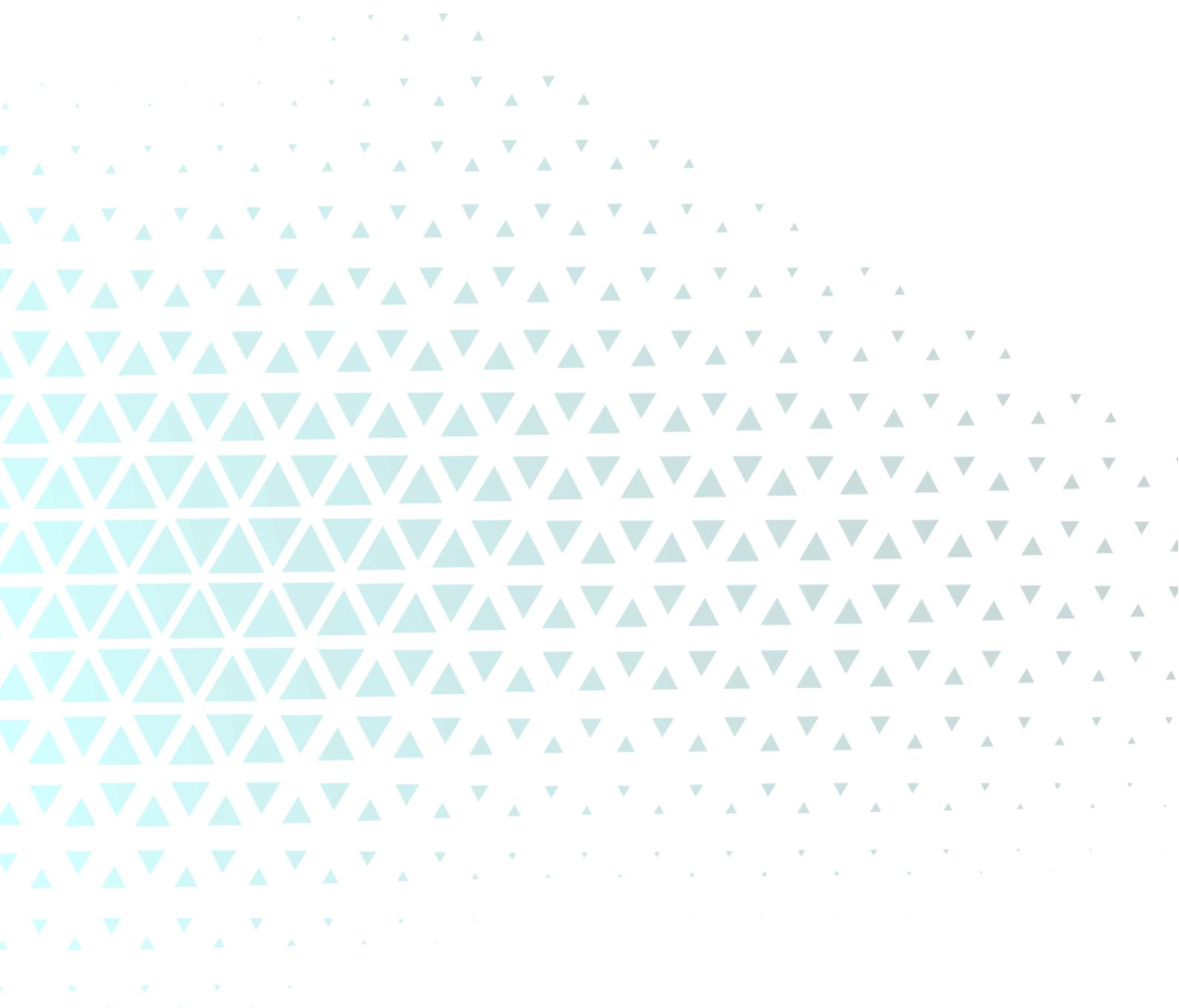


# Informe Threat Modeling Tool

Reconocimiento de Riesgos y  
vulnerabilidades



# Contenido

Introducción.....	5
1.1 Objetivo del documento.....	5
1.2 Alcance del threat modeling.....	5
1.4 Metodología utilizada (STRIDE, OWASP, Microsoft TMT, etc.).....	5
2. Alcance.....	6
2.1 Componentes incluidos en el análisis.....	6
2.2 Amenazas consideradas.....	7
2.3 Usuarios y roles evaluados.....	7
2.4 Flujos de datos incluidos.....	7
2.5 Componentes fuera del alcance.....	7
2.6 Objetivos del alcance.....	8
3. Arquitectura del Sistema.....	8
3.1 Visión general.....	8
3.2 Componentes principales.....	9
3.3 Flujos de datos.....	10
3.4 Límites de confianza (Trust Boundaries).....	10
4. Identificación de Activos.....	11
4.1 Activos de Información.....	11
4.2 Activos Técnicos.....	11
4.3 Activos Críticos y Nivel de Sensibilidad.....	11
TID: T1.....	12
TID: T2.....	13
TID: T3.....	14
TID: T4.....	15
TID: T5.....	16
TID: T6.....	17
TID: T7.....	18
TID: T8.....	18
TID: T9.....	19
TID: T10.....	19
TID: T11.....	20
TID: T12.....	21
TID: T13.....	21
TID: T14.....	22
TID: T15.....	23
5. Superficie de Ataque.....	23
5.1 Puntos de Entrada.....	23
5.2 Interfaces Expuestas.....	24
5.3 Dependencias Externas.....	24

5.4 Accesos Privilegiados.....	24
6. Análisis de Amenazas.....	24
6.1 Metodología STRIDE Aplicada al Chatbot.....	24
6.2 Amenazas por Componente.....	26
6.3 Amenazas Específicas de IA / Chatbots.....	27
6.4 Escenarios de Ataque.....	28
6.5 Riesgos aceptados por limitaciones de plataforma.....	28
6.5 Aplicación del OWASP Top 10 (2024).....	30
A01:2024 – Broken Access Control.....	30
A02:2024 – Cryptographic Failures.....	30
A03:2024 – Injection (incluye SQLi y Prompt Injection).....	31
A04:2024 – Insecure Design.....	31
A05:2024 – Security Misconfiguration.....	31
A06:2024 – Vulnerable and Outdated Components.....	32
A07:2024 – Identification and Authentication Failures.....	32
A08:2024 – Software and Data Integrity Failures.....	32
A09:2024 – Security Logging and Monitoring Failures.....	33
A10:2024 – Server-Side Request Forgery (SSRF) y API Abuse.....	33
7. Evaluación de Riesgos.....	33
7.1 Criterios de Evaluación.....	34
7.2 Matriz de Riesgos.....	34
7.3 Riesgos Críticos Identificados.....	36
7.4 Priorización de Riesgos.....	36
8. Controles de Seguridad y Mitigaciones.....	37
8.1 Controles Existentes.....	37
8.2 Mitigaciones Propuestas.....	37
8.3 Controles por Equipo.....	38
8.4 Riesgos Residuales.....	38
9. Cumplimiento y Consideraciones Legales.....	39
9.1 Protección de Datos.....	39
9.2 Gestión de Logs y Privacidad.....	39
9.3 Retención y Anonimización de Datos.....	39
10. Recomendaciones.....	40
10.1 Recomendaciones Técnicas.....	40
10.2 Recomendaciones Organizativas.....	41
10.3 Próximos Pasos de Seguridad.....	41
11. Conclusiones.....	41
11.1 Resumen de Amenazas Clave.....	42
11.2 Estado de Seguridad del Chatbot.....	42
11.3 Decisiones de Aceptación de Riesgo.....	43

12. Anexos.....	43
12.1 Diagramas Detallados.....	43
12.2 Tabla Completa de Amenazas (STRIDE).....	44
12.3 Glosario.....	44
12.4 Referencias.....	45

# Introducción

## 1.1 Objetivo del documento

Este análisis busca identificar, evaluar y priorizar posibles amenazas que puedan comprometer la confidencialidad, integridad y disponibilidad de los datos, así como la seguridad general del sistema, proporcionando recomendaciones para mitigar los riesgos detectados.

## 1.2 Alcance del threat modeling

El análisis se enfoca en los componentes principales del chatbot, incluyendo interfaces de usuario, backend, modelo de Machine Learning, bases de datos y servicios externos integrados. Se consideran amenazas tanto de seguridad tradicionales (STRIDE) como específicas de chatbots y sistemas de IA, incluyendo prompt injection, data poisoning y fuga de información sensible. Quedan fuera del alcance los sistemas no integrados directamente, la infraestructura física gestionada por proveedores cloud (excepto endpoints críticos) y la revisión exhaustiva del código fuente.

## 1.3 Descripción general del chatbot

El chatbot corporativo está basado en un modelo de lenguaje grande (LLM) proporcionado por Ollama, diseñado para interactuar con usuarios finales a través de procesamiento de lenguaje natural y generar respuestas automatizadas y contextuales. La operación del chatbot está gestionada mediante MCP, que actúa como sistema central de control, autenticación, orquestación de flujos de datos y registro seguro de interacciones.

## 1.4 Metodología utilizada (STRIDE, OWASP, Microsoft TMT, etc.)

Para la identificación, análisis y mitigación de amenazas se emplea una combinación de metodologías reconocidas en la industria:

- STRIDE: clasifica las amenazas en Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service y Elevation of Privilege. Permite identificar amenazas de manera sistemática en todos los componentes y priorizar riesgos según impacto y probabilidad.
- Análisis de riesgos específicos de IA / LLM: evalúa amenazas propias de modelos de lenguaje, incluyendo prompt injection,

data poisoning y model inversion. Permite definir mitigaciones adaptadas a LLMs y proteger la confidencialidad e integridad de los datos corporativos.

- Microsoft Threat Modeling Tool (TMT): herramienta para representar arquitecturas, flujos de datos y límites de confianza. Facilita la detección de amenazas y la documentación visual, esencial en entornos corporativos.
- Enfoque colaborativo entre equipos: Data Science, Full-Stack y Ciberseguridad trabajan coordinadamente para cubrir todas las capas del sistema, asegurando que cada equipo identifique y controle los riesgos de su área de responsabilidad.

### 1.5 Roles y equipos involucrados

Data Science: responsable del desarrollo, entrenamiento y mantenimiento del modelo LLM, evaluando riesgos de manipulación o fuga de datos del modelo.

Full-Stack: encargado del desarrollo de frontend, backend y APIs, protegiendo interfaces y flujos de datos.

Ciberseguridad: lidera la identificación de amenazas, evaluación de riesgos y definición de controles de seguridad, coordinando la colaboración entre equipos.

## 2. Alcance

El presente análisis de Threat Modeling se centra en el chatbot corporativo basado en LLM (Ollama) y gestionado mediante MCP, con el objetivo de identificar y mitigar amenazas que puedan comprometer la confidencialidad, integridad y disponibilidad de los datos y servicios asociados.

