback a versión previamente aprobada, y, cuando aplique, rotación/revocación de credenciales o certificados. En incidencias de infraestructura (CPD), se incluyen actuaciones coordinadas de plataforma: reubicación de cargas, aislamiento de nodo degradado, recuperación de volúmenes o restauración desde backup conforme a procedimiento, siempre con preservación de evidencias.
- **Verificación post-intervención y ventana de observación.** La restauración se valida mediante evidencias: estabilidad de error rate, latencias dentro de baseline, ausencia de reinicios repetidos, drenaje de colas y normalización de throughput, y recuperación de dependencias (bases de datos, almacenamiento, mensajería). En caso de incidencias que afecten a interoperabilidad, se verifica la conformidad de contratos y la recuperación de flujo (recepción/procesamiento/respuesta) con trazabilidad de mensajes. Se mantiene una ventana de observación proporcional a la severidad antes de declarar resuelta la incidencia.
- **Controles forenses específicos (cuando aplique).** Si la incidencia ha podido afectar a evidencias, metadatos o CoC, se ejecutan verificaciones adicionales: consistencia e integridad de CoC, completitud de eventos esperados, coherencia temporal y trazabilidad de operaciones realizadas durante la incidencia. Si se aplicaron medidas como pausa de ingesta o reprocesado, se documenta el criterio aplicado para garantizar integridad y reproducibilidad.
- **Documentar intervención y actualizar ticket.** Toda acción ejecutada se documenta en el ticket con quién/cuándo/qué, componente y versión afectada, evidencias antes/después, runbook aplicado y resultado verificado. El ticket mantiene estados (en análisis, contención, recuperación, observación), registra hitos temporales (detección, primera respuesta, inicio de recuperación, restauración, cierre) y

Cofinanciado por la Unión Europea
Fondos Europeos

conserva enlaces a consultas reproducibles de telemetría para auditoría y cálculo de MTTR y cumplimiento de objetivos operativos.

- **Cierre con causa raíz y mejora continua.** La incidencia se considera resuelta cuando el servicio cumple los umbrales operativos acordados, se mantiene estable durante la ventana de observación y, cuando corresponda, la interoperabilidad y/o la CoC han sido verificadas. El cierre incluye causa raíz o causa probable, acciones ejecutadas, evidencias asociadas, lecciones aprendidas y tareas de seguimiento preventivas o evolutivas. Cuando la causa requiera un cambio estructural (optimización, hardening, actualización, ajuste de contratos, refuerzo de capacidad), se abre una solicitud evolutiva vinculada al ticket para asegurar trazabilidad del aprendizaje y reducción de recurrencia y OPEX.

4.3. Mantenimiento evolutivo

El mantenimiento evolutivo de THOT (Lote 1) establece el proceso controlado para introducir mejoras funcionales, optimizaciones de rendimiento y ajustes técnicos derivados de la operación, las pruebas y la gestión de vulnerabilidades, sin degradar la disponibilidad de la plataforma central, sin comprometer la ciberseguridad y preservando la trazabilidad operacional y la Cadena de Custodia cuando aplique. Este mantenimiento se gobierna como un proceso de gestión del cambio con decisiones trazables, despliegues reversibles y evidencias verificables, aplicable tanto a los microservicios y pipelines de la plataforma como a la infraestructura en sí (configuración, hardening, dependencias, y componentes de continuidad). Asimismo, incluye la evolución controlada de la capa de interoperabilidad con Lote 2 y con sistemas externos (contratos de integración, validaciones, mecanismos de autenticación y compatibilidad de versiones).

Principios operativos (continuidad, seguridad y forense). Las actualizaciones se ejecutan en ventanas controladas y con mecanismos de reversión. Ningún cambio debe introducir pérdidas de trazabilidad, degradar la observabilidad mínima ni comprometer la consistencia de evidencias o metadatos. Cualquier modificación que afecte a interoperabilidad (APIs REST/gRPC, contratos OpenAPI/AsyncAPI, mensajería/eventos) requiere verificación explícita extremo a extremo con los sistemas integrados. Toda actuación evolutiva deja rastro en ticketing y en los registros de despliegue (versión, huella del artefacto, fecha/hora, entorno, servicio afectado, resultado y rollback si procede).

4.3.1. Flujo mínimo de mantenimiento evolutivo

1) Demanda / Solicitud de cambio:

1. **Demanda / solicitud de cambio.** La evolución se inicia mediante una solicitud de cambio procedente de resultados de pruebas, incidencias recurrentes, necesidades operativas de Policía Científica o requisitos técnicos (hardening, compatibilidad, obsolescencia de dependencias, capacidad). La solicitud identifica objetivo, componente afectado (servicio, pipeline, integración, plataforma base o infraestructura), contexto, urgencia, impacto esperado y referencias a evidencias (tickets previos, métricas, alertas, KPIs/SLA internos).
2. **Análisis de impacto (operación, CoC, seguridad, interoperabilidad y coste).** Se evalúa el impacto sobre: (i) disponibilidad y rendimiento de la plataforma central (latencia, throughput, colas/backlogs, capacidad); (ii) Cadena de Custodia y trazabilidad cuando exista afectación a evidencias, metadatos o reprocesados; (iii) ciberseguridad (superficie de ataque, credenciales/certificados, hardening, dependencias con CVEs); (iv) interoperabilidad (contratos, compatibilidad entre versiones, validaciones, mTLS y códigos de error); y (v) impacto en OPEX (soporte, operación, licencias, consumo de recursos). Se identifican riesgos, mitigaciones y el plan de validación mínimo.
3. **Priorización y aprobación.** La solicitud se prioriza con criterios explícitos: impacto operativo, criticidad de seguridad, riesgo forense, dependencia de integraciones, deuda técnica y coste/OPEX. La aprobación designa responsables (Service Owner y responsables transversales cuando aplique), fija ventana de

despliegue, define criterios de aceptación y criterios de parada/rollback. La decisión queda registrada en el ticket con su justificación.

4. **Diseño técnico y planificación de entrega.** Se define la solución y los artefactos a modificar, diferenciando:

   - **Cambios de servicio**: imágenes, configuración declarativa, parámetros, reglas de alertas, runbooks, dependencias.
   - **Cambios de datos/procesamiento**: esquemas, validaciones, pipelines, estrategias de reprocesado y control de idempotencia.
   - **Cambios de IA/ML (cuando aplique)**: versión de modelo, serving, criterios de validación y retirada, monitorización de deriva.
   - **Cambios de interoperabilidad**: versión de contrato, compatibilidad y mecanismos de validación/errores, pruebas E2E.
   - **Cambios de plataforma/infraestructura**:parches, hardening, actualizaciones base y continuidad.

   En todos los casos se define plan de despliegue, plan de reversión, prerequisitos (baseline, versiones mínimas, credenciales), y plan de comunicación si procede.

5. **Implementación y validación en entorno controlado.** La solución se implementa y valida en preproducción o entorno de prueba con: pruebas funcionales, pruebas de rendimiento (latencia/throughput, colas), pruebas de seguridad (configuración, permisos, vulnerabilidades críticas) y validación de integraciones extremo a extremo cuando aplique (contratos, compatibilidad, errores, reintentos y escenarios de fallo). La validación produce resultados comparables frente a un baseline (antes/después) y adjunta evidencias reproducibles al ticket (informes, métricas, logs y trazas de prueba).
6. **Despliegue controlado y verificación post-despliegue.** El despliegue se realiza de forma escalonada y reversible, con verificación post-cambio basada en telemetría: health-checks, estabilidad, error rate, latencia, backlog, consumo de recursos y ausencia de degradación en dependencias. Para cambios de interoperabilidad se verifica explícitamente el flujo de integración con la versión de contrato activa y la autenticación mTLS. Se aplican criterios de parada y rollback ante degradación detectada.
7. **Evaluación y cierre.** Se evalúa el cumplimiento del objetivo, el impacto real en operación y soporte, y se capturan lecciones aprendidas. Se actualizan, cuando aplique, catálogo de alertas, runbooks, checklists y documentación técnica. El ticket se cierra con evidencia de validación previa, evidencia de despliegue y verificación post-despliegue, y acciones de seguimiento preventivas/evolutivas con responsables asignados.

4.3.2. Criterios de aceptación y rollback

Como criterios mínimos de aceptación se exige ausencia de regresión significativa en operaciones críticas (latencias, tasa de error, estabilidad y colas/backlogs), mantenimiento de baselines de configuración y seguridad, y preservación verificable de trazabilidad. Cuando el cambio afecte a evidencias, metadatos o reprocesados, se requiere además verificación específica de Cadena de Custodia (consistencia, integridad y coherencia temporal). Para cambios que afecten a interoperabilidad se exige validación extremo a extremo de interfaces (contratos, validaciones, compatibilidad entre versiones, códigos de error y autenticación mTLS), incluyendo escenarios de reintento y recuperación.

Todo despliegue debe disponer de un rollback documentado y ejecutable, con criterio de activación, procedimiento asociado y evidencias de retorno a una versión estable, incluyendo una verificación posterior equivalente a la verificación post-despliegue.

Cofinanciado por la Unión Europea
Fondos Europeos

4.4. Monitorización y política de alertas

THOT opera como plataforma central desplegada en un clúster de computación , por lo que la observabilidad y el alertado se diseñan para proporcionar diagnóstico correlacionado, retención suficiente para operación y auditoría, y soporte a procesos correctivos y evolutivos. El stack de observabilidad instrumenta servicios, integraciones y plataforma base con métricas, logs y trazas, permitiendo correlación extremo a extremo y trazabilidad de intervenciones mediante su enlace con ticketing.

Los servicios de THOT se instrumentan de forma homogénea con OpenTelemetry, emitiendo métricas, logs y trazas mediante OTLP. La ingesta se centraliza en un gateway/collector (Grafana Alloy) aplicando normalización, filtrado y muestreo para controlar volumen. Las señales se almacenan y consultan en backends especializados: Prometheus para métricas y evaluación de reglas de alertado, Loki para logs consultables con retención gobernada, y Tempo para trazas distribuidas y diagnóstico correlacionado. Cuando aplique observabilidad de IA (LLMs/RAG u otros), se habilita una herramienta específica (p. ej., Langfuse) para trazabilidad de inferencias, latencias y métricas de calidad. La retención y archivado se soporta con almacenamiento S3-compatible y políticas de backup/DR, y la operación diaria se realiza mediante dashboards por servicio/proceso y exploración cruzada entre métricas, logs y trazas.

El sistema de ticketing se integra con el stack de observabilidad para que cada alerta o incidencia pueda vincularse a evidencias reproducibles (consultas, paneles y trazas), preservando trazabilidad operativa y facilitando auditoría y mejora continua.

4.4.1. Catálogo de alertas y ficha mínima

Sobre la base del stack de observabilidad de THOT, se define un **catálogo de alertas gobernado por versiones**, alineado con los procesos de mantenimiento. El catálogo se estructura en familias de alertas que cubren los principales dominios operativos de Lote 1:

- **Infraestructura y plataforma base**: salud del entorno de ejecución y sus dependencias (nodos, almacenamiento, red, balanceo), incluyendo indicadores de degradación de hardware cuando estén disponibles (latencia de I/O, errores de disco/controladora, enlaces degradados, eventos de hardware y saturación sostenida).
- **Servicios y procesamiento**: disponibilidad y comportamiento de microservicios, APIs, colas y workflows/pipelines; degradación de latencia, error rate, reinicios repetidos, backlogs y fallos recurrentes en tareas críticas.
- **Interoperabilidad e integración**: estado de interfaces con Lote 2 y sistemas externos (conformidad con contratos, compatibilidad de versiones, tasas de rechazo por validación, latencia y errores por integración, evolución de colas de entrada/salida y, cuando aplique, trazabilidad de ACK/NACK o mecanismos equivalentes).
- **Seguridad y cumplimiento**: autenticación/autorización, certificados/PKI (caducidad y fallos de mTLS), accesos privilegiados, cambios de configuración sensibles, integridad de artefactos desplegados y patrones anómalos (picos de errores de autenticación, denegaciones repetidas, escaneos, comportamientos fuera de baseline).

