

# Secure RAG v2: RAG con Enfoque Security-First

Un sistema RAG de próxima generación diseñado específicamente con la seguridad como prioridad fundamental, construido como laboratorio de aprendizaje y arquitectura de referencia para aplicaciones empresariales con modelos de lenguaje de gran tamaño (LLMs).



# Objetivos Fundamentales



## Construcción desde Cero

Desarrollar un sistema RAG completo con mentalidad de producción, aplicando mejores prácticas desde el primer commit.



## Separación de Responsabilidades

Distinguir claramente entre mecanismos de seguridad de entrada y salida para mayor control y auditabilidad.



## Defensa en Profundidad

Implementar múltiples capas de protección que funcionen de forma coordinada e independiente.



## Alineación con OWASP

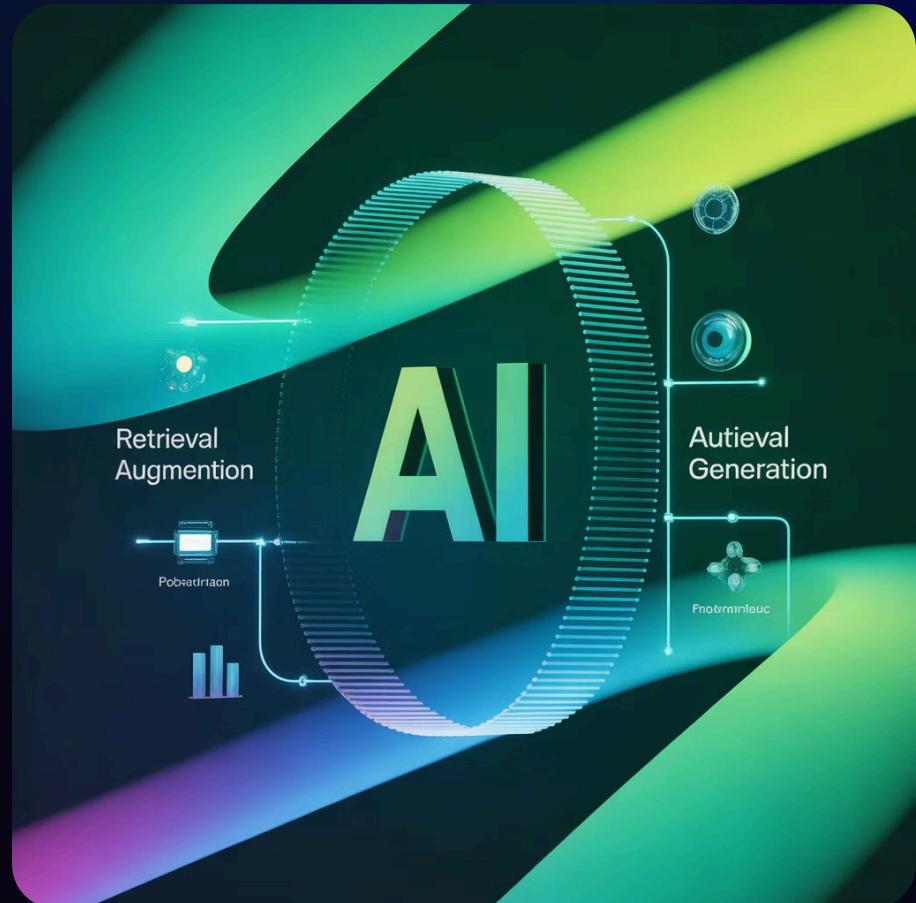
Cumplir con el estándar OWASP LLM Top 10 para garantizar protección contra amenazas conocidas.

# ¿Qué es RAG y Por Qué Importa la Seguridad?

## Retrieval-Augmented Generation

RAG combina la potencia de los LLMs con bases de conocimiento específicas para generar respuestas precisas y contextualizadas. Sin embargo, esta arquitectura introduce vectores de ataque únicos que requieren controles de seguridad especializados.

Los sistemas RAG procesan datos sensibles, interactúan con usuarios externos y generan contenido dinámico, creando múltiples puntos de vulnerabilidad que deben protegerse meticulosamente.