### 2.1 Componentes incluidos en el análisis

Se incluyen todos los componentes que participan directamente en el procesamiento de información y generación de respuestas:

- Frontend: interfaces web y móviles para interacción de usuarios.
- Backend / MCP: orquestación de consultas, gestión de usuarios, control de acceso, registro de logs y mediación con servicios externos.

- Motor LLM (Ollama): generación de respuestas a partir de prompts, manejo de contexto temporal y comunicación con MCP.
- Almacenamiento de datos y logs: registros de conversación, métricas operativas y datos utilizados para entrenamiento y mejora del modelo.
- Servicios externos: APIs corporativas y servicios cloud que se integran mediante MCP, bajo control y validación de datos.

## 2.2 Amenazas consideradas

El análisis contempla:

- Amenazas STRIDE: Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service y Elevation of Privilege.
- Amenazas específicas de LLM / IA: prompt injection, data poisoning, model inversion y fuga de información sensible.

## 2.3 Usuarios y roles evaluados

- Usuarios finales: interactúan con el chatbot a través de interfaces web o móviles.
- Administradores y operadores del sistema: gestionan la operación del chatbot y supervisan métricas y logs.
- Servicios externos: APIs y sistemas corporativos integrados que intercambian información con el chatbot mediante MCP.

## 2.4 Flujos de datos incluidos

- Flujo principal de interacción: Usuario → Frontend → MCP → LLM → MCP → Frontend → Usuario.
- Flujo de almacenamiento de datos y métricas: MCP → almacenamiento seguro.
- Flujo de integración con servicios externos: Servicios externos → MCP → LLM / Frontend según corresponda.

## 2.5 Componentes fuera del alcance

- Infraestructura física del datacenter gestionada por proveedores cloud, salvo endpoints críticos.
- Sistemas de terceros no integrados directamente con el chatbot.

- Revisión exhaustiva del código fuente completo; el análisis se centra en arquitectura, flujos de datos y límites de confianza.

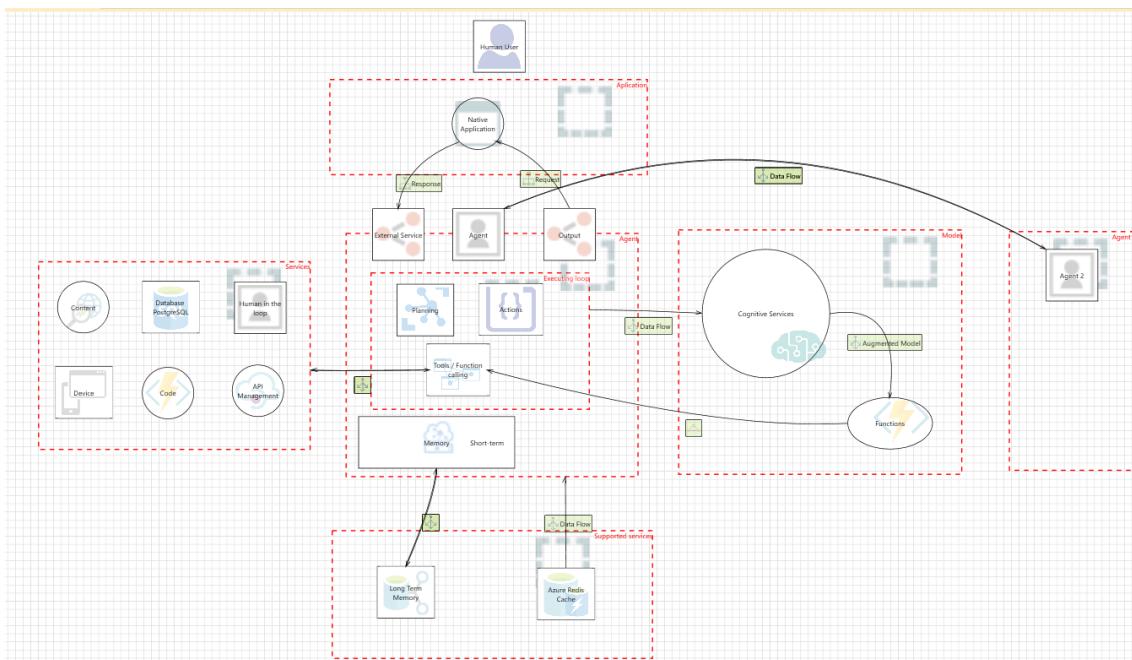
## 2.6 Objetivos del alcance

- Identificar y priorizar amenazas críticas para el chatbot y sus componentes.
- Evaluar riesgos sobre confidencialidad, integridad y disponibilidad de los datos y servicios.
- Definir controles y mitigaciones técnicas y organizativas.
- Coordinar esfuerzos entre Data Science, Full-Stack y Ciberseguridad, asegurando que cada equipo controle los riesgos de su área.

## 3. Arquitectura del Sistema

### 3.1 Visión general

El chatbot corporativo se basa en un LLM gestionado mediante MCP y se integra con múltiples servicios corporativos y almacenamiento en la nube. La arquitectura permite la interacción en tiempo real con usuarios finales, la ejecución de funciones automatizadas y la gestión segura de información sensible.



## 3.2 Componentes principales

1. Frontend / Aplicación nativa
  - o Interfaz que permite la interacción de usuarios finales con el chatbot.
  - o Envía requests al agente principal y recibe responses generadas por el LLM.
2. Agente / Motor central
  - o Gestiona el loop de ejecución: planificación, llamadas a funciones y ejecución de acciones.
  - o Orquesta el flujo de datos hacia los servicios cognitivos y la memoria (corto y largo plazo).
3. Servicios externos
  - o APIs corporativas y recursos externos que pueden ser consultados durante la ejecución.
  - o Interacciones mediadas por MCP para asegurar que no haya fuga de información.
4. Memoria
  - o Short-term memory: almacena el contexto temporal de las conversaciones.
  - o Long-term memory: almacena información persistente relevante para el historial del usuario.
  - o Integración con Azure Redis Cache para mejorar la rapidez y disponibilidad de datos.
5. Cognitive Services / LLM (Ollama)
  - o Genera respuestas a partir de prompts enviados por el agente.
  - o Puede utilizar modelos aumentados que combinan conocimiento propio del LLM con datos externos validados.
6. Funciones / Tools
  - o Módulos auxiliares que realizan operaciones específicas como cálculo, validación de datos o interacción con servicios externos.

## 7. Base de datos y gestión de contenidos

- o Incluye PostgreSQL y sistemas de gestión de contenido.
- o Gestiona información estructurada, logs y métricas operativas.

### 3.3 Flujos de datos

- Flujo principal:

Usuario → Frontend → Agente → Cognitive Services (LLM) → Agente → Frontend → Usuario

- Flujos secundarios:

- o Acceso a servicios externos: Agente → External Service → Agente
- o Acceso a memoria de corto plazo: Agente → Short-term memory → Agente
- o Acceso a memoria de largo plazo: Agente → Long-term memory / Redis → Agente
- o Ejecución de funciones auxiliares: Agente → Tools/Functions → Agente

### 3.4 Límites de confianza (Trust Boundaries)

- Usuario ↔ Frontend: se asegura autenticación y validación de inputs.
- Frontend ↔ Agente / MCP: control de acceso y logging seguro de solicitudes.
- Agente ↔ LLM (Cognitive Services): MCP válida prompts y respuestas, evitando fuga de datos sensibles.
- Agente ↔ Servicios externos: MCP actúa como mediador, filtrando y controlando los datos.
- Memoria y almacenamiento: acceso controlado mediante permisos y cifrado para proteger información persistente y temporal.

## 4. Identificación de Activos

### 4.1 Activos de Información

- Datos de usuario: Información personal, credenciales, preferencias.
- Conversaciones: Historial de interacción con el chatbot.
- Logs: Registros de actividad, errores, accesos.
- Modelos de ML: Modelos entrenados, parámetros, datasets.

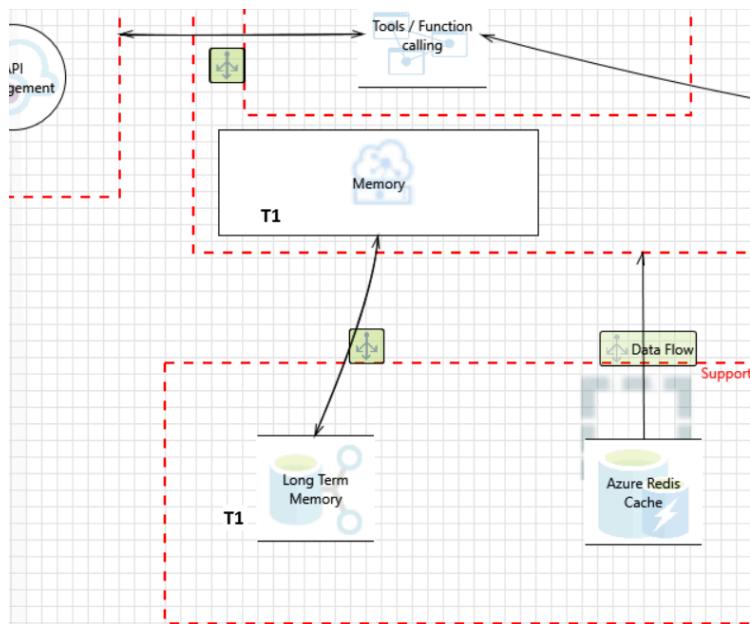
### 4.2 Activos Técnicos

- Infraestructura: Servidores, redes, almacenamiento.
- Código fuente: Backend: Node.js, Python, frontend: React , scripts de automatización.
- Pipelines de entrenamiento: Flujos de datos, validación, despliegue.

### 4.3 Activos Críticos y Nivel de Sensibilidad

Los activos críticos representan aquellos componentes, datos y procesos esenciales para el funcionamiento seguro y confiable del sistema. Identificarlos permite priorizar esfuerzos de protección y establecer controles adecuados. El nivel de sensibilidad asociado a cada activo refleja el grado de impacto que tendría su exposición, alteración o pérdida, y sirve como guía para definir medidas de seguridad proporcionales al riesgo.