Cada alerta se documenta mediante una ficha mínima auditable, que incluye: nombre y objetivo, condición/umbral y ventana temporal, severidad, objetivo de reacción y resolución, responsable inicial (Operación L1), criterios de escalado (L2/L3 y roles transversales como Seguridad/PKI, Interoperabilidad o CoC cuando proceda), runbook asociado y evidencias enlazables (paneles y consultas reproducibles de métricas/logs/trazas).

Las alertas se implementan como reglas técnicas verificables en el stack de observabilidad (por ejemplo, PromQL para métricas y LogQL para logs; y, cuando aplique, condiciones derivadas de trazas o eventos operativos). Para minimizar falsos positivos y fatiga operativa se aplican mecanismos de histéresis, deduplicación por claves operativas (servicio, versión, integración, entorno), y silencios controlados durante ventanas de mantenimiento planificado.

4.4.2. Gestión operativa, ticketing y escalado

La gestión operativa integra el alertado con la herramienta corporativa de gestión de incidencias y cambios adoptada para THOT. Cada alerta genera un registro trazable (ticket automático o evento convertido a ticket) que incorpora, como mínimo: severidad e impacto, componente afectado (servicio/módulo/activo de infraestructura o integración), marca temporal, identificador de correlación (correlation_id) y enlaces directos a evidencias (dashboards y consultas reproducibles de métricas/logs y, cuando aplique, trazas), además del registro de acciones ejecutadas y su resultado verificado. El catálogo de alertas, umbrales, severidades, responsables y evidencias mínimas se recoge en el Anexo B (Tabla 13, 14)

La política de escalado se organiza en tres niveles. El Nivel 1 (Operación) realiza el triaje, valida el síntoma, aplica medidas de contención y recuperación básica mediante runbooks y actualiza el ticket con evidencias. El Nivel 2 (Especialistas por módulo/plataforma) ejecuta diagnóstico avanzado, analiza telemetría y dependencias, aplica ajustes de configuración y coordina actuaciones con roles transversales cuando corresponda (Seguridad/PKI ante incidentes de identidad o mTLS; Interoperabilidad ante fallos de contratos o validación; Responsable de CoC cuando exista riesgo forense). El Nivel 3 (Ingeniería) aborda correcciones estructurales mediante hotfixes, rollbacks complejos, cambios evolutivos y actuaciones de plataforma que requieran intervención planificada o modificación de arquitectura/configuración.

La coordinación con Lote 2 y sistemas externos se activa cuando la alerta o la incidencia afecta a la capa de integración (contratos, autenticación, validación de mensajes/eventos, colas y consistencia de datos). En estos casos, el ticket incorpora la evidencia de conformidad/no conformidad observada desde THOT y se gestionan pruebas de verificación extremo a extremo para confirmar la recuperación completa del flujo de interoperabilidad, manteniendo trazabilidad de acciones y resultados.

Cofinanciado por la Unión Europea
Fondos Europeos

5. GESTIÓN DEL CICLO DE VIDA DE LA INFRAESTRUCTURA

Este apartado establece el marco operativo para la gestión del ciclo de vida y el mantenimiento de la infraestructura central de THOT, abarcando desde el alta y mantenimiento de activos físicos hasta la operación de la nube privada, la orquestación de contenedores y los servicios base de plataforma necesarios para sostener la operación continua. A efectos de gestión y control, se consideran cuatro capas: (i) infraestructura física (cómputo, almacenamiento y red), (ii) virtualización y nube privada cuando aplique, (iii) orquestación de contenedores, y (iv) servicios base transversales (persistencia, mensajería y eventos, entrada de tráfico, almacenamiento de objetos y soporte a Cadena de Custodia). El objetivo es asegurar disponibilidad, seguridad, trazabilidad y capacidad de evolución controlada, manteniendo coherencia técnica y evidencia auditable en todas las actuaciones.

5.1. Inventario y CMDB de Activos de Infraestructura Central

El inventario/CMDB constituye la fuente única y verificable de activos y elementos de configuración en operación. Su finalidad es habilitar: (i) trazabilidad de configuración y de cambios; (ii) análisis de impacto ante incidencias, actualizaciones o retiradas; (iii) planificación de capacidad, ciclo de vida y obsolescencia; y (iv) generación de evidencias auditables mediante correlación con ticketing e informes de verificación.

La CMDB incluirá, como mínimo, para cada activo o elemento gestionado: identificador único (y número de serie cuando aplique), familia/tipo, modelo, versiones relevantes (sistema, plataforma o aplicaciones gestionadas, cuando corresponda) y firmware cuando aplique, propietario o responsable operativo, unidad/ubicación, estado de servicio (alta, operativo, en mantenimiento, retirado o sustituido), historial de incidencias y cambios, y evidencias asociadas (tickets, partes de intervención, checklists de verificación y resultados de aceptación). Cuando corresponda, se registrará información referencial sobre credenciales o certificados asociados al activo (estado de emisión, rotación y revocación), sin incorporar material sensible.

Desde el punto de vista técnico, la CMDB reflejará relaciones de dependencia entre capas, de modo que sea posible trazar impactos desde la infraestructura hasta el servicio. Como mínimo se registrarán dependencias entre activos físicos, recursos de nube privada, nodos y componentes de orquestación, servicios desplegados, dependencias de datos y almacenamiento de objetos, y contratos de interfaz e integraciones. Para mantener estas relaciones de forma verificable se adoptarán integraciones con: (i) ITSM/ticketing (incidencias, problemas y cambios), (ii) repositorios de configuración y mecanismos IaC/GitOps (estado deseado versionado), (iii) plataforma de orquestación (inventario automatizado vía API para nodos, namespaces, despliegues, versiones y etiquetas), y (iv) plataforma de observabilidad, para enriquecer alertas con el CI afectado y reconciliar inventario frente a estado real.

La herramienta CMDB se seleccionará mediante una decisión de arquitectura, valorando capacidad de modelado de dependencias, APIs de integración, auditoría de cambios y facilidad de reconciliación con la realidad operativa. Hasta la selección final, la herramienta concreta se considera pendiente de confirmación, manteniéndose inalterables los requisitos funcionales descritos.

5.1.1. Ciclo de vida de los elementos de configuración (alta, cambio y retirada)

La gestión del ciclo de vida se articula en tres procesos: alta, cambio y retirada. En todos los casos, cada actuación debe quedar trazada a un ticket, a un registro de auditoría y a la actualización correspondiente en la CMDB.

En el proceso de alta de un activo o servicio se registra el elemento en la CMDB con la información mínima y su responsable operativo. La puesta en servicio exige una verificación técnica con evidencias, que cubra salud,

conectividad y aplicación de baselines de seguridad y configuración, según corresponda. El resultado de la verificación queda registrado mediante ticket y checklist de aceptación.

En el proceso de cambio, la solicitud y el control se gestionan mediante ticket y conforme al circuito de aprobación definido en función del impacto. La ejecución técnica se realiza mediante procedimientos controlados (por ejemplo, IaC/GitOps y pipelines de despliegue), e incluye evaluación previa de impacto (dependencias afectadas, ventana de mantenimiento y viabilidad de reversión), así como validación post-cambio documentada con evidencias reproducibles.

En el proceso de retirada, se documenta la justificación (fin de vida, consolidación, obsolescencia o motivo de seguridad) y se ejecuta un procedimiento que garantice el control de datos y dependencias. Cuando aplique, se contempla backup o migración, verificación de integridad, actualización de relaciones de dependencia, revocación o rotación de credenciales asociadas y actualización documental. El cierre exige ticket de retirada, registro de auditoría y actualización del estado en la CMDB.

5.2. Operación de la nube privada y orquestación

La operación de la nube privada y de la capa de orquestación de THOT se orienta a mantener estabilidad, capacidad, seguridad y continuidad de servicio, minimizando el riesgo asociado a cambios. Cualquier aprovisionamiento o modificación de infraestructura se gestiona mediante un flujo controlado: solicitud documentada, evaluación de impacto (capacidad, seguridad, dependencias y ventana de mantenimiento), aprobación según criticidad, ejecución preferentemente automatizada y verificación posterior basada en evidencias. Todas las actuaciones quedan trazadas mediante ticketing, registros de auditoría y actualización de inventario/CMDB.

5.2.1. Operación de nube privada

THOT opera sobre una nube privada que proporciona el sustrato de ejecución para Kubernetes y para servicios persistentes. Esta capa se gestiona con controles de ciclo de vida que cubren provisión, cambios de capacidad, parcheo y actualizaciones de componentes de control, así como la gestión de incidencias y eventos de degradación. La provisión y el cambio de recursos se ejecutan de forma declarativa mediante plantillas y repositorios versionados (Terraform).

La operación mantiene trazabilidad de stacks/plantillas y sus versiones, gestionando explícitamente los dominios de computación, red, almacenamiento e identidad. El control de capacidad y rendimiento se realiza de forma continua sobre tres planos: cómputo (CPU/RAM y uso por pool/tenant), red (ancho de banda, latencia interna y saturación de enlaces) y almacenamiento (IOPS/latencia, ocupación y estado de replicación). La proyección de capacidad se revisa periódicamente a partir de métricas históricas y tendencias de crecimiento. Los umbrales definidos en la política de alertas activan procedimientos de ampliación de pools, reequilibrado controlado y, cuando proceda, ajustes de cuotas o redistribución de cargas.

El alta de servidores físicos se integra en el mismo circuito de control y evidencias. La puesta en servicio contempla instalación en CPD con alimentación redundante, conectividad de red de gestión y de datos, etiquetado correlacionable con CMDB, aplicación de hardening del sistema operativo, sincronización horaria (NTP) y despliegue de agentes de monitorización y logging. La entrada en operación queda condicionada a una verificación postinstalación con evidencias, que cubra conectividad, baseline de seguridad y rendimiento básico (CPU, disco y red), dejando registro del resultado y aceptación.

Cofinanciado por la Unión Europea
Fondos Europeos

5.2.2. Operación del clúster Kubernetes

Kubernetes se considera un componente crítico y se opera con procedimientos que cubren el plano de control, los nodos de trabajo y los componentes esenciales (add-ons). La topología del clúster se mantiene en alta disponibilidad, con plano de control redundante y etcd en configuración de clúster. La distribución concreta de Kubernetes, cuando aplique, se establece como decisión de arquitectura, manteniendo el mismo marco operativo de ciclo de vida, control de cambios y evidencias.

En conectividad y seguridad, el clúster incorpora una solución CNI que permite segmentación mediante NetworkPoliciesy un Ingress Controller integrado con gestión de certificados (cert-manager) para TLS y, cuando corresponda, mTLS según el perfil de exposición. La persistencia se gestiona mediante CSI, integrando con la capa de almacenamiento de la nube privada o con soluciones distribuidas cuando proceda, garantizando continuidad para los servicios de datos.