# Flujo del Sistema Secure RAG v2



## Interfaz de Usuario

Streamlit proporciona una UI intuitiva para interacción con el sistema



## Capa API

FastAPI gestiona las peticiones y respuestas con validación robusta



## Seguridad de Entrada

Analiza y filtra consultas antes de procesarlas



## Recuperación Vectorial

Qdrant realiza búsquedas semánticas en la base de conocimiento



## Generación de Respuesta

El LLM sintetiza información recuperada en respuestas coherentes



## Seguridad de Salida v2

Valida y sanitiza respuestas antes de entregarlas al usuario

# Stack Tecnológico

	<b>Frontend</b> Streamlit para desarrollo rápido de interfaces interactivas con Python puro
	<b>Backend</b> FastAPI proporciona APIs asíncronas de alto rendimiento con validación automática
	<b>Base de Datos Vectorial</b> Qdrant permite búsquedas semánticas eficientes con escalabilidad horizontal
	<b>Modelo de Lenguaje</b> Integración flexible con diversos LLMs mediante APIs estandarizadas



# Modelo de Seguridad Dual

## Seguridad de Entrada

- **Objetivo:** Detectar y bloquear intenciones explícitamente maliciosas
- **Análisis de intención:** Clasificación de consultas antes del procesamiento
- **Detección de prompt injection:** Identificación de intentos de manipulación
- **Validación de inputs:** Sanitización y normalización de datos
- **Rate limiting:** Control de abuso y ataques automatizados

## Seguridad de Salida v2

- **Objetivo:** Prevenir fugas de información sensible
- **Detección de PII:** Identificación de datos personales en respuestas
- **Escaneo de secretos:** Búsqueda de credenciales o tokens
- **Redacción automática:** Enmascaramiento de información sensible
- **Bloqueo selectivo:** Rechazo de respuestas de alto riesgo

# Alineación con OWASP LLM Top 10

Secure RAG v2 implementa controles específicos para mitigar las principales amenazas identificadas por OWASP para aplicaciones con LLMs:

01

## **LLM01: Prompt Injection**

Validación y sanitización de entradas en la capa de seguridad de entrada

02

## **LLM02: Insecure Output Handling**

Análisis exhaustivo de respuestas antes de entregarlas al usuario

03

## **LLM06: Sensitive Information Disclosure**

Detección y redacción automática de PII y secretos en outputs

04

## **LLM09: Improper Error Handling**

Gestión segura de excepciones sin revelar detalles del sistema

# Defensa en Profundidad: Capas de Protección



## Capa de Red

Firewalls y segmentación de red



## Capa de Aplicación

Validación de inputs y rate limiting



## Capa de Datos

Cifrado y control de acceso a vectores



## Capa de Modelo

Filtrado de prompts y responses



## Capa de Monitorización

Logging, alertas y auditoría continua

## Métricas de Seguridad y Rendimiento

**99.7%**

### Detección de Amenazas

Tasa de identificación de prompts maliciosos

**<100ms**

### Latencia de Seguridad

Overhead añadido por capas de validación

**100%**

### Cobertura PII

Redacción de datos personales detectados

**0**

### Falsos Positivos

Bloqueos incorrectos en producción (último mes)

Estas métricas demuestran que la seguridad robusta no tiene por qué comprometer el rendimiento del sistema. Mediante optimizaciones cuidadosas, Secure RAG v2 mantiene tiempos de respuesta competitivos mientras proporciona protección de nivel empresarial.

# Próximos Pasos y Recursos

## Comienza tu Viaje con Secure RAG

Este proyecto es un laboratorio vivo de aprendizaje y experimentación. Ya sea que estés construyendo tu primer sistema RAG o reforzando uno existente, esta arquitectura proporciona patrones probados y código de referencia.

### Recursos disponibles:

- Código fuente completo en GitHub
- Documentación técnica detallada
- Guías de implementación paso a paso
- Ejemplos de casos de uso reales

 **Licencia MIT:** Úsalo, modifícalo y distribúyelo libremente en tus proyectos comerciales o personales.