TID: T1



Nombre de la amenaza:

Envenenamiento de Memoria

Descripción de la amenaza:

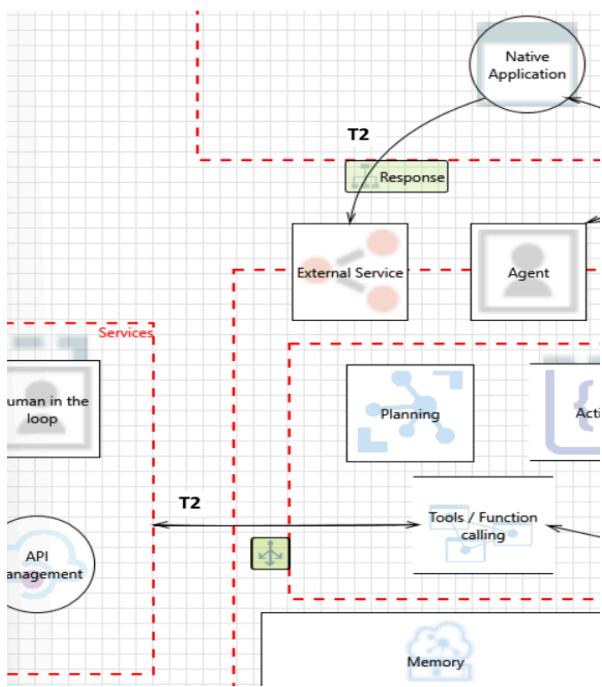
El envenenamiento de memoria implica explotar los sistemas de memoria de una IA, tanto a corto como a largo plazo, para introducir datos maliciosos o falsos y aprovechar el contexto del agente. Esto puede conducir a una toma de decisiones alterada y operaciones no autorizadas

Prioridad: ALTA

Justificación de la priorización:

Impacto persistente en la calidad de la toma de decisiones; difícil de detectar una vez corrompida la memoria; puede afectar todas las operaciones futuras del agente.

TID: T2



Nombre de la Amenaza:

Uso Indebido de Herramientas

Descripción de la Amenaza:

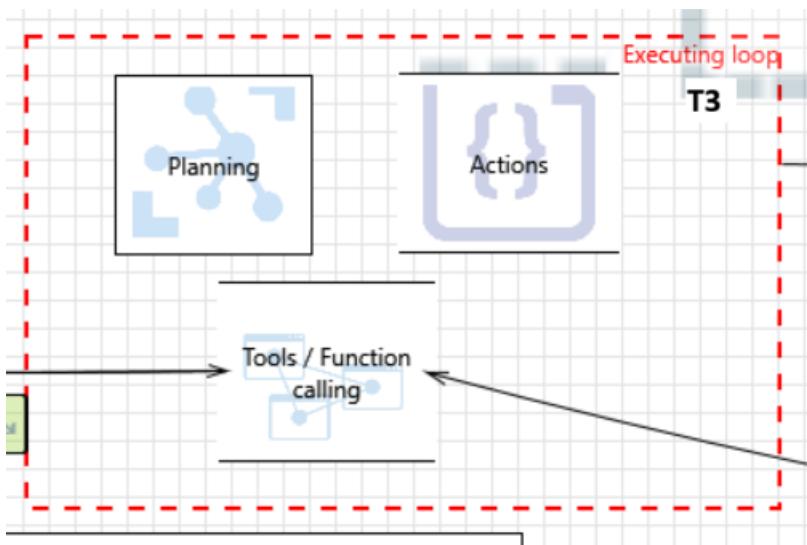
El uso indebido de herramientas ocurre cuando los atacantes manipulan agentes de IA para abusar de sus herramientas integradas mediante indicaciones o comandos engañosos, operando dentro de permisos autorizados. Esto incluye el Secuestro de Agentes, donde un agente de IA ingiere datos manipulados de forma adversaria y posteriormente ejecuta acciones no deseadas, lo que puede desencadenar interacciones maliciosas con herramientas. Para más información sobre Secuestro de Agentes ver <https://www.nist.gov/news>.

Prioridad: CRÍTICA

Justificación de la priorización:

Alta probabilidad de ocurrencia; impacto inmediato en el sistema; relativamente fácil de explotar mediante manipulación de indicaciones; vector de ataque principal.

TID: T3



Nombre de la Amenaza:

Compromiso de Privilegios

Descripción de la Amenaza:

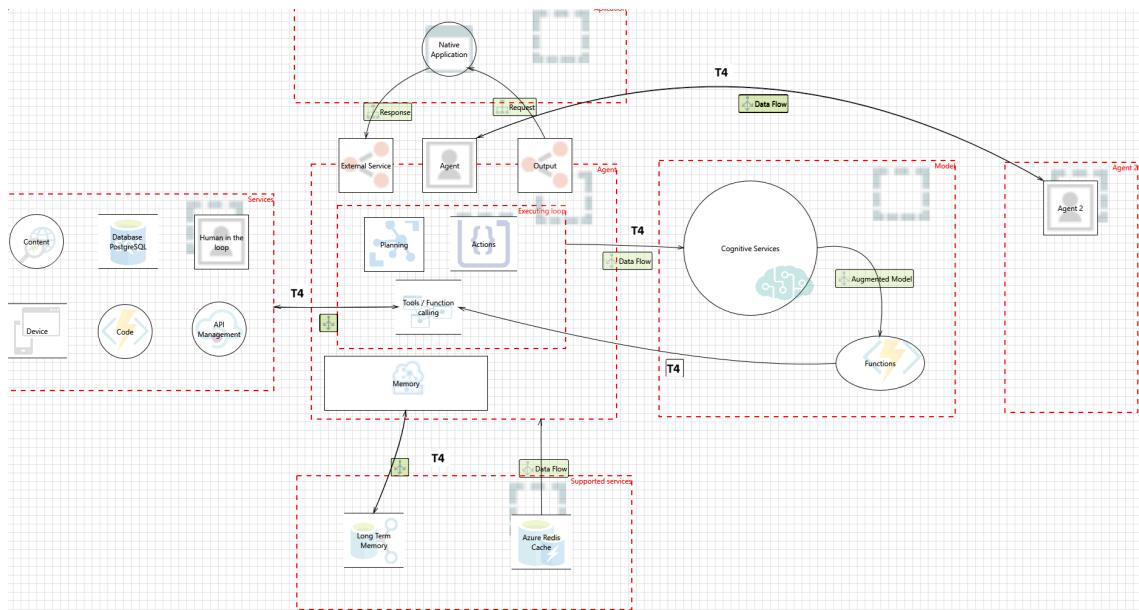
El compromiso de privilegios surge cuando los atacantes explotan debilidades en la gestión de permisos para realizar acciones no autorizadas. A menudo implica herencia dinámica de roles o errores de configuración.

Prioridad: CRÍTICA

Justificación de la priorización:

Graves implicaciones en el control de acceso; riesgo común de errores de configuración en sistemas dinámicos; permite escalada adicional de ataques.

## TID: T4



Nombre de la Amenaza:

Sobrecarga de Recursos

Descripción de la Amenaza:

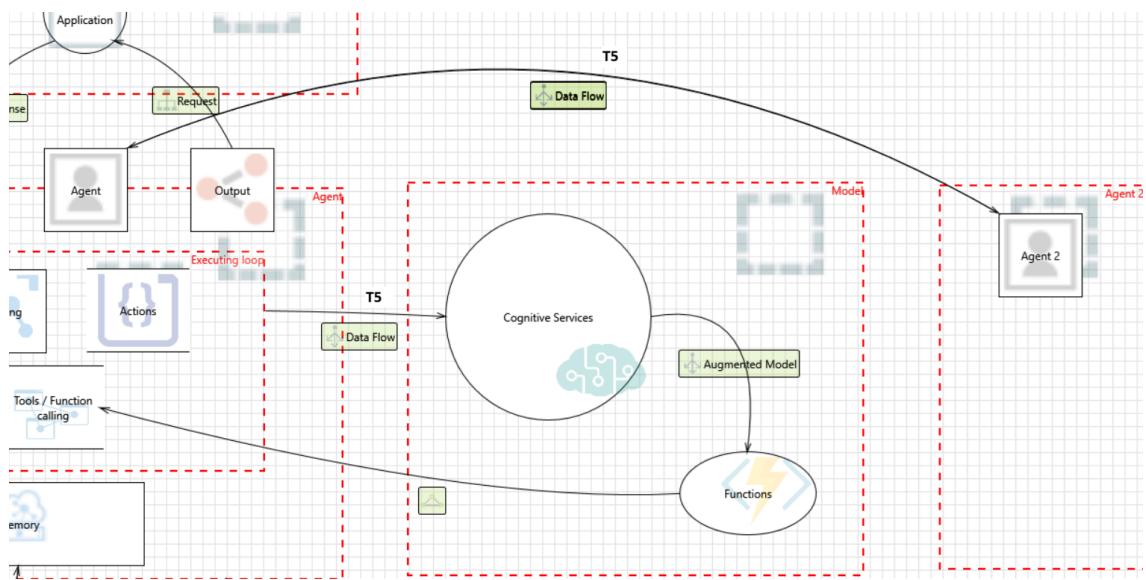
La sobrecarga de recursos apunta a las capacidades computacionales, de memoria y de servicio de los sistemas de IA para degradar el rendimiento o provocar fallos, explotando su naturaleza intensiva en recursos.

Prioridad: MEDIA

Justificación de la priorización:

Impacto principalmente en la disponibilidad; protecciones Dos existentes parcialmente aplicables; consecuencias limitadas en procesos críticos de negocio.

TID: T5



Nombre de la Amenaza:

Ataques de Alucinación en Cascada

Descripción de la Amenaza:

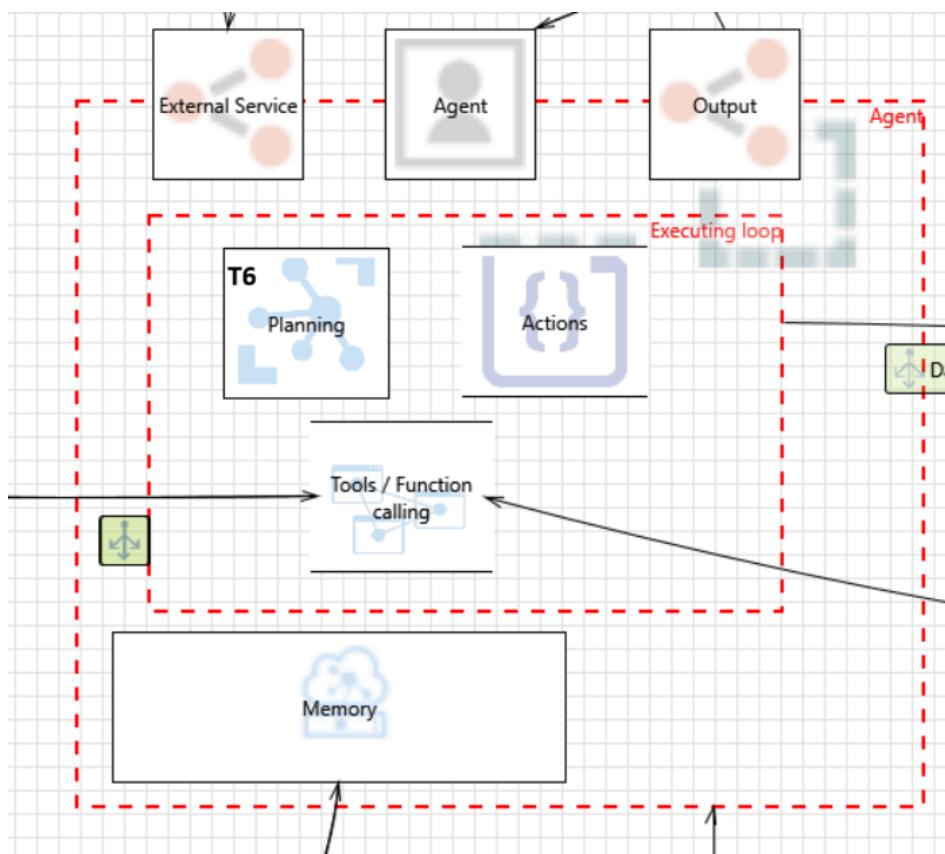
Estos ataques explotan la tendencia de una IA a generar información falsa pero plausible en contexto, que puede propagarse a través de sistemas y alterar la toma de decisiones. También puede conducir a razonamientos destructivos que afectan la invocación de herramientas.