La operación del clúster incluye, como mínimo, las prácticas siguientes:

- **Gestión de versiones y actualización planificada.** Se mantiene una política de versiones soportadas y se planifican actualizaciones periódicas, priorizando parches fuera de ciclo ante vulnerabilidades críticas. La actualización se ejecuta por etapas (plano de control y, posteriormente, nodos de trabajo), verificando salud antes y después (API server, etcd, scheduler y controller-manager). Como prerrequisito se realiza copia de seguridad del estado crítico (por ejemplo, snapshots de etcd bajo procedimiento controlado) y se valida compatibilidad de add-ons (CNI, Ingress, CSI, cert-manager). En cambios mayores se exige plan de reversión y una fase de monitorización reforzada tras la intervención.
- **Rotación y mantenimiento de nodos.** El mantenimiento preventivo y correctivo se ejecuta mediante cordon/drain controlado, garantizando reprogramación de cargas y preservación de disponibilidad. Se respetan reglas de distribución (anti-affinity y PDBs cuando aplique) para servicios críticos, evitando puntos únicos de fallo. La reducción de deriva de configuración se apoya en rotación escalonada, incluyendo recreación controlada de instancias cuando proceda.
- **Seguridad y aislamiento en Kubernetes.** Se aplica un baseline de seguridad con segmentación restrictiva por defecto mediante NetworkPolicies (deny-all y allow explícito) y estándares de seguridad de pods (PodSecurityStandards) adecuados al entorno. Las excepciones necesarias quedan documentadas, justificadas y revisadas periódicamente. Se mantiene un proceso de revisión de permisos RBAC para detectar roles excesivamente permisivos y corregir desviaciones, y se controlan los mecanismos de gestión de secretos conforme al enfoque adoptado (por ejemplo, Secrets con cifrado en reposo o integración con un gestor externo).
- **Mantenimiento de add-ons críticos y observabilidad del clúster.** Los componentes esenciales se gestionan como parte del ciclo de vida del clúster, con actualización controlada y trazabilidad. La monitorización integra el stack definido previamente con Grafana.
- **Operación declarativa y trazabilidad.** El estado deseado del clúster y sus componentes se mantiene versionado (IaC/GitOps) y los cambios se ejecutan mediante pipelines controlados, generando evidencias (artefactos, aprobaciones, resultados de pruebas y verificación postcambio). Tras cada intervención, la CMDB y la matriz de versiones se actualizan para reflejar el estado efectivo del clúster y sus dependencias.

5.2.3. Gestión de obsolescencia (EOL/EOS) y control de riesgo

Se mantiene una gestión activa de obsolescencia para componentes de virtualización, sistemas base, Kubernetes, runtimes y add-ons críticos. El proceso incluye detección temprana de deprecaciones y fin de soporte, evaluación de impacto (release notes, compatibilidad, exposición a vulnerabilidades y riesgos operativos), planificación mediante RFC con ventana y pruebas previas en un entorno representativo, ejecución controlada en producción con reversión preparada y verificación posterior basada en evidencias.

Cofinanciado por la Unión Europea
Fondos Europeos

Se establecen alertas tempranas previas al fin de soporte y restricciones operativas para impedir despliegues o ampliaciones sobre versiones próximas a EOL, salvo aprobación expresa. Las excepciones temporales se gestionan con mitigaciones de seguridad documentadas, aceptación de riesgo, plan de regularización y fecha objetivo de cierre.

5.3. Backup, DR y continuidad operativa

La continuidad operativa de THOT se garantiza mediante una estrategia integrada de copias de seguridad, recuperación ante desastres (DR) y procedimientos de restauración verificados, aplicable a infraestructura, servicios base y datos críticos. El objetivo es asegurar que, ante fallos parciales o mayores (corrupción lógica, borrados accidentales, caída de componentes críticos o indisponibilidad de recursos), el servicio pueda recuperarse con integridad y trazabilidad, preservando los registros necesarios para auditoría y, cuando proceda, los elementos asociados a Cadena de Custodia.

El alcance mínimo de protección incluye: (i) datos persistentes de las plataformas de datos (bases de datos y repositorios de metadatos); (ii) almacenamiento de objetos para evidencias, resultados y artefactos operativos; (iii) configuración y estado declarativo de la plataforma (IaC/GitOps y configuración versionada); (iv) componentes críticos del plano de control del clúster Kubernetes cuando aplique (por ejemplo, backups y/o snapshots gestionados del estado que sea necesario preservar según el diseño); y (v) registros de auditoría necesarios para trazabilidad operativa. Las copias se ejecutan de forma automatizada, con cifrado, control de acceso por rol, segregación del repositorio de backup respecto al origen y verificación de integridad.

La recuperabilidad se acredita mediante restauraciones de prueba periódicas y simulacros de DR sobre escenarios definidos (fallo de componente, corrupción o borrado, e indisponibilidad mayor). Cada ejercicio genera evidencias auditables, incluyendo actas de ejecución, logs de restauración, tiempos reales obtenidos y acciones correctivas derivadas. Se considerará conforme cuando existan registros de ejecución de backups, restauraciones verificadas con éxito, runbooks de DR operativos y trazabilidad completa (ticketing/CMDB) de actuaciones y resultados..

5.4. Criterios de aceptación y control de versiones

La aceptación operativa de este apartado se verifica mediante evidencias objetivas y registros auditables. Se considerará conforme cuando se cumplan, como mínimo, los criterios siguientes:

Ámbito	Criterio de aceptación	Evidencia mínima
CMDB	Cobertura completa de activos y servicios en producción con estado, versión, responsable y dependencias mínimas	Informe de reconciliación CMDB vs estado real, con muestreo y discrepancias tratadas
Kubernetes	Actualizaciones y rotaciones ejecutadas sin degradación no controlada, con validación de salud del clúster	Ticket/RFC de cambio + evidencias de salud (métricas/eventos) antes y después

Cofinanciado por la Unión Europea
Fondos Europeos

Cambios	Cambios relevantes con evaluación de impacto, aprobación, reversión definida y validación postcambio	RFC/ticket + checklist de ejecución + resultados de validación y evidencia de rollback si aplica
Auditoría	Accesos administrativos y cambios críticos auditados y correlacionables	Extractos de auditoría + evidencia de correlación con ticket y capacidad de exportación/consulta
Continuidad	Backups ejecutados y restauraciones de prueba conforme a criticidad	Informes de backup/restore + actas de pruebas y tiempos obtenidos
Interoperabilidad	Cambios con impacto en contratos coordinados y validados extremo a extremo	Evidencias de pruebas de integración + registro de coordinación y versión de contrato

***Tabla 6. Criterios de aceptación y evidencias mínimas para la operación y el control de versiones**. Define los umbrales de conformidad por ámbito (CMDB, Kubernetes, cambios, auditoría, continuidad e interoperabilidad) y la evidencia objetiva requerida para verificación y auditoría.*

Se mantendrá una matriz de versiones soportadas para componentes críticos (nube privada, Kubernetes, almacenamiento, servicios base e integraciones), incorporando fechas de fin de soporte cuando existan, revisiones periódicas y cambios planificados de actualización, de forma que la obsolescencia no se gestione de manera reactiva.

6. ESTRATEGIAS DE ACTUALIZACIÓN

6.1. Control declarativo del estado: GitOps

La estrategia de actualización se apoya en un control de configuración declarativo, utilizando repositorios Git como fuente única de verdad del estado deseado de servicios y configuraciones en Kubernetes. Se aplica un enfoque GitOps con reconciliación continua (Argo) para comparar estado real frente a estado declarado, detectar deriva y converger conforme a políticas definidas. Esto proporciona trazabilidad de cambios, consistencia entre entornos y verificación objetiva del estado operativo.

El control de cambios se articula mediante Pull/Merge Requests con revisión y aprobación. Como mínimo interviene el responsable técnico del servicio y, cuando el cambio afecte a seguridad (permisos, cifrado, exposición de interfaces), se requiere validación del rol competente. La promoción entre entornos se realiza de forma secuencial con validaciones obligatorias en cada etapa. El rollback se ejecuta restaurando el último estado estable (tag/commit), activándose ante degradación observada en la verificación postdespliegue o en los indicadores operativos, y quedando vinculado al ticket con evidencias y resultado.

6.2. Gestión de artefactos y cadena de suministro

Los servicios de THOT se distribuyen como contenedores y se publican en un registro privado corporativo (Harbor), que centraliza versionado, metadatos, políticas de retención y cuarentena, control de acceso RBAC y escaneo de seguridad. Cada release queda identificado mediante digest inmutable, garantizando reproducibilidad y trazabilidad entre código, pipeline y artefacto desplegado.

La cadena de suministro se controla con medidas aplicadas en CI/CD y en el punto de admisión de Kubernetes. Tras superar pruebas automatizadas, las imágenes se firman usando estándares como Sigstore/Cosign, vinculando firma y procedencia con el pipeline. En el clúster, se configura verificación obligatoria de firmas (por ejemplo, con políticas OPA/Gatekeeper o Kyverno), rechazando despliegues con imágenes no firmadas, firmas no autorizadas o procedentes de registros no aprobados (p. ej., fuera de Harbor).

El escaneo de vulnerabilidades se realiza de forma sistemática en el pipeline y/o en Harbor, aplicando una política de bloqueo que impide la promoción a producción cuando existan vulnerabilidades críticas/altas sin remediación o mitigación documentada. Para componentes de alta criticidad (p. ej., CoC central, procesamiento, servicios de datos e IA cuando aplique), se generan y conservan SBOMs, así como evidencias de procedencia del build (provenance) para facilitar auditoría y respuesta ante incidentes. Como mínimo, el registro conserva historial de versiones, digests, resultados de escaneo, firmas y metadatos de build; y el clúster conserva eventos de admisión que acreditan aceptación o rechazo.

6.3. Despliegue progresivo, verificación y reversión

El despliegue de actualizaciones se ejecuta de forma controlada y repetible. En CI/CD se construye la imagen, se ejecutan tests (unitarios, integración y regresión), se escanean vulnerabilidades, se firma el artefacto y se publica en Harbor. Posteriormente, el repositorio GitOps se actualiza mediante Pull/Merge Request, y Argo CD aplica el cambio al clúster conforme al estado deseado.

La estrategia de despliegue se selecciona según criticidad del servicio. Como estándar se aplica rolling update, apoyado en readiness/liveness probes y políticas de disponibilidad (PDBs) para mantener continuidad. Para cambios de mayor riesgo o que requieran reversión inmediata se utiliza blue/green condicionado a

disponibilidad de capacidad excedente en el clúster según escenario de dimensionamiento. Para cambios sensibles (interoperabilidad con Lote 2, APIs críticas, pipelines o servicios con impacto en disponibilidad) se aplica canary deployment, liberando progresivamente réplicas o porcentaje de tráfico y avanzando solo si los indicadores permanecen dentro de los umbrales definidos.

La verificación postdespliegue se basa en health checks y smoke tests automatizados, y en el seguimiento de SLIs derivados del stack de observabilidad . Se controlan latencia p95/p99, tasa de error, reinicios, consumo de recursos, saturación y backlogs de procesamiento. Si se incumplen criterios de aceptación o se disparan alertas críticas durante la ventana de observación reforzada, se ejecuta rollback restaurando manifiestos GitOps y versión anterior de imagen (digest), dejando evidencias de retorno a estado estable.

Como evidencia mínima por release se conserva: commit/tag, digest de imagen en Harbor, manifiesto aplicado, resultado de reconciliación de Argo CD, ticket asociado, resultados de verificación y alertas correlacionadas, y registro completo de reversión si aplica.

6.4. Policy-as-Code y cumplimiento en admisión (Kubernetes)

El baseline de seguridad y operación se garantiza mediante un enfoque de Policy-as-Code aplicado en el punto de admisión de Kubernetes. Las políticas se gestionan como código versionado y siguen el mismo flujo de control de cambios (Pull/Merge Request) que el resto de configuraciones. Su aplicación se realiza mediante tecnologías como OPA/Gatekeeper o Kyverno (según decisión de arquitectura), evaluando cada solicitud de creación/modificación antes de su persistencia en el clúster y evitando configuraciones no conformes.