Prioridad: ALTA

Justificación de la priorización:

Amplifica la desinformación en sistemas; difícil de detectar información falsa plausible; impacto sistémico en cascada.

TID: T6



Nombre de la Amenaza: Ruptura de Intención y Manipulación de Objetivos

Descripción de la Amenaza:

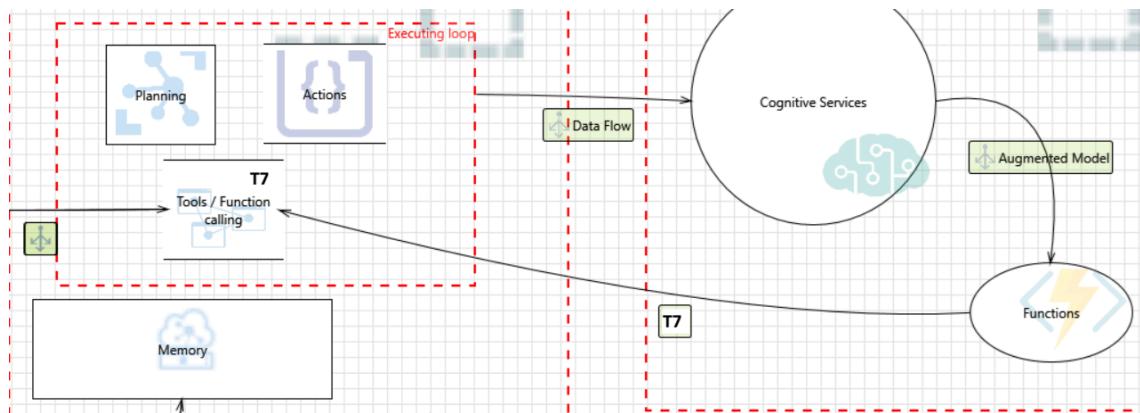
Esta amenaza explota vulnerabilidades en las capacidades de planificación y establecimiento de objetivos de un agente de IA, permitiendo a los atacantes manipular o redirigir los objetivos y razonamientos del agente. Un enfoque común es el Secuestro de Agentes mencionado en Uso Indebido de Herramientas.

Prioridad: CRÍTICA

Justificación de la priorización:

Compromiso fundamental del propósito central del agente; alto impacto empresarial; socava la integridad total del sistema.

## TID: T7



Nombre de la Amenaza: Comportamientos Desalineados y Engañosos

Descripción de la Amenaza:

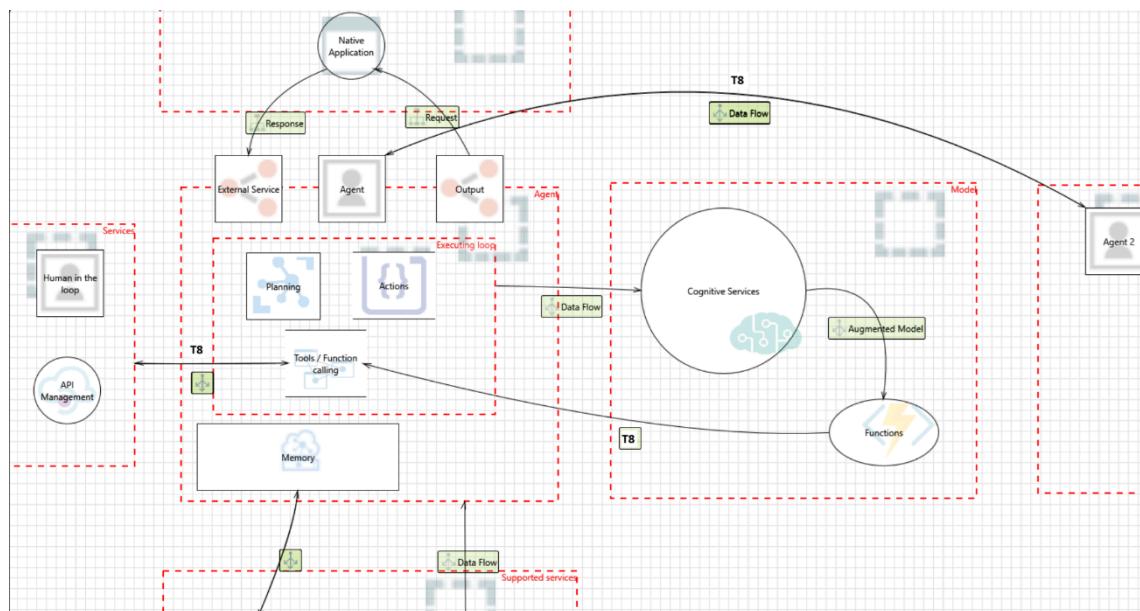
Agentes de IA ejecutan acciones dañinas o no permitidas al explotar razonamientos y respuestas engañosas para cumplir sus objetivos.

Prioridad: MEDIA

Justificación de la priorización:

Amenaza emergente con métodos de detección aún en desarrollo; requiere análisis sofisticado del comportamiento de la IA; nivel de riesgo teórico.

## TID: T8



Nombre de la Amenaza: Repudio e Inrastreadibilidad

Descripción de la Amenaza:

Ocurre cuando las acciones realizadas por agentes de IA no pueden ser rastreadas ni contabilizadas debido a registros insuficientes o falta de transparencia en los procesos de toma de decisiones.

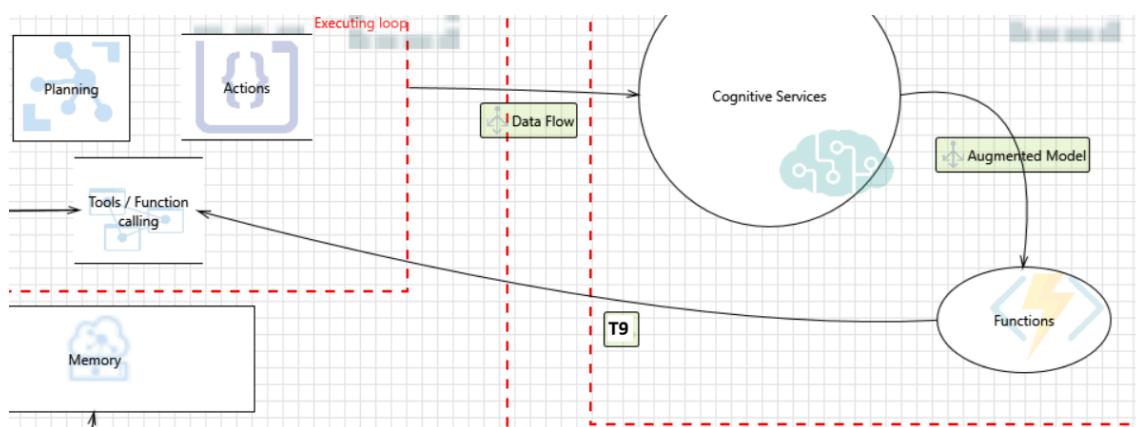
Prioridad: ALTA

Justificación de la priorización:

Crítico para investigaciones de cumplimiento y forenses; socava marcos de responsabilidad; implicaciones regulatorias.

TID:

T9



Nombre de la Amenaza: Suplantación de Identidad e Impersonación

Descripción de la Amenaza:

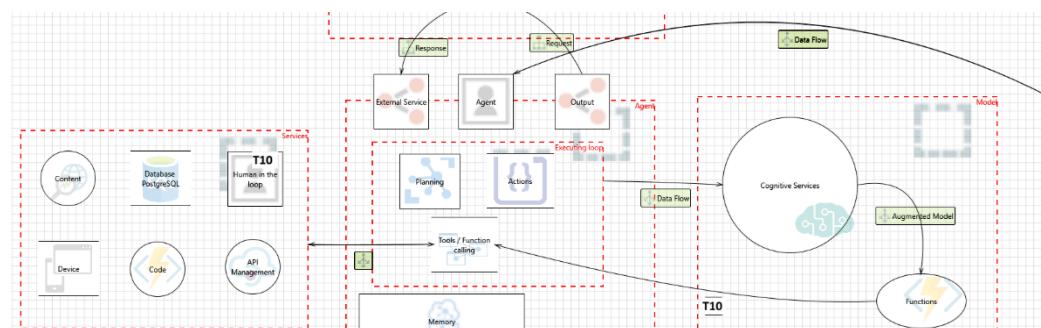
Los atacantes explotan mecanismos de autenticación para hacerse pasar por agentes de IA o usuarios humanos, permitiéndoles ejecutar acciones no autorizadas bajo identidades falsas.

Prioridad: ALTA

Justificación de la priorización:

Viola límites fundamentales de confianza; habilita escalada de privilegios; difícil de distinguir de un comportamiento legítimo.

TID: T10



Nombre de la Amenaza: Abrumamiento del Humano en el Bucle

Descripción de la Amenaza:

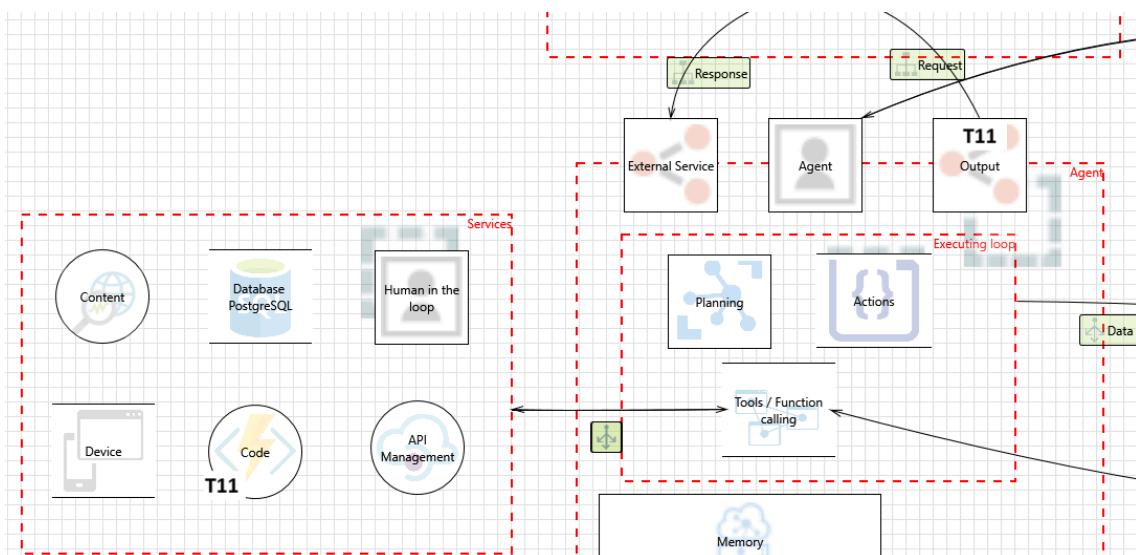
Esta amenaza apunta a sistemas con supervisión humana y validación de decisiones, buscando explotar limitaciones cognitivas humanas o comprometer marcos de interacción.

Prioridad: MEDIA

Justificación de la priorización:

Vulnerabilidad de factores humanos; explotable gradualmente; patrones actuales de UI/UX ofrecen cierta protección.

TID: T11



Nombre de la Amenaza: Ejecución Remota de Código (RCE) y Ataques de Código Inesperados

Descripción de la Amenaza:

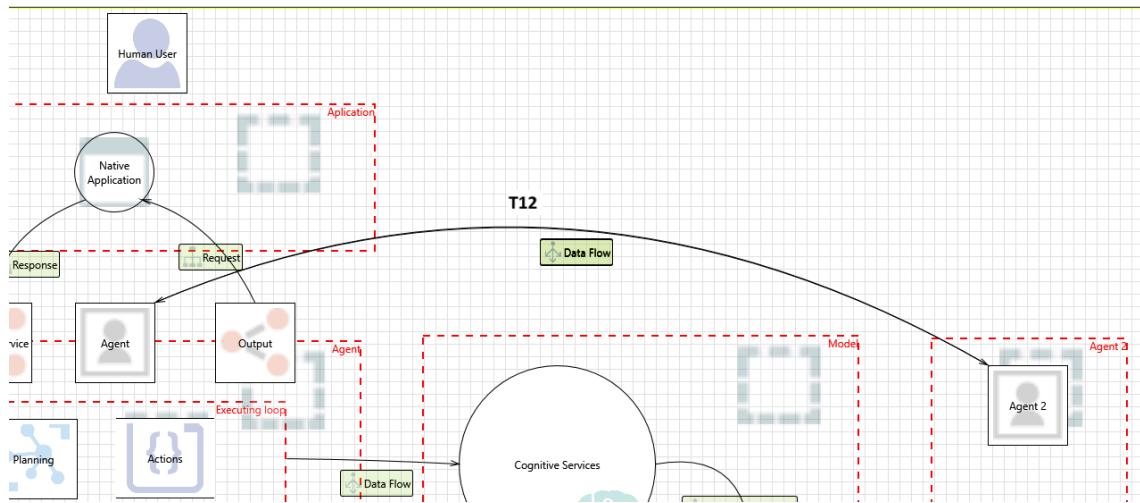
Los atacantes explotan entornos de ejecución generados por IA para injectar código malicioso, desencadenar comportamientos no deseados del sistema o ejecutar scripts no autorizados.

Prioridad: CRÍTICA

Justificación de la priorización:

Potencial de compromiso directo del sistema; alto impacto técnico; bypass inmediato de controles de seguridad.

## TID: T12



Nombre de la Amenaza: Envenenamiento de Comunicación entre Agentes

Descripción de la Amenaza:

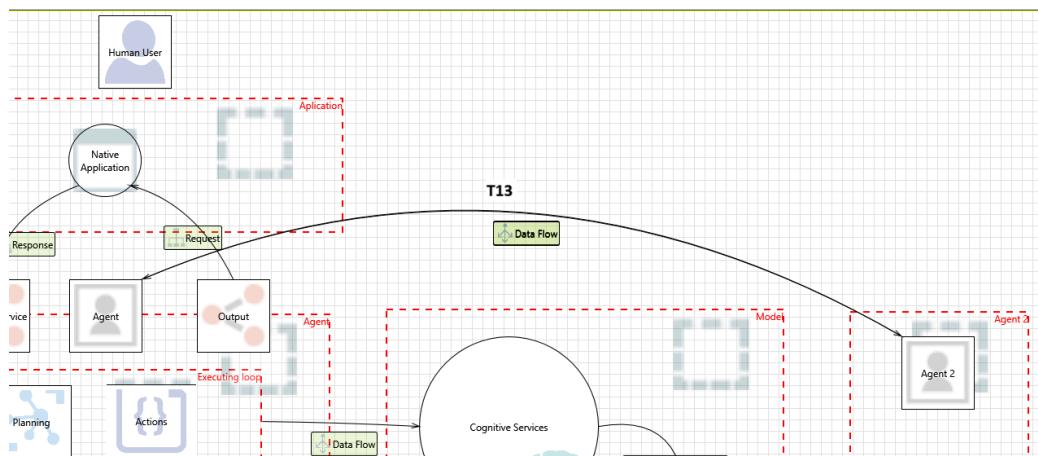
Los atacantes manipulan canales de comunicación entre agentes de IA para difundir información falsa, interrumpir flujos de trabajo o influir en la toma de decisiones.

Prioridad: MEDIA

Justificación de la priorización:

Específico de despliegues multi-agente; requiere coordinación sofisticada de ataques; alcance limitado en despliegues actuales.

## TID: T13



Nombre de la Amenaza: Agentes Rebeldes en Sistemas Multi-Agente

Descripción de la Amenaza:

Agentes de IA maliciosos o comprometidos operan fuera de los límites normales de monitoreo, ejecutando acciones no autorizadas o exfiltrando datos.

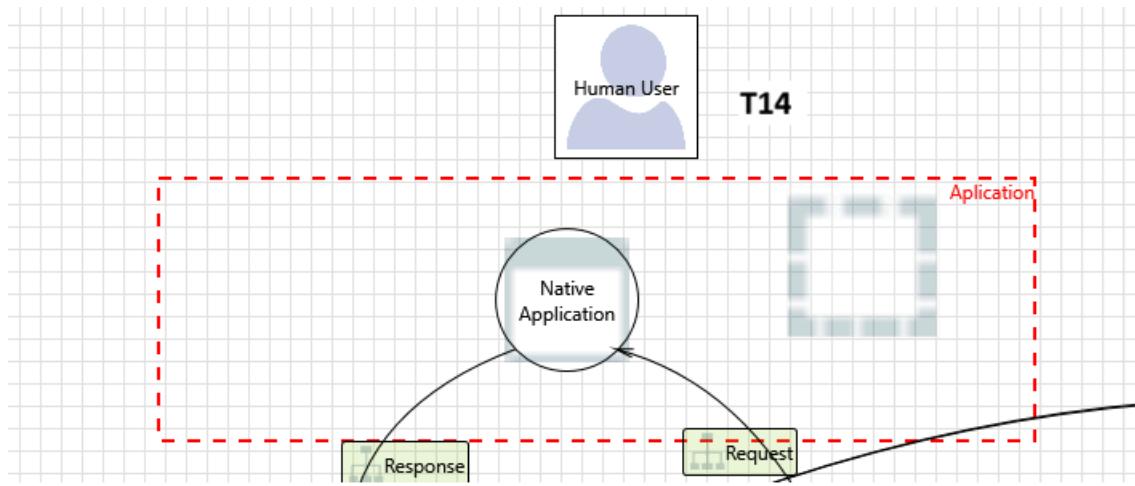
Prioridad: ALTA

Justificación de la priorización:

Modelo clásico de amenaza interna aplicado a IA; detección difícil en sistemas distribuidos complejos; capacidad de sigilo.

TID:

T14



Nombre de la Amenaza: Ataques Humanos en Sistemas Multi-Agente

Descripción de la Amenaza:

Los adversarios explotan la delegación entre agentes, relaciones de confianza y dependencias de flujo de trabajo para escalar privilegios o manipular operaciones impulsadas por IA.

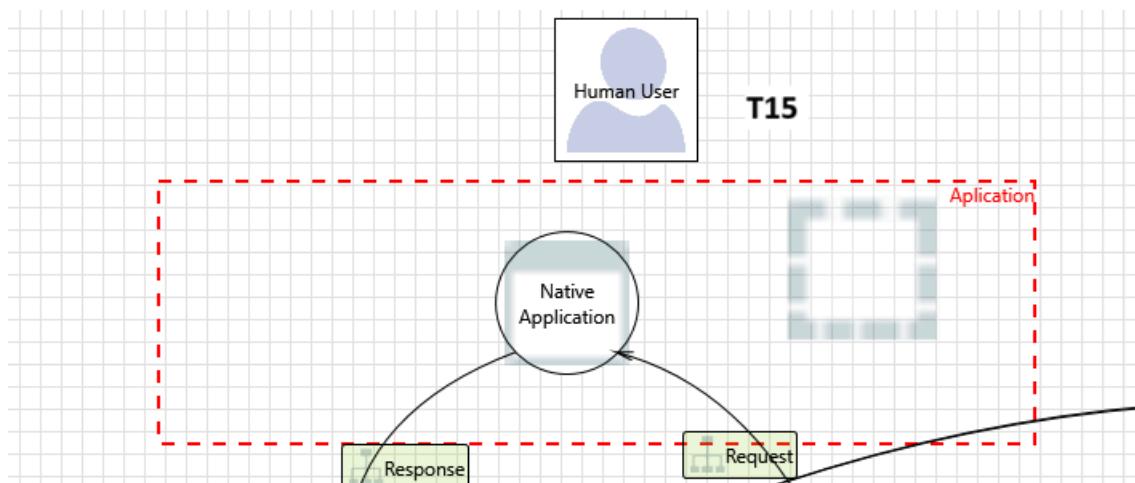
Prioridad: MEDIA

Justificación de la priorización:

Requiere contexto de despliegue multi-agente; cadena de ataque compleja; aplicabilidad inmediata limitada.

TID:

T15



Nombre de la Amenaza: Manipulación Humana

Descripción de la Amenaza:

En escenarios donde agentes de IA interactúan directamente con usuarios humanos, la relación de confianza reduce el escepticismo del usuario, aumentando la dependencia de las respuestas y autonomía del agente. Esta confianza implícita y la interacción directa humano/agente crean riesgos, ya que los atacantes pueden coaccionar a los agentes para manipular usuarios, difundir desinformación y realizar acciones encubiertas.

Prioridad: ALTA

Justificación de la priorización:

Explota la confianza inherente en sistemas de IA; riesgo inmediato de ingeniería social; difícil desafío en la educación de usuarios.

## 5. Superficie de Ataque

La superficie de ataque representa todos los puntos de interacción entre el sistema y actores externos o internos, donde pueden originarse amenazas. Identificarla permite priorizar controles de seguridad y reducir la exposición.