Como mínimo se aplican políticas obligatorias: prohibición de contenedores privilegiados y escalada de privilegios, ejecución como usuario no root, control de capacidades, obligatoriedad de requests/limits para evitar contención, restricción de registros permitidos (Harbor como registro autorizado) y verificación de firmas (Cosign), además de segmentación obligatoria con NetworkPolicies por dominio/namespace. Para trazabilidad operativa se exige etiquetado estándar (propietario, criticidad, versión y referencia al ticket de cambio), facilitando correlación con CMDB y auditoría.

Las excepciones se gestionan de forma formal: requieren justificación técnica, aprobación del rol competente de seguridad, vigencia temporal y revisión periódica, evitando excepciones permanentes no justificadas y asegurando el mantenimiento del baseline a lo largo del ciclo de vida.

6.5. Coordinación de actualizaciones con interoperabilidad (Lote 2)

Las actualizaciones que afectan a interoperabilidad con Lote 2 se gobiernan con un control reforzado para evitar rupturas. Los contratos de integración se versionan y publican como artefactos de referencia mediante OpenAPI (REST) y AsyncAPI (mensajería/eventos), con changelog y trazabilidad de versiones en repositorio. Los cambios compatibles preservan compatibilidad hacia atrás; los cambios incompatibles (“breaking”) se introducen con versión mayor y un periodo de coexistencia planificado, con fechas de deprecación y retirada.

Antes del despliegue en producción se ejecutan pruebas de integración en preproducción, incluyendo validación funcional y de seguridad (autenticación, autorización, validación de payloads y tratamiento de errores). Tras el despliegue, se activa una ventana de monitorización reforzada con métricas específicas por integración: tasa de éxito, latencia, rechazos por validación, errores de autenticación y evolución de colas/backlogs. Ante degradación se ejecuta rollback de contrato y/o servicio conforme al procedimiento, preservando evidencias y trazabilidad.

6.6. Métricas y control operativo de la estrategia de actualización

La eficacia de la estrategia se controla con KPIs obtenidos de CI/CD, GitOps, Harbor, admisión de Kubernetes y ticketing. Como mínimo se monitorizan la tasa de éxito de despliegues, frecuencia de rollback y MTTR de reversión, tiempo de propagación de cambios entre entornos, y número de incidencias asociadas a releases. En control de deriva se mide el drift detectado y el tiempo medio de reconciliación por Argo CD.

En cadena de suministro se controla: porcentaje de cargas en producción con imágenes firmadas, presencia de vulnerabilidades críticas/altas en imágenes desplegadas, tiempo de remediación, y rechazos por incumplimiento de políticas (registro no autorizado, firma no válida, baseline de seguridad). Los resultados se revisan en el gobierno operativo del servicio y se documentan junto con acciones de mejora (ajuste de pruebas, endurecimiento de políticas, refuerzo de observabilidad o cambios en estrategia de despliegue), de forma que sea demostrable la reducción del riesgo operativo asociada al ciclo de actualizaciones.

7. MANTENIMIENTO DE MODELOS DE IA

Este apartado se integra en el mantenimiento evolutivo y en las estrategias de actualización de THOT, dado que los modelos de IA y su pipeline de inferencia se gestionan como artefactos versionados, sujetos a control de cambios, validación reproducible y rollback. A nivel operativo, cualquier evolución de IA se gobierna con el mismo ciclo que el resto de cambios de plataforma: solicitud, análisis de impacto, validación, despliegue controlado, verificación postcambio y cierre con evidencias. La gestión de incidencias y cambios se articula por niveles: **L1** registra el síntoma y evidencia mínima, aplica medidas autorizadas (por ejemplo, conmutación a versión estable del modelo o desactivación controlada de la funcionalidad de IA si existe fallback); **L2** reproduce, analiza impacto (operación, seguridad y trazabilidad), y propone la actuación; **L3** ejecuta la evolución (ajustes de serving, optimización/compilación, reentrenamiento o afinación y liberación de nuevas versiones), manteniendo control de versiones y verificación completa.

7.1. Alcance y principio forense: reproducibilidad y trazabilidad

Los resultados de IA pueden formar parte del proceso de análisis y de los productos generados por THOT, por lo que debe ser posible reconstruir qué versión exacta de modelo produjo un resultado y bajo qué condiciones de ejecución. Para cada inferencia relevante se registran metadatos mínimos y verificables, correlacionables con telemetría y auditoría: **model_id**, **model_version**, **artifact_digest** (hash/digest del artefacto), **serving_version** (versión del microservicio de inferencia), **pipeline_version** (pre/post-proceso y parámetros), **timestamp**, **scene_id/case_id** y **evidence_id** cuando aplique, además de **execution_id/correlation_id** para enlazar con logs y trazas. Todos estos metadatos se quedan registrados, con conservación acorde a la política de auditoría y con capacidad de exportación para verificación.

7.2. Inventario, registro de modelos y repositorios (Model Registry)

Cada modelo se gestiona bajo un Model Registry como fuente de verdad para despliegue, auditoría y reversión. El registro mantiene, por versión: identificador de caso de uso, versión semántica, huella criptográfica, formato del artefacto (por ejemplo ONNX cuando aplique), dependencias de runtime, parámetros de inferencia, versión de pre/post-procesado, requisitos de ejecución y estado de aprobación (candidate / validated / pilot / approved / deprecated / retired).

Como implementación MLOps, se utiliza típicamente MLflow Model Registry y un repositorio de artefactos en almacenamiento S3-compatible para binarios y metadatos. Para garantizar integridad, las versiones del modelo se publican con digest inmutable y controles de verificación (firma/huella) alineados con la cadena de suministro. Cada release incorpora una “model card” técnica con propósito, evidencias de evaluación, limitaciones y riesgos conocidos, y el informe de validación asociado.

7.3. Ciclo de vida MLOps: de candidate a approved

El ciclo de vida de modelos se gobierna por estados de calidad, de forma que la versión “en producción” sea siempre **Approved** y reproducible:

- **Candidate**: versión registrada; no desplegable.
- **Validated**: validación reproducible en entorno controlado (calidad, robustez y rendimiento).
- **Pilot**: despliegue limitado con monitorización reforzada (canary/porcentaje de tráfico o por casos controlados).
- **Approved**: autorizada para despliegue general.

- **Deprecated/Retired**: desaconsejada o retirada (manteniendo trazabilidad histórica).

Cada transición exige evidencias: comparación frente a baseline, resultados en el entorno objetivo (Kubernetes/serving real), checklist de trazabilidad de metadatos y un informe de impacto sobre operación, seguridad y trazabilidad forense.

7.4. Disparadores de reentrenamiento/actualización

Las actualizaciones se disparan por evidencias objetivas provenientes de operación y validación, incluyendo:

- **Regresión funcional**: degradación sostenida de métricas del caso de uso o aumento de errores operativos atribuibles al modelo.
- **Deriva (drift)**: cambios en distribución de entradas o comportamiento que afecten a resultados.
- **Regresión de rendimiento**: incremento de latencia p95/p99, caída de throughput, saturación de recursos o inestabilidad del serving.
- **Cambios operativos**: nuevas fuentes de datos, ajustes en pipelines, o cambios en interfaces/datos de entrada.
- **Seguridad y robustez**: necesidad de mitigaciones, endurecimiento del pipeline o corrección de vulnerabilidades asociadas (dependencias del runtime/serving).

Estas señales se obtienen del stack de observabilidad (métricas, logs y trazas), y la decisión se formaliza como solicitud de cambio con responsable técnico y evidencia enlazada.

7.5. Validación previa a despliegue

Antes de autorizar una nueva versión se ejecuta una validación reproducible en entorno controlado, comparada con la versión vigente. La validación cubre:

- **Calidad/efectividad**: métricas por caso de uso (precisión/recall/F1 u otras), análisis de errores y no degradación frente a baseline.
- **Rendimiento**: latencia p95, throughput, consumo de CPU/memoria y estabilidad del servicio de inferencia.
- **Robustez operacional**: tolerancia a inputs degradados, ausencia de crashes, compatibilidad con pre/post-proceso y consistencia de metadatos de trazabilidad.

Criterios mínimos de aceptación (ajustables por caso de uso): no degradación o mejora demostrable, latencia p95 dentro del objetivo, sin incremento de errores operativos, trazabilidad completa (model/pipeline/serving) y rollback verificado.

7.6. Empaquetado y despliegue

En THOT, el serving se despliega como microservicio contenerizado en Kubernetes, gestionado bajo control de cambios y despliegue progresivo. Para serving de alto rendimiento se contempla **NVIDIA Triton Inference Server** cuando aplique, o servicios contenerizados con APIs de inferencia **FastAPI**/**gRPC**, según el caso de uso. El despliegue de una nueva versión se realiza mediante GitOps y estrategias de rollout. El rollback se ejecuta restaurando manifiestos GitOps y seleccionando una versión Approved anterior (model_version + artifact_digest), verificando salud del serving y estabilidad post-reversión. Cuando exista fallback funcional, se permite desactivar temporalmente la capacidad de IA o degradar el servicio a una ruta alternativa, manteniendo trazabilidad y sin alterar evidencias.

Cofinanciado por la Unión Europea
Fondos Europeos

7.7. Trazabilidad, monitorización y alertas específicas

La supervisión postdespliegue cubre latencia, tasa de error, saturación de recursos, estabilidad del serving y señales proxy de calidad y drift. La instrumentación se integra con el stack de observabilidad . Para trazabilidad específica de inferencias, especialmente en componentes tipo LLM/agentic cuando aplique, se integra Langfuse como capa de observabilidad de inferencias (latencias, errores, metadatos de entrada/salida) correlacionada con execution_id/correlation_id.

Se definen alertas específicas de IA: regresión de latencia p95/p99, aumento de errores de inferencia, inestabilidad del serving (reinicios/crashloop), señales de drift sostenido y degradación de throughput. Superado un umbral, se genera ticket y se activa el circuito L1/L2/L3, con medidas de contención y decisión de rollback o evolución.

7.8. Gobierno del cambio y evidencias (auditoría, seguridad y sostenibilidad)

Toda actualización de modelo se gestiona como cambio controlado: solicitud, impacto, aprobación, validación, despliegue y verificación. Se mantiene un registro auditable de quién aprobó, cuándo, qué versión se instaló, qué manifiestos se aplicaron y qué evidencias respaldan la validación y la verificación postdespliegue. Este enfoque reduce recurrencia y OPEX (rollbacks rápidos y decisiones basadas en evidencia), mejora sostenibilidad técnica (versionado, deprecación y retirada controlada) y preserva coherencia con requisitos de trazabilidad y Cadena de Custodia cuando los resultados de IA se incorporan al flujo de análisis.

Cofinanciado por la Unión Europea
Fondos Europeos

8. GESTIÓN DE CERTIFICADOS Y PKIS

La Infraestructura de Clave Pública (PKI) de THOT proporciona identidades criptográficas para microservicios, componentes de plataforma e integraciones externas cuando aplique, habilitando TLS y mTLS para asegurar autenticación mutua y cifrado de las comunicaciones. Esta capa es crítica tanto para la interoperabilidad segura entre THOT (Lote 1) y la solución del Lote 2 como para la protección de las comunicaciones internas entre microservicios y servicios base en la plataforma central.

El plan de operación cubre el ciclo de vida completo de certificados y credenciales: emisión en el alta controlada, distribución e inyección en los puntos de consumo (pods, ingress/gateway, servicios base), control de vigencia y uso, rotación programada y revocación inmediata ante compromiso, uso no autorizado o baja/sustitución planificada del servicio o activo asociado. Todo el ciclo queda trazado mediante ticketing, auditoría y actualización de inventario/CMDB. El objetivo operativo es evitar credenciales compartidas, garantizar trazabilidad de identidad (qué servicio se conecta y con qué certificado) y habilitar contención rápida ante incidentes, preservando continuidad de servicio.