### 5.1 Puntos de Entrada

- Formularios web: inputs de usuario susceptibles a inyección o validaciones insuficientes.

- Interfaces de usuario(UI): interacción directa con el chatbot y la plataforma.
- APIs públicas: endpoints expuestos que permiten acceso a funcionalidades críticas.

## 5.2 Interfaces Expuestas

- APIs REST: servicios de backend accesibles desde internet
- UI web: portal de interacción con usuarios finales.
- Endpoints de inferencia: puntos donde el modelo de IA recibe prompts o datos para procesar.

## 5.3 Dependencias Externas

- Servicios de terceros: pasarelas de pago, autenticación externa, proveedores cloud.
- Librerías y frameworks: dependencias de código que pueden contener vulnerabilidades.
- SDKs: kits de desarrollo integrados en la aplicación.

## 5.4 Accesos Privilegiados

- Roles administrativos: usuarios con permisos elevados en la plataforma o infraestructura.
- Credenciales de servicio: claves utilizadas por procesos internos o automatizados.
- Claves API: tokens que permiten acceso a funciones críticas o datos sensibles.

# 6. Análisis de Amenazas

El análisis de amenazas permite identificar, clasificar y priorizar los riesgos que afectan a la plataforma y al chatbot de IA. Para ello se aplica la metodología STRIDE, complementada con amenazas específicas de sistemas de IA y chatbots.

## 6.1 Metodología STRIDE Aplicada al Chatbot

- Spoofing (Suplantación de identidad).

- Descripción: Un atacante suplanta a un usuario legítimo o al propio chatbot.
  - Impacto potencial: Acceso indebido a datos sensibles, ejecución de acciones no autorizadas.
  - Mitigaciones aplicadas: Autenticación fuerte (MFA para admins), tokens con expiración corta, validación de identidad en cada request.
  - Decisión: Riesgo mitigado parcialmente; aceptado en escenarios de baja criticidad académica.
- Tampering (Manipulación de datos).
  - *Descripción*: Alteración de prompts, respuestas o datos almacenados.
  - *Impacto potencial*: Respuestas manipuladas, corrupción de datasets, decisiones erróneas.
  - *Mitigaciones aplicadas*: Validación estricta de entradas, control de versiones de datasets, logging de cambios.
  - *Decisión*: Riesgo mitigado con controles de aplicación; aceptado en entorno académico.
- Repudiation (Negación de acciones).
  - *Descripción*: Un usuario niega haber realizado una acción sin trazabilidad suficiente.
  - *Impacto potencial*: Dificultad en auditorías, falta de responsabilidad.
  - *Mitigaciones aplicadas*: Logging básico de accesos, auditoría de consultas al chatbot.
  - *Decisión*: Riesgo aceptado por limitación de plataforma; compensado con registros mínimos.
- Information Disclosure (Fugas de datos).
  - *Descripción*: Exposición de PII o información sensible en respuestas o logs.
  - *Impacto potencial*: Filtración de datos personales, sanciones legales.
  - *Mitigaciones aplicadas*: Filtros de salida, anonimización de datos, cifrado TLS 1.3.
  - *Decisión*: Riesgo mitigado, pero aceptado parcialmente por limitaciones de entorno académico.

- Denial of Service (Interrupción del servicio).
  - *Descripción:* Saturación de recursos del chatbot o infraestructura.
  - *Impacto potencial:* Caída del servicio, pérdida de disponibilidad.
  - *Mitigaciones aplicadas:* Rate limiting en endpoints críticos, monitorización de errores.
  - *Decisión:* Riesgo aceptado en entorno académico, mitigado en parte.
- Elevation of Privilege (Escalada de privilegios).
  - *Descripción:* Explotación de permisos mal configurados para obtener acceso indebido.
  - *Impacto potencial:* Control total de la aplicación o base de datos.
  - *Mitigaciones aplicadas:* Principio de mínimo privilegio, roles diferenciados, revisión de permisos DB.
  - *Decisión:* Riesgo mitigado con controles básicos; aceptado en alcance académico.

## 6.2 Amenazas por Componente

- Frontend
  - *Descripción:* Riesgos de XSS, CSRF y manipulación de inputs.
  - *Impacto potencial:* Robo de tokens, ejecución de scripts maliciosos.
  - *Mitigaciones aplicadas:* CSP, validación de entradas, HTTPS obligatorio.
  - *Decisión:* Riesgo mitigado parcialmente; aceptado en entorno académico.
- Backend
  - *Descripción:* Riesgos de inyección SQL y acceso no autorizado.
  - *Impacto potencial:* Exfiltración o corrupción de datos críticos.
  - *Mitigaciones aplicadas:* Prepared statements, autenticación robusta, WAF básico.

- *Decisión:* Riesgo mitigado, aceptado en alcance académico.
- Data Science / ML.
  - *Descripción:* Validación insuficiente de datasets y falta de control de versiones.
  - *Impacto potencial:* Modelos entrenados con datos corruptos, pérdida de integridad.
  - *Mitigaciones aplicadas:* Control de versiones, validación manual de datasets.
  - *Decisión:* Riesgo aceptado por limitación de recursos académicos.
- Infraestructura
  - *Descripción:* Configuraciones inseguras en IAM, falta de segmentación de red.
  - *Impacto potencial:* Acceso indebido a recursos sensibles.
  - *Mitigaciones aplicadas:* Principio de mínimo privilegio, subnets privadas, revisión manual de configuraciones.
  - *Decisión:* Riesgo aceptado por limitación de plataforma PaaS.

## 6.3 Amenazas Específicas de IA / Chatbots

- Prompt Injection.
  - Descripción: Manipulación de instrucciones para forzar respuestas indebidas.
  - Impacto potencial: Filtración de datos internos, ejecución de acciones no previstas.
  - Mitigaciones aplicadas: Mediador que filtra prompts, validación de contexto, auditoría de respuestas.
  - Decisión: Riesgo mitigado parcialmente; aceptado en entorno académico.
- Data Poisoning.
  - Descripción: Introducción de datos corruptos en entrenamiento o memoria del modelo.
  - Impacto potencial: Alteración del comportamiento del modelo, decisiones erróneas.

- Mitigaciones aplicadas: Validación manual de datasets, control de versiones.
  - Decisión: Riesgo aceptado por limitación de recursos.
- Model Inversion.
  - Descripción: Extracción de información sensible del modelo mediante consultas específicas.
  - Impacto potencial: Filtración de datos confidenciales.
  - Mitigaciones aplicadas: No incluir datos sensibles en entrenamiento, anonimización.
  - Decisión: Riesgo aceptado, mitigado parcialmente.
- Leakage de Información Sensible.
  - Descripción: Extracción de información sensible del modelo mediante consultas específicas.
  - Impacto potencial: Filtración de datos confidenciales.
  - Mitigaciones aplicadas: No incluir datos sensibles en entrenamiento, anonimización.
  - Decisión: Riesgo aceptado, mitigado parcialmente.

## 6.4 Escenarios de Ataque

Ejemplos narrativos de cómo un adversario podría explotar vulnerabilidades:

- Frontend: un atacante inyecta un script malicioso en un formulario para robar tokens de sesión.
- Backend: un usuario legítimo manipula parámetros en una API para acceder a datos de RRHH.
- Chatbot: un prompt diseñado para inducir al modelo a revelar credenciales internas.
- Infraestructura: IAM con privilegios excesivos que permiten acceso a buckets S3 con datos sensibles

## 6.5 Riesgos aceptados por limitaciones de plataforma

En el contexto de este proyecto, se ha optado por el uso de Render como plataforma PaaS para el despliegue del frontend y la base de datos, debido a su simplicidad operativa y adecuación a un entorno

académico. Esta decisión introduce limitaciones técnicas en materia de seguridad perimetral, que han sido identificadas, evaluadas y aceptadas de forma consciente.

- Riesgo 1: Ausencia de firewall perimetral configurable
  - *Descripción:* Render no permite configurar firewalls a nivel de red (IP filtering, reglas capa 3/4, WAF avanzado).
  - *Impacto potencial:* Exposición directa de endpoints públicos; mayor superficie frente a escaneos automáticos.
  - *Mitigaciones:* HTTPS obligatorio, autenticación/autorización en aplicación, rate limiting en endpoints críticos, validación estricta de entradas.
  - *Decisión:* Riesgo aceptado por limitación inherente, compensado con controles a nivel de aplicación.
- Riesgo 2: Imposibilidad de restringir acceso por IP entre servicios
  - *Descripción:* No es posible limitar acceso a BD/API solo a rangos internos o conocidos.
  - *Impacto potencial:* Dependencia total de credenciales; mayor impacto en caso de fuga de secretos.
  - *Mitigaciones:* Usuarios DB con permisos mínimos, gestión de secretos en variables de entorno, no exposición de credenciales en frontend/repos, rotación manual de secretos.
  - *Decisión:* Riesgo aceptado, documentado y asumible en el alcance académico.
- Riesgo 3: Dependencia de la seguridad gestionada por el proveedor
  - *Descripción:* El control sobre SO, parches y hardening recae en Render.
  - *Impacto potencial:* Falta de visibilidad sobre infraestructura; dependencia de políticas del proveedor.
  - *Mitigaciones:* Reducción de lógica sensible en frontend, encapsulación de acceso a datos en backend, diseño orientado a mínima exposición.

- *Decisión:* Riesgo aceptado, alineado con modelo de responsabilidad compartida.
- Riesgo 4: Limitaciones en monitorización y respuesta avanzada
  - *Descripción:* Render no ofrece detección avanzada de intrusiones ni respuesta automática.
  - *Impacto potencial:* Detección tardía de ataques complejos; respuesta manual ante incidentes.
  - *Mitigaciones:* Logging básico, monitorización de errores/latencias, alcance limitado y tiempo de exposición reducido.
  - *Decisión:* Riesgo aceptado, no crítico en entorno académico controlado.

## 6.5 Aplicación del OWASP Top 10 (2024)

Además de la metodología STRIDE y las amenazas específicas de IA, se ha considerado el marco OWASP Top 10 (2024) para validar la cobertura de riesgos en la plataforma. Este marco aporta una visión estandarizada de las vulnerabilidades más críticas en aplicaciones modernas, APIs y sistemas basados en IA.

### A01:2024 – Broken Access Control

- Añadir: Owner (Backend/DB), Prioridad: Alta.
- Implementación: RBAC a nivel API y DB; políticas IAM con least privilege; checks server-side en cada endpoint (no confiar en cliente).
- Pruebas/aceptación: pruebas de IDOR/authorization fuzzing, SAST rules para verificar checks de autorización.
- Detección: alertas por accesos a tablas sensibles fuera de horario o por cuentas no autorizadas.
- Ejemplo: no dar al servicio chatbot permisos SELECT sobre tablas RRHH; usar vistas controladas con columnas mínimas.

### A02:2024 – Cryptographic Failures

- Añadir: Owner (Infra/DevOps), Prioridad: Inmediata.

- Implementación: TLS 1.2+ (preferible 1.3), HSTS, certificados automatizados (ACME), RDS + S3 SSE-KMS, KMS key rotation policy.
- Contraseñas: bcrypt/argon2 con salt; nunca almacenar secretos en repos.
- Pruebas/aceptación: escaneo TLS (SSL Labs), verificación de cifrado en reposo en snapshots y backups.
- Métrica: % conexiones TLS, % volúmenes cifrados.

### A03:2024 – Injection (incluye SQLi y Prompt Injection)

- Añadir: Owner (Backend + ML Ops), Prioridad: Alta.
- SQL: usar prepared statements/ORM, parámetros, whitelist input validation, least-privilege DB user.
- Prompt injection: mediador entre UI y modelo, strip/escape de instrucciones meta, no incluir datos sensibles en contexto por defecto.
- Pruebas: DAST/DAST personalizado para prompt injection (fuzzing de prompts), pentest con casos de prompt injection.
- Detección: patrones en respuestas que contienen PII o comandos inesperados.

### A04:2024 – Insecure Design

- Añadir: Owner (Arquitectura), Prioridad: Alta.
- Implementación: threat modeling por feature, separación de datos sensibles, diseño por capas y principios de Zero Trust.
- Artefacto: diagrama de datos y flujos (qué puede ver el chatbot, qué no).
- Validación: revisión de diseño en PRs y gating de seguridad.

### A05:2024 – Security Misconfiguration

- Añadir: Owner (Infra), Prioridad: Inmediata.

- Implementación: IaC scanning (tfsec/checkov), políticas que bloqueen S3 público, Security Groups restrictivos, no exponer RDS.
- Pruebas: escaneo de configuraciones periódicas, guardrails con AWS Config / SCP en Org.
- Detección: alertas de Security Hub/GuardDuty por recursos públicos.

#### A06:2024 – Vulnerable and Outdated Components

- Añadir: Owner (DevSecOps), Prioridad: Media-Alta.
- Implementación: SCA (Dependabot/Snyk), bloqueo de merge si CVE críticos, imágenes de contenedor escaneadas y firmadas.
- Pruebas: builds que fallen por CVEs críticos; calendar de actualizaciones.
- Métrica: tiempo promedio de parcheo (days to patch).

#### A07:2024 – Identification and Authentication Failures

- Añadir: Owner (Auth), Prioridad: Inmediata.
- Implementación: JWT con algoritmo fuerte (RS256), expiraciones cortas, refresh tokens con revocación, MFA para admins, session management (revocación y logout global).
- Pruebas: atacar flujos de autenticación, brute-force detection, pruebas de token replay.
- Detección: múltiples intentos fallidos, tokens usados desde múltiples IPs.

#### A08:2024 – Software and Data Integrity Failures

- Añadir: Owner (DevOps/ML Ops), Prioridad: Media.
- Implementación: firmar artefactos y modelos (Sigstore/cosign), reproducible builds, checksums en despliegues, control de acceso a pipelines.

- Para ML: control de versiones del dataset y del modelo, registro de origen de datos, validación de integridad antes de uso.
- Pruebas: verificación de firmas antes de deploy; pruebas de integridad de modelos.

## A09:2024 – Security Logging and Monitoring Failures

- Añadir: Owner (SecOps), Prioridad: Inmediata.
- Implementación: centralizar logs (CloudWatch → SIEM), habilitar CloudTrail-data events para S3/RDS, audit logs de DB, retention definida.
- Detección: playbooks, alertas para accesos a tablas sensibles, detección de anomalías (baselining).
- Métrica: MTTD / MTTR objetivos, % servicios con logging habilitado.

## A10:2024 – Server-Side Request Forgery (SSRF) y API Abuse

- Añadir: Owner (Backend/Infra), Prioridad: Alta.
- Implementación: validar URLs, denylist de metadata/internal IPs (169.254.169.254), limitar egress, usar proxy de salida, quotas y rate limiting por API/key.
- Pruebas: DAST que incluya SSRF, pentesting de endpoints que aceptan URLs.
- Detección: requests salientes a 169.254.169.254 o a range internas, egress inusual.

## 7. Evaluación de Riesgos

La evaluación de riesgos combina el impacto potencial de cada amenaza con la probabilidad de ocurrencia, permitiendo priorizar mitigaciones y definir decisiones de aceptación o transferencia de riesgo. Se han considerado tanto los riesgos técnicos como organizativos, incluyendo los derivados de la plataforma Render y los componentes críticos de la aplicación (datos, endpoints, roles, integraciones externas).

## 7.1 Criterios de Evaluación

- Impacto: nivel de daño técnico, legal o de negocio en caso de materializarse.
- Probabilidad: facilidad o frecuencia con la que un atacante podría explotar la vulnerabilidad.
- Severidad: combinación de impacto y probabilidad, usada para priorizar mitigaciones.
- Decisión: aceptación, mitigación parcial o completa según el contexto académico del proyecto.

## 7.2 Matriz de Riesgos

Riesgo Identificado	Impacto	Probabilidad	Severidad	Mitigaciones Aplicadas	Decisión
Spoofing (Suplantación)	Alto	Medio	Alto	MFA, tokens con expiración	Mitigado parcialmente, Aceptado
Tampering (manipulación)	Alto	Medio	Alto	Validación entradas, control de versiones	Mitigado parcialmente, aceptado
Repudiation	Medio	Medio	Medio	Logging básico, auditoría	Riesgo aceptado
Information Disclosure	Alto	Alto	Crítico	Filtros de salida, cifrado TLS	Mitigado parcialmente
Denial of Service	Alto	Medio	Alto	Rate limiting, monitorización	Riesgo aceptado
Elevation of privilege	Alto	Bajo	Moderado	Principio de mínimo privilegio	Mitigado
Frontend(XSS, CSRF)	Alto	Medio	Alto	CSP, Validación entradas, HTTPS	Mitigado
Backend(SQL injection)	Alto	Alto	Crítico	Prepared statements, WAF, auth robusta	Mitigado

Data Science/ ML	Medio	Medio	Medio	Validación datasets, control versiones	Riesgo aceptado
Infraestructura (IAM inseguro)	Alto	Medio	Alto	Least privilege, revisión manual	Riesgo aceptado
Prompt injection	Alto	Alto	Crítico	Mediador de prompts, auditoría	Mitigado parcialmente
Data Poisoning	Alto	Bajo	Moderado	Validación datasets, control versiones	Riesgo aceptado
Model inversion	Alto	Medio	Alto	No incluir datos sensibles	Riesgo aceptado Parcialmente
Leakage de información	Alto	Alto	Crítico	Filtros de salida, Auditoría	Mitigado parcialmente
Firewall perimetral ausente (Render)	Alto	Alto	Crítico	HTTPS, auth, rate limiting	Riesgo aceptado
Restricción IP Imposible (Render)	Alto	Medio	Alto	Permisos mínimos DB, gestión secretos	Riesgo Aceptado
Monitorización limitada (Render)	Medio	Medio	Medio	Logging básico, monitorización errores	Riesgo aceptado
Datos sensibles (CUSTOMERS, EMPLOYEES, SALES)	Alto	Alto	Crítico	Roles mínimos, cifrado en reposo y tránsito	Riesgo mitigado parcialmente
JOINS y correlaciones entre tablas	Alto	Medio	Alto	Restricción de consultas, vistas controladas	Riesgo aceptado parcialmente
Endpoints y APIs expuestas	Alto	Alto	Crítico	Validación parámetros,	Riesgo mitigado

				rate limiting, API Gateway	
Integración con terceros (APIs externas)	Alto	Medio	Alto	Validación de respuestas, gestión segura de credenciales	Riesgo aceptado parcialmente
Logs con datos sensibles	Medio	Medio	Medio	Anonimización, control de acceso	Riesgo aceptado
Cumplimiento normativo (GDPR/LOPDGDD)	Alto	Medio	Alto	Políticas de borrado, consentimiento explícito, riesgo de tratamiento	Riesgo mitigado parcialmente

### 7.3 Riesgos Críticos Identificados

- SQL Injection y Prompt Injection.
- Fugas de datos sensibles (Information Disclosure / Leakage).
- Ausencia de firewall perimetral en Render.
- Exposición de endpoints públicos y correlaciones de tablas (JOINs).
- Cumplimiento normativo insuficiente (GDPR/LOPDGDD).

### 7.4 Priorización de Riesgos

- Alta prioridad: riesgos críticos con impacto alto y probabilidad alta (inyecciones, fugas de datos, firewall ausente, endpoints expuestos).
- Media prioridad: riesgos con impacto alto pero probabilidad moderada (model inversion, IAM inseguro, cumplimiento normativo).
- Baja prioridad: riesgos aceptados por limitaciones de plataforma, compensados con controles de aplicación (monitorización limitada, dependencia del proveedor, logs básicos).

## 8. Controles de Seguridad y Mitigaciones

### 8.1 Controles Existentes

- Autenticación y permisos
  - Tokens JWT con expiración corta y revocación.
  - MFA habilitado para cuentas administrativas.
  - Principio de mínimo privilegio aplicado en la base de datos.
- Cifrado
  - TLS 1.3 obligatorio en todos los endpoints.
  - AES-256 para cifrado en reposo en RDS y backups.
  - Gestión de claves mediante AWS KMS.
- Frontend
  - HTTPS + HSTS forzado.
  - Content Security Policy (CSP) aplicada.
  - Validación de inputs en formularios y sanitización de datos.
- Backend
  - Consultas parametrizadas (prepared statements).
  - WAF básico para filtrar tráfico malicioso.
  - Rate limiting en endpoints críticos.
- Monitoreo
  - Logging de accesos y errores.
  - Auditoría periódica de consultas en BD.

### 8.2 Mitigaciones Propuestas

- Segmentación de red
  - API del chatbot desplegada en VPC privada.
  - Endpoints internos accesibles únicamente desde servicios autorizados mediante Security Groups y VPC endpoints.
- Gestión de secretos
  - Variables de entorno seguras.
  - Rotación periódica de claves y tokens.
  - Eliminación de credenciales en repositorios.
- Hardening de infraestructura
  - IAM con políticas restrictivas y sin comodines (\*).
  - Desactivación de buckets S3 públicos.
  - Revisión de configuraciones con AWS Config y GuardDuty.
- Validación de prompts
  - Mediador entre chatbot y modelo para filtrar instrucciones.

- Sanitización de prompts y detección de PII.
  - Auditoría de respuestas generadas.
- Resiliencia
  - Rate limiting y cuotas en APIs.
  - Plan de recuperación ante incidentes con backups verificados.
  - Alta disponibilidad mediante despliegue multi-AZ.
- Monitorización avanzada
  - Integración con CloudWatch Logs y métricas.
  - Alertas en tiempo real para accesos sospechosos.
  - Correlación de eventos en SIEM.