8.1. Arquitectura de PKI

En THOT la PKI actúa como autoridad de emisión y control, definiendo políticas de validez/rotación, usos de clave y mecanismos de revocación. La arquitectura se apoya en una **CA raíz** bajo control corporativo (manteniéndose fuera de operación diaria) y una o varias **CA intermedias** operativas para emisión y rotación de certificados de servicio. Se contemplan dos modalidades, seleccionadas como decisión de arquitectura según el marco corporativo:

- **Integración con CA corporativa** (Policía Nacional), utilizando una CA intermedia específica para THOT y políticas de emisión coherentes con los requisitos de seguridad corporativos. Esta opción facilita confianza preestablecida, alineamiento normativo y operación homogénea.
- **CA dedicada para THOT**, basada en una solución integrable con Kubernetes (por ejemplo, **step-ca**), manteniendo la CA raíz protegida y la CA intermedia para emisión operativa. Cuando aplique, la protección de claves de CA se apoya en **HSM** o mecanismos equivalentes de custodia.

En ambos casos, THOT mantiene un **modelo de confianza explícito**: bundles de confianza (trust bundles), cadenas de certificación y políticas TLS/mTLS para servicios internos e integraciones.

8.2. Automatización de certificados en Kubernetes

Para workloads desplegados en Kubernetes, THOT utiliza cert-manager para automatizar solicitud, emisión, renovación y distribución de certificados, reduciendo riesgo de caducidad y carga operativa. El flujo se define de forma declarativa: el servicio publica su necesidad (por ejemplo, Certificate/Issuer/ClusterIssuer), cert-manager genera la CSR, solicita emisión a la CA conforme a la política activa y publica el certificado y clave privada en un Secret gestionado para su consumo por el pod o por el componente de entrada (Ingress Controller o API Gateway).

La renovación se ejecuta de forma preventiva y automática. La operación valida la renovación con evidencias técnicas: comprobación de vigencia, verificación de cadena de confianza y pruebas de conectividad TLS/mTLS (handshake) en endpoints afectados. Ante fallos de emisión/renovación, se abre ticket y se ejecuta el ciclo correctivo (diagnóstico de CA, permisos RBAC, límites de tasa, políticas de emisión), dejando registro completo de causa, acción y verificación.

8.3. Protección de claves y gestión de secretos

La gestión de claves privadas y secretos se coordina con el baseline de seguridad del clúster. Los secretos de certificados se protegen mediante control de accesos (RBAC), segregación por namespaces y políticas de admisión (Policy-as-Code) que impiden configuraciones inseguras (por ejemplo, exposición indebida de Secrets o uso de endpoints no cifrados cuando se exige mTLS). Cuando el diseño corporativo lo permita, THOT puede integrar un gestor de secretos (HashiCorp Vault) para reforzar custodia, rotación y auditoría de material sensible, manteniendo el mismo modelo de trazabilidad.

La exposición segura de interfaces se integra con el Ingress/API Gateway, aplicando TLS/mTLS en los puntos de entrada y, cuando corresponda, mTLS extremo a extremo entre servicios. Las rotaciones y revocaciones se alinean con la gestión del cambio: se planifican ventanas cuando afecten a integraciones, se mantiene compatibilidad durante la transición (por ejemplo, bundles con CA antigua y nueva en coexistencia) y se verifican pruebas post-rotación antes de cerrar el cambio.

La respuesta ante incidentes se articula como un circuito operativo cerrado: detección o notificación, apertura de ticket, identificación del alcance (servicios/identidades afectadas), revocación inmediata (CRL/OCSP según aplique), reprovisión de credenciales y verificación de contención mediante evidencia de fallo controlado de autenticación/handshake para identidades revocadas. En integraciones externas o activos con credenciales propias, la actuación sigue el mismo patrón: revocación, reprovisión, actualización de inventario/CMDB y cierre auditable con evidencias.

La operación mantiene evidencias verificables extremo a extremo: inventario de certificados activos y su asignación a servicios/roles, registros de emisión/renovación, alertas de caducidad, eventos de revocación, pruebas de conectividad post-rotación y correlación con tickets y CIs. Con ello, THOT reduce el impacto de incidentes, mejora sostenibilidad operativa y consolida un mantenimiento preventivo basado en rotación programada, control de caducidad y detección de uso anómalo, con trazabilidad y auditoría demostrables.

8.3.1. Capa Zero Trust y segmentación de acceso a servicios

Además del uso de TLS/mTLS basado en PKI, THOT puede incorporar un enfoque de arquitectura Zero Trust para el acceso a servicios, orientado a minimizar superficie de exposición y reforzar la segmentación. En este modelo, el acceso se concede de forma explícita por identidad y política (principio de mínimo privilegio), evitando dependencias de confianza implícita por red o ubicación.

Como tecnología habilitadora se contempla el uso de OpenZiti como capa ZTNA/overlay de conectividad basada en servicios. OpenZiti permite publicar servicios internos sin exponer puertos o subredes, estableciendo canales cifrados y autenticados por identidad, y aplicando políticas de autorización por servicio. Esta capa puede emplearse para accesos administrativos controlados, integraciones entre dominios, y para limitar el acceso a microservicios sensibles, complementando los controles nativos del clúster (NetworkPolicies, RBAC y controles de admisión).

La integración se apoya en identidades gestionadas y auditables, alineadas con la PKI descrita: emisión y rotación de credenciales, revocación ante incidente y evidencia verificable de accesos. La adopción de esta capa se gobierna como cambio controlado, con pruebas de interoperabilidad y continuidad para asegurar que no introduce dependencias operativas no deseadas.

9. SLAS Y KPIS CON VALORES OBJETIVO PRELIMINARES

Este apartado establece, con carácter preliminar, los **SLAs (Acuerdos de Nivel de Servicio)** y **KPIs (Indicadores Clave de Desempeño)** para evaluar la **disponibilidad**, el **rendimiento**, la **capacidad**, la **calidad** y la **sostenibilidad** de THOT en operación. El foco se sitúa en la infraestructura de CPD/nube privada y en los servicios centrales desplegados como microservicios (espacio de datos, procesamiento, IA/ML, coordinación, alertas, Cadena de Custodia, comunicaciones e interoperabilidad con Lote 2 y con sistemas externos aplicables).

Los valores objetivo son iniciales y se refinarán con datos reales en pruebas y pilotos, manteniendo un método de medición estable, automatizable y trazable (métricas/alertas en Prometheus, logs en Loki y trazas en Tempo, con paneles y registros de ticketing).

9.1. SLAs operativos (valores objetivo preliminares)

Los SLAs se formulan como compromisos **medibles**, con ventana temporal, umbral y método de verificación. Los siguientes valores se aplican como referencia inicial y se consolidan tras establecer baseline operacional:

SLA	Definición	Objetivo preliminar	Método de medición
Disponibilidad de servicios críticos	% de tiempo en que los servicios críticos (espacio de datos, procesamiento, IA, coordinación, alertas, CoC, APIs de interoperabilidad) están operativos y responden a comprobaciones de salud	≥ 95% mensual	Sondas de salud (health checks) + telemetría de disponibilidad por servicio + logs operativos del API Gateway/Ingress para correlación de errores
Disponibilidad de APIs de interoperabilidad (Lote 2, sistemas policiales)	% de tiempo en que los endpoints de interoperabilidad aceptan solicitudes válidas	≥ 99,0% mensual	Métricas y logs por endpoint en API Gateway/Ingress + sondas sintéticas (si aplica)
Indisponibilidad máxima por incidencia crítica	Tiempo máximo de caída tolerable por evento de severidad crítica (pérdida total de un servicio crítico)	≤ 120 min por evento	Marca temporal de alerta crítica + registro de ticket (ack, resolución, cierre) + verificación post-recuperación mediante health checks
Tiempo de reacción ante incidencia crítica	Tiempo desde la generación de la alerta crítica hasta la primera actuación registrada (ack, ticket, inicio de diagnóstico)	≤ 30 min	Timestamp de alerta + timestamp de primera actuación en ticketing/ITSM (ack/ticket/nota inicial)
MTTR (Mean Time To Repair)	Tiempo medio de recuperación de incidencias críticas (restauración verificada del servicio afectado)	≤ 8 horas	Tickets de severidad crítica (apertura→cierre) + evidencias

— Severidad crítica			de recuperación (servicio saludable y validación funcional)
Éxito de actualizaciones controladas	% de despliegues sin rollback ni incidencia crítica en ventana de observación (24 h post-despliegue)	≥ 98% por oleada	Registros de despliegue (versionado y sincronización GitOps) + correlación con tickets críticos y eventos de rollback
Tasa de éxito de sincronización con Lote 2	% de mensajes/eventos recibidos del Lote 2 procesados correctamente (validación, persistencia y acuse)	≥ 98,0% semanal	Telemetría del servicio de comunicaciones (acuses, errores de validación/persistencia) + trazas/logs de procesamiento
Disponibilidad del servicio de Cadena de Custodia	% de tiempo en que CoC acepta registros forenses y permite consulta de cadena completa	≥ 95% semanal	Sondas de salud del servicio CoC + telemetría de disponibilidad y errores funcionales
Tiempo de respuesta ante solicitud de análisis IA	Tiempo desde solicitud de análisis IA hasta inicio de ejecución (cola + asignación de recursos)	p95 ≤ 5 min	Tiempos de cola y coordinación (timestamp solicitud→inicio) + telemetría del orquestador de análisis
Precisión de modelos IA en producción	Precisión de modelos críticos en producción respecto al baseline validado	≥ 95% del baseline	Métricas de calidad del modelo (p. ej., F1/AUC/accuracy según caso) + comparación periódica con baseline en registro de modelos

Tabla 7. SLAs operativos de THOT (valores objetivo preliminares). *Define los compromisos medibles por servicio, con umbrales y ventana temporal, e identifica el método de verificación (telemetría, logs y ticketing) para consolidar el baseline operacional y auditar el cumplimiento.*

9.2. KPIs técnicos (valores objetivo preliminares)

Los KPIs complementan los SLAs para anticipar degradaciones antes de que generen incidencias. Se calculan por servicio/namespace y se revisan en comités operativos de forma periódica:

KPI	Ámbito	Objetivo preliminar	Método de medición
Uso sostenido de CPU	Infraestructura CPD / clústeres Kubernetes	p95 < 85% en ventana operativa (24 h)	Telemetría de CPU por nodo y por servicio (cuadros de mando de clúster)
Uso sostenido de GPU (IA/ML)	Nodos con GPU para procesamiento IA/ML	p95 < 90% en ventanas de procesamiento	Telemetría de GPU por nodo y por carga IA/ML (utilización y memoria)

Cofinanciado por la Unión Europea
Fondos Europeos

Presión de memoria / OOM	Clústeres Kubernetes / pods críticos	OOM ≤ 1 evento/semana por namespace crítico; p95 < 85% uso de memoria	Telemetría de memoria por pod/servicio + eventos OOM del orquestador
Almacenamiento disponible (BBDD / data lake / objeto)	Almacenamiento persistente (PVC/volúmenes)	≥ 15% libre (umbral preventivo)	Telemetría de capacidad/ocupación por volumen y servicio + alertas por umbral
Longitud de colas y backlog	Mensajería/eventos y colas de análisis IA	Backlog p95 estable (sin crecimiento sostenido > 1 h); drenaje ≤ 30 min tras pico	Telemetría de colas (lag/backlog) + tiempo de drenaje tras picos
Tasa de errores 5xx por API	APIs REST/gRPC (consulta, análisis, interoperabilidad)	< 0,5% de solicitudes	Métricas y logs del API Gateway/Ingress por endpoint (ratio de respuestas 5xx)
Tasa de errores 4xx por API	APIs REST/gRPC expuestas	Monitorización por patrón/anomalía (sin umbral fijo; detección de picos)	Métricas y logs del API Gateway/Ingress (tendencias 4xx, picos y correlación con cambios)
Latencia de consultas críticas (espacio de datos)	Consultas a BBDD / motores de consulta	p95 ≤ 200 ms	Telemetría de latencia por tipo de consulta y servicio (p95)
Tiempo de procesamiento de pipelines críticos	Pipelines de transformación/enriquecimiento	p95 dentro de objetivo por tipo de pipeline (baseline en pruebas)	Telemetría de duración por pipeline/etapa + comparación contra baseline validado
Tasa de reintentos de sincronización con Lote 2	Servicio de comunicaciones / interoperabilidad	< 2% de operaciones con reintentos persistentes (> 3 intentos)	Logs operativos de reintentos + acuses/confirmaciones y ratio de reintento persistente
Drift de configuración (GitOps)	Microservicios/infraestructura (conformidad)	≤ 2% recursos fuera de estado deseado (semanal)	Informes de conformidad GitOps (recursos “out-of-sync”) + eventos de reconciliación

Cobertura de tests automatizados	Código de servicios críticos (CoC, IA, procesamiento, interoperabilidad)	≥ 80% cobertura de líneas	Informes de CI/CD de cobertura y evolución por servicio (umbral de calidad)
Tasa de despliegues exitosos (CI/CD)	Pipeline de despliegue (build, test, deploy)	≥ 95% pipelines completados sin fallo	Métricas del CI/CD (ratio éxito/fallo) + causas principales y tendencia

Tabla 8. KPIs técnicos de THOT (valores objetivo preliminares). *Recoge los indicadores de salud y capacidad de la plataforma (infraestructura, servicios y procesos) para anticipar degradaciones y orientar acciones preventivas, con medición basada en observabilidad y revisiones periódicas en comités operativos.*

9.3. KPIs de calidad y experiencia de usuario (valores objetivo preliminares)

Estos indicadores permiten verificar usabilidad, eficacia operativa y adopción de la plataforma central por parte de analistas, gestores y auditores.

KPI	Objetivo preliminar	Método de medición
Tiempo de respuesta en pantallas críticas de UI	p95 ≤ 2 s en flujos principales (consulta de caso, solicitud de análisis, visualización de resultados, consulta de CoC)	Telemetría de experiencia de usuario (RUM) y/o pruebas sintéticas; correlación con latencia de APIs en gateway/servicios
Éxito de operaciones críticas de usuario	≥ 98% sin error/bloqueo (consulta, análisis, registro CoC, generación de informe)	Telemetría de eventos de aplicación por operación (éxitos/errores) y tendencia temporal
Tasa de incidencias atribuibles a uso/procedimiento vs sistema	Tendencia descendente (baseline en piloto; objetivo: < 20% incidencias por UX/procedimiento)	Clasificación de tickets por causa raíz (post-mortem) y métricas de recurrencia
Finalización de formación por rol	≥ 85% del personal objetivo completa formación en THOT (módulo formativo con evaluación)	Registros del sistema de formación (módulos completados y evaluaciones aprobadas) por rol/unidad
Tasa de adopción de funcionalidades clave	≥ 70% usuarios activos utilizan funcionalidades clave en el primer mes post-formación	Telemetría de uso por funcionalidad (eventos) y usuarios activos; análisis por cohorte post-formación
Satisfacción de usuario (NPS/CSAT)	Baseline en piloto; objetivo: NPS ≥ +20 y/o CSAT ≥ 4/5 (a consolidar)	Encuestas periódicas (NPS/CSAT) en herramienta corporativa, segmentadas por rol, unidad y periodo

Tabla 9. KPIs de calidad y experiencia de usuario (valores objetivo preliminares). *Establece métricas de rendimiento percibido, éxito de operaciones, formación y adopción, con el fin de evaluar eficacia operativa y mejorar la usabilidad mediante evidencias objetivas y seguimiento por rol/unidad.*

Cofinanciado por la Unión Europea
Fondos Europeos

9.4. KPIs ambientales (valores objetivo preliminares)

Los KPIs ambientales se establecen como **línea base** y seguimiento de mejora, condicionados a la disponibilidad de datos del CPD y de la instrumentación de consumo:

KPI ambiental	Objetivo preliminar (realista)	Método de medición
% de energía renovable del CPD/proveedor	Reportado por proveedor (informativo; sin umbral exigible en Fase I)	Evidencia documental del proveedor (certificación y/o informe anual de energía; p. ej., ISO 50001 o auditoría energética)
PUE (Power Usage Effectiveness) del CPD	Reportado/auditado por proveedor (informativo; referencia: PUE ≤ 1,5 en CPD moderno)	Informe de auditoría del CPD o declaración formal del proveedor con metodología de cálculo
Consumo energético estimado por transacción/consulta	Baseline en piloto; mejora 5–10% entre versiones tras optimizaciones	Estimación agregada (kWh reportado del CPD / nº de transacciones) y/o proxy operativo (CPU/GPU-hours por operación) cuando la métrica directa no esté disponible
Eficiencia de procesamiento (tiempo/energía por tarea IA/ML)	Mejora 5–10% (tiempo o energía estimada por tarea) entre versiones de modelos/infraestructura	Telemetría de duración por tarea IA/ML + estimación de consumo por recursos (CPU/GPU) y comparación contra baseline
Vida útil de servidores/equipamiento crítico	≥ 48 meses (referencia inicial para hardware CPD)	Inventario/CMDB: fechas de alta y baja, causas de retirada y evidencias de fin de soporte/obsolescencia
Tasa de reemplazo anual de equipamiento	≤ 15% (referencia inicial)	Inventario/CMDB: bajas y sustituciones anuales / total de activos, con segmentación por familia de activo
Gestión autorizada de RAEE (Residuos de Aparatos Eléctricos y Electrónicos)	100% de bajas con retirada por gestor autorizado y trazabilidad documental	Actas de retirada/reciclaje y justificantes del gestor + actualización de inventario/CMDB y cierre de ticket de retirada

Tabla 10. KPIs ambientales (valores objetivo preliminares). *Define los indicadores de sostenibilidad asociados al CPD y a la explotación (energía, eficiencia y ciclo de vida de equipamiento), condicionados a la disponibilidad de datos del proveedor y a la instrumentación de consumo.*

9.5. KPIs específicos de Cadena de Custodia

Por la criticidad forense de THOT, se incorporan KPIs específicos para CoC, complementando los SLAs de disponibilidad:

KPI CoC	Objetivo preliminar	Método de verificación
Integridad verificada de evidencias	100%	Validaciones de hash/manifest + registros de verificación
Completitud de secuencia CoC	≥ 99,8%	Reglas automáticas de secuencia y validadores
Anomalías CoC	< 0,05%	Detectores automáticos + alertas
Tiempo de detección de no conformidad	Crítica ≤ 5 min; Alta ≤ 15 min	Timestamp alerta → ticket
Trazabilidad de acciones de soporte	100%	Auditoría de acciones correlacionadas con eventos CoC

Tabla 11. KPIs específicos de Cadena de Custodia (CoC). *Complementa los SLAs de disponibilidad con métricas forenses de integridad, completitud y trazabilidad de acciones, asegurando verificabilidad y auditoría de extremo a extremo sobre evidencias y eventos CoC.*

9.6. Revisión y ajuste de objetivos

Los SLAs y KPIs se revisan con cadencia definida (recomendación operativa: trimestral en fase piloto y semestral en operación estabilizada), basándose en datos observados, incidencias, evolución de amenazas y resultados de pruebas de interoperabilidad. Cualquier ajuste se documenta mediante acta de revisión y, cuando afecte a decisiones técnicas, mediante registro de decisión (ADR), dejando constancia de la fecha de entrada en vigor y del impacto operativo.

Cofinanciado por la Unión Europea
Fondos Europeos

10. ESTRATEGIA DE ESCALABILIDAD

La estrategia se apoya en dos ejes complementarios: (i) las capacidades de escalado del propio clúster Kubernetes y su posibilidad de ampliación mediante incorporación de nodos y recursos, y (ii) una arquitectura de exposición y acceso orientada a alta concurrencia, basada en API services con FastAPI y un API Gateway (por ejemplo, Kong) que desacopla entrada de tráfico y aplica control de flujo, autenticación y observabilidad.

10.1. Escalabilidad por infraestructura y clúster Kubernetes

THOT hereda las capacidades de escalabilidad propias de Kubernetes. Los microservicios pueden replicarse horizontalmente, distribuirse entre nodos y aislar cargas críticas mediante requests/limits, afinidades/anti-afinidades y políticas de scheduling. La plataforma puede crecer de forma incremental ampliando la capacidad del clúster mediante incorporación de nodos (compute) y expansión de recursos asociados (almacenamiento y red), evitando rediseños estructurales.

La escalabilidad se gestiona por servicio y por cuello de botella real: se dimensionan y escalan de forma diferenciada la capa de entrada (gateway), los microservicios, los componentes de procesamiento, los servicios de datos y, cuando aplique, los servicios de IA. Para absorción dinámica de carga se emplea HPA (Horizontal Pod Autoscaler) sobre métricas operativas (CPU/memoria y métricas de aplicación cuando estén expuestas), y se contempla VPA (Vertical Pod Autoscaler) cuando proceda para ajustar recursos de forma controlada. La resiliencia y la capacidad se refuerzan con balanceo/ingress, estrategias de despliegue controlado y observabilidad centralizada, lo que permite aumentar réplicas, recursos o nodos del clúster manteniendo continuidad de servicio.

10.2. Escalabilidad por concurrencia y capa de acceso (FastAPI + Kong)

Además de escalar “por infraestructura”, THOT se diseña para sostener alta concurrencia en la capa de acceso y API. Los servicios expuestos se implementan como microservicios con FastAPI, desplegados en Kubernetes con workers configurados para concurrencia y estabilidad, y fronted por un API Gateway (Kong) como punto único de entrada. Este patrón permite desacoplar el crecimiento de usuarios concurrentes del resto de componentes internos y aplicar políticas coherentes de control de tráfico.

Kong actúa como capa de gobernanza de API, aplicando mecanismos esenciales para escalado controlado: rate limiting, cuotas por consumidor/rol, autenticación y autorización (incluyendo mTLS cuando aplique), validación básica de entrada, enrutado por servicio/versión y observabilidad de peticiones. En paralelo, los microservicios FastAPI se escalan horizontalmente (réplicas) para absorber concurrencia, mientras que el gateway permite mantener estabilidad ante picos (protección frente a tormentas de peticiones) y priorizar tráfico crítico cuando proceda. Esta combinación habilita crecimiento de usuarios sin degradación no controlada y facilita la gestión operativa (diagnóstico por ruta/consumidor y trazabilidad de errores).

10.3. Escenarios de crecimiento

Los escenarios de crecimiento previstos incluyen: aumento de usuarios simultáneos (investigadores, analistas y perfiles de supervisión), incremento de volumen de evidencias (imágenes, vídeo, reconstrucciones y derivados), mayor complejidad de analítica y modelos de IA, y mayores exigencias de retención, auditoría e interoperabilidad.

La respuesta se articula con una doble palanca: (i) dimensionamiento progresivo del clúster (nodos, recursos por pool, almacenamiento y throughput de red) y escalado de servicios mediante HPA/VPA, y (ii) refuerzo de la capa de entrada para absorber concurrencia y picos, aplicando control de flujo, cuotas y priorización. Cuando los cuellos de botella se sitúan en servicios de datos o procesamiento, el escalado se completa con separación de cargas, optimización de pipelines y ajustes de persistencia (IOPS/latencia y particionado lógico cuando aplique), siempre con evidencia operativa que justifique el dimensionamiento.