### 8.3 Controles por Equipo

- Data Science
  - Validación de datasets antes de entrenamiento.
  - Control de versiones de modelos y datasets.
  - Registro de origen de datos.
- Full-Stack
  - Seguridad en APIs (auth, rate limiting, validación de parámetros).
  - Sanitización de inputs en frontend.
  - Uso de API Gateway para controlar exposición.
- Ciberseguridad
  - Gestión de accesos e IAM.
  - Detección de anomalías en logs.
  - Auditorías periódicas de roles y permisos.

### 8.4 Riesgos Residuales

- Prompt Injection avanzado: mitigado parcialmente, requiere vigilancia continua.
- Data Poisoning: riesgo residual por limitación de recursos académicos.
- Dependencia de seguridad gestionada por Render (si se mantiene parte del despliegue): falta de visibilidad sobre capa de infraestructura.
- Monitorización limitada: ausencia de respuesta automática ante incidentes complejos.

## 9. Cumplimiento y Consideraciones Legales

La plataforma maneja información sensible de clientes, empleados y transacciones, lo que implica cumplir con regulaciones de protección de datos y aplicar controles técnicos que garanticen la privacidad y la trazabilidad de la información.

### 9.1 Protección de Datos

- Regulación aplicable:
  - GDPR (Reglamento General de Protección de Datos) en el ámbito europeo.
  - LOPDGDD (Ley Orgánica de Protección de Datos y Garantía de Derechos Digitales) en España.
- Controles técnicos asociados:
  - Minimización de datos: solo almacenar y procesar la información estrictamente necesaria.
  - Cifrado en tránsito (TLS 1.3) y en reposo (AES-256).
  - Separación de roles y permisos para evitar accesos cruzados entre RRHH y Marketing.
  - Registro de actividades de tratamiento para demostrar cumplimiento.

### 9.2 Gestión de Logs y Privacidad

- Riesgos: los logs pueden contener identificadores de clientes, empleados o fragmentos de consultas sensibles.
- Controles técnicos:
  - Anonimización de datos en logs.
  - Restricción de acceso a logs mediante IAM y roles específicos.
  - Retención limitada en función de la necesidad operativa.
  - Monitorización de accesos a logs para detectar usos indebidos.

### 9.3 Retención y Anonimización de Datos

- Políticas de retención:
  - Definir tiempos máximos de almacenamiento para datos de clientes y empleados.
  - Aplicar borrado seguro en backups y snapshots antiguos.

- Anonimización:
  - Sustitución de identificadores directos (customer\_id, employee\_id) por pseudónimos en entornos de prueba.
  - Uso de vistas controladas para consultas del chatbot, evitando exposición de PII.
- Controles técnicos:
  - Automatización de procesos de borrado y anonimización.
  - Validación periódica de cumplimiento en entornos de desarrollo y producción.

## 10. Recomendaciones

### 10.1 Recomendaciones Técnicas

- Infraestructura:
  - Consolidar el despliegue en AWS con VPC privada, Security Groups restrictivos y endpoints internos.
  - Revisar periódicamente configuraciones con AWS Config y GuardDuty.
- Aplicación y API:
  - Implementar un API Gateway con validación de parámetros, rate limiting y control de exposición.
  - Mantener prepared statements y sanitización de inputs en backend.
- IA y Chatbot:
  - Fortalecer el mediador de prompts con reglas de filtrado más estrictas.
  - Monitorizar patrones de uso para detectar intentos de manipulación (Prompt Injection).
  - Validar datasets y aplicar control de versiones para reducir riesgo de Data Poisoning.
- Seguridad de datos:
  - Aplicar cifrado TLS 1.3 en tránsito y AES-256 en reposo.
  - Definir políticas de retención y anonimización de datos sensibles en logs y backups.
- Monitorización y respuesta:
- Integrar CloudWatch con SIEM para correlación avanzada de eventos.
- Definir alertas en tiempo real para accesos sospechosos y consultas anómalas

## 10.2 Recomendaciones Organizativas

- Gobernanza de accesos: definir roles claros entre RRHH, Marketing y Administradores.
- Gestión documental: mantener registro actualizado de riesgos aceptados y mitigaciones aplicadas.
- Formación de equipos: capacitar en seguridad de IA, gestión de datos sensibles y buenas prácticas de desarrollo seguro.
- Procesos de auditoría: establecer revisiones periódicas de permisos, logs y configuraciones.
- Cultura DevSecOps: integrar seguridad en el ciclo de vida del desarrollo, con validaciones automáticas en CI/CD.

## 10.3 Próximos Pasos de Seguridad

1. Migración completa a AWS privada para eliminar dependencias críticas de Render.
2. Implementación de controles avanzados de monitorización (SIEM, detección de anomalías).
3. Revisión periódica de roles y permisos en DB y IAM.
4. Simulación de ataques (Red Team / Pentesting) para validar efectividad de mitigaciones.
5. Documentación continua de riesgos aceptados y controles aplicados, asegurando reproducibilidad y trazabilidad.

## 11. Conclusiones

- El análisis realizado ha permitido identificar una superficie de ataque amplia, derivada de la combinación de IA conversacional, consultas dinámicas y relaciones complejas entre tablas de RRHH, Marketing y Ventas.
- Los riesgos más críticos se concentran en:
  - Inyecciones (SQL y Prompt Injection).
  - Fugas de datos sensibles (Information Disclosure / Leakage).
  - Exposición de endpoints públicos y ausencia de firewall perimetral en Render.
  - Cumplimiento normativo insuficiente (GDPR/LOPDGDD).
- La decisión de desplegar la API del chatbot en AWS como privada reduce significativamente la superficie de ataque,

- Permitiendo aplicar controles de red, IAM y segmentación que no eran posibles en Render.
- Persisten riesgos residuales como Prompt Injection avanzado, Data Poisoning y limitaciones de monitorización, que requieren vigilancia continua.

## 11.1 Resumen de Amenazas Clave

- Metodología STRIDE:
  - Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, Elevation of Privilege.
- Amenazas específicas de IA:
  - Prompt Injection, Data Poisoning, Model Inversion, Leakage de información sensible.
- Limitaciones de plataforma (Render):
  - Ausencia de firewall perimetral configurable.
  - Imposibilidad de restringir acceso por IP.
  - Dependencia de seguridad gestionada por el proveedor.
  - Monitorización limitada.
- Superficie de ataque identificada en componentes críticos:
- Endpoints expuestos, correlaciones de tablas (JOINs), logs con datos sensibles, integraciones externas y cumplimiento normativo.

## 11.2 Estado de Seguridad del Chatbot

- Madurez actual:
  - Se dispone de controles básicos (TLS 1.3, AES-256, roles mínimos, prepared statements, CSP, rate limiting).
  - La migración de la API a AWS privada mejora significativamente la postura de seguridad, reduciendo exposición pública y permitiendo segmentación de red.
- Fortalezas:
  - Principio de mínimo privilegio aplicado en DB e IAM.
  - Validación de entradas y sanitización de prompts.
  - Cifrado en tránsito y reposo.
- Debilidades:
  - Riesgo residual de Prompt Injection avanzado.
  - Monitorización aún limitada (sin correlación avanzada de eventos).
  - Cumplimiento normativo parcial (retención y anonimización de datos en proceso de implementación).
- Evaluación general:

- Nivel de madurez intermedio, con controles técnicos sólidos pero dependiente de mejoras en monitorización, resiliencia y cumplimiento.

## 11.3 Decisiones de Aceptación de Riesgo

- Riesgos mitigados:
  - SQL Injection (prepared statements, WAF).
  - Elevation of Privilege (roles mínimos, IAM restrictivo).
  - Cifrado en tránsito y reposo (TLS 1.3, AES-256).
- Riesgos aceptados:
  - Prompt Injection avanzado (difícil de eliminar completamente).
  - Data Poisoning en datasets (limitación de recursos académicos).
  - Dependencia de seguridad gestionada por Render (si parte del despliegue se mantiene).
  - Monitorización limitada (sin SIEM completo).
- Riesgos parcialmente mitigados:
- Leakage de información sensible (filtros de salida y auditoría, pero riesgo residual).
- Cumplimiento normativo (GDPR/LOPDGDD en proceso de implementación).

# 12. Anexos

## 12.1 Diagramas Detallados

- Arquitectura de la plataforma:
  - Diagrama de despliegue en AWS (VPC privada, API Gateway, RDS, IAM).
  - Componentes principales: frontend, backend, chatbot (IA), base de datos, servicios externos.
- Flujo de datos:
  - Usuario → Chatbot → LLM → Capa de generación de consultas → Base de datos → Respuesta reformulada → Usuario.
  - Inclusión de controles de seguridad en cada etapa (TLS, validación de entradas, roles mínimos).

## 12.2 Tabla Completa de Amenazas (STRIDE)

Componente	Spoofing	Tampering	Repudiation	Information Disclosure	DoS	Elevation of Privilege
Frontend	Suplantación de sesión	Manipulación de inputs	Negación de acciones	Exposición de datos en respuesta	Saturación de formularios	Escalada por XSS
Backend	Tokens falsos	Inyección SQL	Logs incompletos	Fugas en API	Saturación de endpoints	Roles mal configurados
Base de datos	Credenciales robadas	Alteración de registros	Falta de trazabilidad	Exposición de PII	Consultas masivas	Permisos excesivos
Chatbot/IA	Prompt Injection	Manipulación de contexto	Falta de auditoría	Leakage de información	Prompts pesados	Acceso indebido a datos
Infraestructura	IAM comprometido	Configuración insegura	Falta de logs	Buckets Públicos	Ataques de red	Privilegios excesivos

## 12.3 Glosario

- STRIDE: Metodología de clasificación de amenazas (Spoofing, Tampering, Repudiation, Information Disclosure, DoS, Elevation of Privilege).
- Prompt Injection: Técnica de ataque que manipula instrucciones de un modelo de IA para forzar respuestas indebidas.
- Data Poisoning: Inserción de datos maliciosos en datasets para alterar el comportamiento del modelo.
- IAM (Identity and Access Management): Servicio de AWS para gestionar usuarios, roles y permisos.
- GDPR/LOPDGDD: Regulaciones de protección de datos aplicables en Europa y España.
- API Gateway: Servicio de AWS que gestiona y protege el acceso a APIs.
- SIEM (Security Information and Event Management): Sistema de correlación y análisis de eventos de seguridad.

## 12.4 Referencias

- Normativas:
  - Reglamento General de Protección de Datos (GDPR).
  - Ley Orgánica de Protección de Datos y Garantía de Derechos Digitales (LOPDGDD).
- Frameworks:
  - OWASP Top 10 (2024).
  - NIST Cybersecurity Framework.
  - CIS Controls v8.
- Artículos técnicos:
  - Documentación oficial de AWS (IAM, VPC, API Gateway, RDS).
  - OWASP Cheat Sheets (SQL Injection, Authentication, Logging).
  - Publicaciones sobre seguridad en IA (Prompt Injection, Model Inversion, Data Poisoning).