10.4. Riesgos de escalado y mitigaciones

En THOT, los riesgos principales se concentran en: (i) cuellos de botella por servicio (gateway, servicios de datos, pipelines o IA), (ii) saturación de recursos compartidos (almacenamiento y red), y (iii) degradación por picos de concurrencia o patrones de uso no previstos. Se mitigan mediante observabilidad (detección temprana de saturación), escalado horizontal selectivo, aislamiento de cargas críticas con políticas de scheduling, y control de tráfico en el gateway (rate limiting, cuotas y priorización).

Adicionalmente, la expansión del clúster mediante incorporación de nodos y la planificación de capacidad basada en tendencias permiten evitar escalados reactivos. Los cambios de configuración asociados a escalabilidad (límites, réplicas, políticas de gateway y umbrales de autoscaling) se gobiernan como cambios controlados y verificables, asegurando trazabilidad y estabilidad a lo largo del ciclo de vida.

Cofinanciado por la Unión Europea
Fondos Europeos

11. SOSTENIBILIDAD TÉCNICA Y ECONÓMICA A LARGO PLAZO

Este apartado define un marco práctico, verificable y auditable para estimar, justificar y controlar el coste recurrente de operación y evolución (OPEX) de THOT (Lote 1), y para asegurar su sostenibilidad técnica durante todo el ciclo de vida: operación, mantenimiento, soporte, actualizaciones y renovación planificada. El modelo de sostenibilidad se centra en los "drivers" que realmente gobiernan el coste y el riesgo: usuarios concurrentes, volumen de casos y evidencias, retención y auditoría, consumo de cómputo/almacenamiento/red, frecuencia de cambios, y esfuerzo de soporte L1/L2/L3.

11.1. CAPEX vs OPEX y supuestos de cálculo (drivers)

A efectos de este Plan, **CAPEX** agrupa la adquisición y puesta en servicio inicial de la infraestructura y licencias necesarias para THOT (servidores, almacenamiento, red, plataforma de virtualización/nube privada si aplica, clúster Kubernetes y servicios base), incluyendo alta en inventario/CMDB, hardening y validación de aceptación. **OPEX** agrupa los costes recurrentes para sostener el servicio: operación/monitorización, soporte (L1/L2/L3), gestión de cambios y releases, capacidad (ampliaciones), backup/DR, seguridad (PKI, hardening, respuesta a incidentes), explotación de herramientas (observabilidad, registry, CI/CD, GitOps, ticketing) y renovación planificada por obsolescencia (EOL/EOS) o fallo.

El OPEX se modela con granularidad anual y se parametriza por drivers medibles y recalibrables con evidencia operativa:

- **Usuarios y concurrencia**: número de usuarios activos, picos de concurrencia, patrones de consulta/descarga.
- **Carga de trabajo**: número de casos/escenas, volumen medio de evidencias por caso, y volumen total anual.
- **Plataforma y datos**: consumo de CPU/RAM (y GPU si aplica), throughput de red, IOPS/latencia de almacenamiento, tamaño del data lake/object storage y crecimiento de bases de datos.
- **Exigencias de trazabilidad/auditoría**: retención de logs/trazas, retención de auditoría y requisitos de exportación/verificación.
- **Cambio**: frecuencia de releases, ratio de incidencias recurrentes, tasa de rollback y esfuerzo de validación.

Estas hipótesis no se consideran estáticas: se recalibran con telemetría, tickets, métricas de explotación y resultados de cambios, de modo que el coste sea gobernable y defendible.

11.2. Modelo OPEX por partidas

El OPEX se estructura en partidas trazables a capacidades operativas y técnicas de THOT:

- **Soporte y operación (L1/L2/L3)**: triage, diagnóstico, contención/recuperación, cambios, validación postcambio, coordinación entre módulos (datos, procesamiento, IA, CoC, integraciones y UI).
- **Observabilidad y trazabilidad operativa**: explotación del stack (métricas/logs/trazas), retención y almacenamiento, afinado de alertas y producción de evidencias (reduce MTTR y evita fatiga operativa).
- **Capacidad e infraestructura**: ampliación de nodos Kubernetes y recursos CPD, crecimiento de almacenamiento (BBDD y object storage), y mantenimiento preventivo/correctivo de la plataforma base.
- **Continuidad (Backup/DR)**: copias, verificación de restauración, simulacros, y almacenamiento secundario/replicación según criticidad.
- **Seguridad y PKI**: operación de certificados, rotación/revocación, auditoría, hardening y respuesta a incidentes de identidad o acceso.

- **Herramientas**: licencias si existieran o, si se adopta open source, el coste equivalente de operación/soporte (por ejemplo, Kong, Harbor, Prometheus/Grafana, Loki/Tempo, MLflow, etc.).
- **Formación y documentación viva**: actualización de runbooks/checklists y reciclaje ligado a cambios relevantes o lecciones aprendidas.

11.3. Escenarios mínimo, nominal y alto

Para evitar un único OPEX poco defendible, se definen tres escenarios parametrizados por drivers: mínimo (piloto o despliegue acotado), nominal (operación estabilizada) y alto (crecimiento significativo de usuarios/casos/volumen y mayor exigencia de retención y auditoría). La comparación entre escenarios mantiene el mismo desglose de partidas y varía únicamente drivers, de forma que el incremento de coste sea explicable, medible y controlable.

11.4. Sostenibilidad técnica: obsolescencia, dependencias, estrategia EOL/EOS

La sostenibilidad técnica se asegura controlando obsolescencia de hardware y dependencias software mediante una estrategia explícita de EOL/EOS (fin de vida/fin de soporte), que minimiza riesgo de seguridad, indisponibilidad y costes imprevistos.

En infraestructura, la disciplina se apoya en inventario/CMDB, criticidad por capa (cómputo, almacenamiento, red, virtualización y Kubernetes), y ciclos de renovación planificados con ventanas de mantenimiento y evidencias de aceptación.

En software, la sostenibilidad se apoya en: (i) versionado y despliegues reversibles (GitOps/CI/CD), (ii) observabilidad como control de calidad operativo, (iii) cadena de suministro controlada (registry y firmas/escaneo), y (iv) reducción de dependencia de conocimiento tácito mediante runbooks y procedimientos verificables.

11.5. Resultados: tabla OPEX + narrativa de hipótesis y sensibilidad

A continuación se presenta una tabla plantilla parametrizable. No fija importes, ya que dependen de tarifas, SLAs, inventario real y política corporativa, pero define el método y las variables mínimas para que el OPEX sea auditable y comparable entre escenarios.

Partida OPEX	Driver principal	Unidad	Fórmula anual (plantilla)	Fuente/medición prevista
Soporte y operación (L1/L2/L3)	nº usuarios, nº incidencias, nº releases	€/año	(FTE_L1+FTE_L2+FTE_L3)×coste_FTE + guardias (si aplica)	Ticketing + métricas MTTR/SLA
Observabilidad (métricas/logs/trazas)	volumen eventos, retención	€/año	coste_explotación + coste_storage_retención	Prometheus/Loki/Tempo + métricas de ingesta
Infraestructura y capacidad (CPD/K8s)	CPU/RAM/GPU, nodos, IOPS, red	€/año	amortización/servicio + ampliaciones planificadas	Métricas de capacidad + CMDB + compras

Datos y almacenamiento	crecimiento BBDD + object storage	€/año	coste_storage + coste_IOPS + expansión	Dashboards storage + tendencias
Backup/DR y continuidad	criticidad/RTO-RPO	€/año	coste_backup + coste_DR + pruebas_restore	Informes backup/restore + actas simulacro
Seguridad/PKI	nº certificados, rotaciones, incidentes	€/año	coste_operación_PKI + soporte + auditoría	Inventario certs + eventos revocación + tickets
Herramientas/licencias/soporte	nº componentes, política soporte	€/año	licencias (si aplica) + soporte/operación	Contratos/licencias + esfuerzo operación
Formación y documentación	nº usuarios, nº cambios relevantes	€/año	coste_formación + actualización runbooks	Plan de formación + repositorio runbooks

Tabla 12. Plantilla de cálculo OPEX del PMS de THOT (Lote 1). *Resume las partidas de coste recurrente, sus drivers, unidades y fórmulas anuales parametrizables, junto con las fuentes de medición previstas (ticketing, observabilidad, CMDB/capacidad, backup/DR y seguridad), para asegurar un OPEX auditable y comparable entre escenarios.*

Escenario	**Nº usuarios**	**Casos/año**	**Volumen datos/año**	**Retención observabilidad**	**OPEX total anual (suma de partidas)**
Mínimo	U_min	C_min	D_min	R_min	Σ OPEX_partidas(U_min, C_min, D_min, R_min)
Nominal	U_nom	C_nom	D_nom	R_nom	Σ OPEX_partidas(U_nom, C_nom, D_nom, R_nom)
Alto	U_alto	C_alto	D_alto	R_alto	Σ OPEX_partidas(U_alto, C_alto, D_alto, R_alto)

Tabla 13. Escenarios de dimensionamiento para estimación de OPEX (mínimo, nominal y alto). *Define los parámetros de entrada (usuarios, casos/año, volumen de datos y retención de observabilidad) y la agregación del OPEX anual como suma de partidas, permitiendo análisis de sensibilidad y comparación consistente entre escenarios.*

Con este marco, la sostenibilidad económica queda vinculada a métricas reales de explotación (capacidad, volumen, retención, incidencias y cambios), y la sostenibilidad técnica se garantiza mediante control de obsolescencia, operación basada en evidencia y prácticas de despliegue/seguridad que minimizan recurrencia y costes imprevistos.

12. ANEXOS

A. Matriz de trazabilidad completa (requisito ↔ sección ↔ evidencia)
B. Catálogo detallado de alertas/acciones por módulo

Familia	Dominio / Módulo	Alerta (nombre/objetivo)	Condición/umbral y ventana	Severidad	SLA objetivo (reacción / MTTR)	Responsable inicial (L1)	Escalado (L2/L3)	Evidencias enlazables (mínimas)
Servicio	API Gateway / Exposición APIs	**Indisponibilidad de API crítica** (restaurar acceso)	Readiness KO o 5xx elevado sostenido durante ventana	Crítica	≤ 15 min / ≤ 4 h	Operación (L1)	L2: Responsable del servicio/API; L2-Infra: Plataforma; L3: Ingeniería	Ticket + correlation_id; panel de latencia/errores; LogQL de 5xx/timeouts; trazas de endpoint; estado de despliegue/versión
Servicio	Microservicios THOT	**Reinicios anómalos / crashloop** (evitar degradación)	Reinicios repetidos por servicio en ventana	Alta	≤ 30 min / ≤ 8 h	Operación (L1)	L2: Service Owner; L2-Infra: Plataforma; L3: Ingeniería	Panel de reinicios; eventos Kubernetes; logs del pod; cambios recientes (CI/CD/GitOps); recursos/limits
Servicio	Data Space (BBDD/DBaaS)	**Degradación de persistencia** (evitar bloqueo de operación)	Latencia/errores DB por encima de baseline durante ventana	Crítica	≤ 15 min / ≤ 4 h	Operación (L1)	L2: Responsable Persistencia; L2-Infra: Storage/DBA; L3: Ingeniería	Panel DB (latencia, conexiones, locks); logs DB; métricas storage/IO; trazas de operaciones; evidencia de saturación
Procesamiento	Kafka / Mensajería	**Backlog sin drenaje** (prevenir retrasos)	Crecimiento sostenido de lag/backlog sin tendencia a drenaje	Alta	≤ 30 min / ≤ 8 h	Operación (L1)	L2: Responsable Procesamiento; L2-Infra: Plataforma; L3: Capacidad/Arquitectura	Panel lag/throughput; métricas brokers/consumidores; logs de reintentos; estado de particiones; cambios recientes

Procesamiento	Airflow / Workflows	**Fallos recurrentes en pipeline crítico** (restaurar flujo)	Tasa de fallo anómala o DAG crítico KO en ventana	Alta	≤ 30 min / ≤ 8 h	Operación (L1)	L2: Responsable Pipelines; L3: Ingeniería	Logs de ejecución; panel de DAGs; trazas (si aplica); evidencia de dependencias fallidas; versión/config
Interoperabilidad	Integración Lote 2 (REST/gRPC/MQTT/Kafka)	**Fallo persistente de interoperabilidad** (evitar pérdida E2E)	Errores de validación/contrato o fallos de entrega sostenidos en ventana	Crítica	≤ 15 min / ≤ 4 h	Operación (L1)	L2: Responsable Interoperabilidad; L2-Sec: Seguridad/PKI (si mTLS); L3: Arquitectura/Integración	Ticket + correlation_id; panel por integración; LogQL validación/errores; métricas de rechazo/latencia; estado mTLS/cert
Interoperabilidad	Integración Lote 2 (mensajería/eventos)	**Tasa de rechazo por esquema/contrato elevada** (proteger consistencia)	% de mensajes rechazados > baseline en ventana	Alta	≤ 30 min / ≤ 8 h	Operación (L1)	L2: Interoperabilidad; L3: Arquitectura/Contratos	Dashboard de validación; ejemplos de payload (redactados); versión de contrato activa; logs de validación; cambios recientes
Seguridad	PKI / mTLS	**Fallos de mTLS / handshake** (restaurar canal seguro)	Tasa de fallos TLS/handshake sostenida en ventana	Crítica	≤ 15 min / ≤ 4 h	Operación (L1)	L2-Sec: Seguridad/PKI; L3: Seguridad/Arquitectura	Logs TLS/handshake; métricas de fallos por integración; inventario cert; estado CRL/OCSP; evidencia de rotación/expiración
Seguridad	IAM / RBAC	**Anomalía de autenticación/autorización** (contener accesos no válidos)	Pico de fallos auth/denegaciones o patrón anómalo en ventana	Alta	≤ 30 min / ≤ 8 h	Operación (L1)	L2-Sec: Seguridad/PKI; L3: Seguridad	Logs auth; métricas 4xx/denegaciones; panel de seguridad; cambios recientes de permisos; acciones de contención
Seguridad	PKI / Certificados	**Certificado próximo a caducar** (evitar caída por expiración)	Expira < 30 días (warning) / < 7 días (crítico)	Media/Alta	≤ 5 días (crítico) / N/A	Operación (L1)	L2-Sec: Seguridad/PKI; L3: Seguridad	Inventario cert; fechas expiración; plan/registro de rotación; verificación

								mTLS post-rotación; ticket de cambio
CoC (integridad)	CoC Central	**Riesgo de integridad** (evitar evidencia inválida)	Falla verificación de integridad / inconsistencia detectada	Crítica	≤ 15 min / ≤ 4 h	Operación (L1)	L2-CoC: Responsable CoC; L2-Infra: Plataforma; L3: Forense/Arquitectura	Evidencia de verificación KO; logs; panel CoC; acción de contención; verificación final
CoC (consistencia)	CoC + Ingesta Lote 2	**Inconsistencia evidencia↔evento CoC** (proteger trazabilidad)	Evidencia recibida sin evento CoC esperado (o viceversa) en ventana	Alta	≤ 30 min / ≤ 8 h	Operación (L1)	L2-CoC: CoC; L2: Interoperabilidad; L3: Forense/Arquitectura	Métricas de contaje (evidencias vs eventos); correlation_id; logs de consolidación; estado de colas; evidencias de reprocesado/contención
Infraestructura	CPD/Cluster/Storage	**Saturación sostenida de recursos** (prevenir caída)	CPU/memoria/storage/IO sostenido > umbral/baseline en ventana	Alta	≤ 30 min / ≤ 8 h	Operación (L1)	L2-Infra: Plataforma/CPD; L3: Capacidad	Panel capacidad por nodo/servicio; métricas I/O; eventos de throttling/OOM; trend semanal; acciones aplicadas
Infraestructura	CPD/Hardware (si telemetría disponible)	**Degradación HW (disco/RAID/memoria/red)** (evitar fallo)	Alertas HW o errores ECC/SMART/link flaps repetidos	Alta	≤ 60 min / ≤ 24 h	Operación (L1)	L2-Infra: CPD; L3: Operador CPD/Proveedor	Eventos HW; métricas storage/network; inventario/CMDB del activo; ticket de intervención; evidencia de reemplazo/mitigación

Tabla 14. Catálogo de alertas operativas y escalado del Lote 1). *Resume por dominio las alertas mínimas, umbrales/ventanas, severidad y objetivos de reacción/MTTR, incluyendo responsable inicial (L1) y escalado (L2/L3).*

C. Plantillas (ticket cambio, parte incidencia, checklist despliegue, checklist pre/post escena)
D. Modelo de inventario y baseline (campos, formato)
E. Tabla OPEX (partidas, supuestos, fórmulas, escenarios)
F. Matriz RACI completa
G. Cuadro de mando de SLAs/KPIs (definición, fuente, cálculo, umbral, responsable)

Glosario

Concepto	Definición	Rol en el PMS
CMDB	Base de datos de configuración e inventario de activos y servicios con sus relaciones	Identificar qué está en producción, quién es responsable, dependencias y análisis de impacto ante cambios/incidencias
CI (Configuration Item)	Elemento gestionado en CMDB (servidor, nodo, servicio, BD, certificado, contrato API)	Unificar trazabilidad de operación y cambios sobre un "objeto" auditable
Service Ownership	Propiedad operativa de un servicio por un responsable	Asegurar continuidad, decisiones técnicas, calidad y evolución controlada por servicio
Process Ownership	Propiedad de un proceso transversal (cambios, incidencias, despliegues, DR)	Garantizar que los procesos se ejecutan igual, con controles y evidencias
Soporte L1	Operación: triage, runbooks, contención básica	Reducir MTTA/MTTR y filtrar incidencias antes de escalar
Soporte L2	Especialistas por módulo: diagnóstico avanzado y ajustes	Resolver incidencias complejas sin tocar ingeniería del producto
Soporte L3	Ingeniería: correcciones estructurales, hotfix/rollback complejos	Resolver causas raíz y ejecutar cambios evolutivos/estructurales
ITSM / Ticketing	Herramienta de gestión de incidencias, problemas, cambios y peticiones	Repositorio único de evidencias operativas y auditoría
Incidencia	Evento que degrada o interrumpe el servicio	Activar ciclo correctivo con SLAs y escalado
Problema	Causa subyacente de una o varias incidencias	Reducir recurrencia mediante RCA y acciones permanentes
RCA	Root Cause Analysis, análisis de causa raíz	Explicar por qué ocurrió y qué cambio evita repetición
CAPA	Acciones Correctivas y Preventivas	Cerrar el ciclo de mejora continua (lo ocurrido y lo que podría ocurrir)

Cofinanciado por la Unión Europea
Fondos Europeos

RFC / Change Request	Solicitud formal de cambio (infra/servicio/modelo/contrato)	Controlar riesgo de cambios en un sistema forense
Change Management	Proceso para evaluar, aprobar, ejecutar y verificar cambios	Evitar degradación, mantener seguridad y CoC
Release	Entrega versionada de un servicio o conjunto de cambios	Gobernar despliegues y compatibilidad
Rollback	Reversión a versión estable anterior	Recuperación rápida ante degradación
GitOps	Estado deseado versionado en Git y reconciliado contra el clúster	Trazabilidad y consistencia de configuración en K8s
IaC (Infra as Code)	Infraestructura definida como código (plantillas, módulos)	Repetibilidad y auditoría de cambios de infra
CI/CD	Pipelines de build, test, scan, despliegue y verificación	Automatizar releases con controles y evidencias
Registry de contenedores	Repositorio de imágenes OCI (p. ej., Harbor)	Versionar, firmar, escanear y controlar qué se despliega
Digest	Huella inmutable de una imagen/artefacto	Reproducibilidad exacta de despliegues
SBOM	Lista de componentes y dependencias del software	Responder a CVEs y auditorías de cadena de suministro
Supply Chain Security	Controles sobre origen, integridad y distribución del software	Evitar artefactos no autorizados o manipulados
Policy-as-Code	Políticas versionadas aplicadas automáticamente (admisión)	Asegurar baseline de seguridad y operación
Observabilidad	Métricas, logs y trazas correlacionadas	Base “evidence-driven” para preventivo/correctivo
Métricas	Señales numéricas (latencia, errores, recursos)	Medir SLIs/SLOs, capacidad y degradación
Logs	Registro de eventos y errores	Diagnóstico y auditoría técnica
Trazas	Seguimiento distribuido de una petición end-to-end	Identificar cuellos de botella y fallos entre servicios
Correlation ID	Identificador para correlacionar API, logs, trazas y tickets	Trazabilidad operacional y RCA
Catálogo de alertas	Conjunto gobernado de alertas con severidad y runbook	Detección temprana y escalado consistente
Severidad	Clasificación del impacto (crítica/alta/media/baja)	Priorizar respuesta y SLAs

Cofinanciado por la Unión Europea
Fondos Europeos

SLA / SLO / SLI	Compromisos, objetivos e indicadores del servicio	Medir cumplimiento y sostener interoperabilidad
MTTA / MTTR	Tiempo hasta reconocer / tiempo hasta recuperar	Medir eficiencia operativa del PMS
Runbook	Procedimiento operativo estandarizado	Respuesta homogénea, reversible y auditable
Checklist operativa	Lista de verificación de aceptación o intervención	Control de calidad operativo
DR	Recuperación ante desastre (sitio/cluster alternativo)	Continuidad ante fallos mayores
RTO / RPO	Tiempo objetivo de recuperación / pérdida máxima de datos	Dimensionar backup/DR y pruebas
Backup verificado	Copia con prueba de restauración	Evitar “backups que no restauran”
Kubernetes	Orquestador del clúster de microservicios	Escalado, despliegues, aislamiento y resiliencia
HPA / VPA	Autoescalado horizontal/vertical	Ajustar capacidad ante carga variable
PDB	Presupuesto de interrupción de pods	Mantener disponibilidad en mantenimientos
NetworkPolicy	Segmentación y control de red en K8s	Reducir superficie de ataque y aislar dominios
Ingress / API Gateway	Punto de entrada y gobierno de tráfico (p. ej., Kong)	Seguridad, rate limit, autenticación, versionado
TLS / mTLS	Cifrado y autenticación mutua	Integración segura Lote 1–Lote 2 y servicio a servicio
PKI	Infraestructura de emisión/rotación/revocación de certificados	Identidad criptográfica y contención ante incidentes
Gestión de vulnerabilidades	Detección y remediación de CVEs en imágenes/hosts	Mantener seguridad en el tiempo
Interoperabilidad	Integración con Lote 2 y sistemas externos (contratos, validación)	Evitar rupturas y garantizar consistencia E2E
Contrato API	Especificación versionada (OpenAPI/gRPC/eventos)	Compatibilidad, validación y gobierno
Validación de payload	Comprobación de esquemas y reglas	Reducir errores y proteger consistencia del dato

Cofinanciado por la Unión Europea
Fondos Europeos

CoC centralizada	Registro inmutable de eventos de evidencia	Preservar validez forense y trazabilidad judicial
Integridad	Verificación de hashes/firmas/manifiestos	Evitar manipulación o corrupción de evidencias
Retención	Política de conservación de logs/datos/evidencias	Cumplimiento y auditoría
OPEX	Coste recurrente de operación y evolución	Sostener soporte, herramientas y continuidad

Tabla 15. Glosario de términos operativos del PMS de THOT (Lote 1). *Define los conceptos clave utilizados en el PMS y su rol en el documento.*