PRUEBA DE ARQUITECTURA DE CORTEX

Arquitectura de Cortex

Ultima actualizacion: 20 de mayo de 2026

Como Cortex aplica aislamiento, auditabilidad y control soberano

Esta pagina describe como esta construido Cortex: el flujo de datos, el modelo de aislamiento, los limites de identidad y acceso, los controles de retencion y eliminacion, el registro de auditoria, las opciones de gobierno de modelos y los modos de despliegue disponibles para compradores que estan revisando Cortex para uso legal, SaaS regulado o empresarial.

Esta pensada como referencia de profundidad para equipos de seguridad, legal, cumplimiento y arquitectura durante una revision de proveedor. Las certificaciones especificas y los terminos contractuales se confirman por proyecto.

Flujo de datos en una pasada

Cada peticion que toca el contexto de un tenant sigue el mismo camino:

1. El llamante (una persona, un agente, un servicio interno o una aplicacion orientada al cliente) se autentica contra el limite de identidad de la plataforma.
2. La peticion se acota a un unico espacio de tenant. Cortex rechaza peticiones que no se puedan asociar a exactamente un tenant.
3. La recuperacion, el ensamblado del prompt y la invocacion de herramientas ocurren dentro de la Unidad de Computo Privada (PCU) de ese tenant. La memoria, los embeddings, los secretos y la politica del tenant viven dentro de esa unidad y nunca salen de ella para servir a otro tenant.
4. Las llamadas a modelo van al proveedor de modelo elegido para ese tenant o flujo, bajo la politica de gobierno configurada para ese espacio.
5. La llamada completa (llamante, tenant, contexto recuperado, prompt, modelo, respuesta, herramientas invocadas) queda registrada en el registro de auditoria por tenant antes de devolver la respuesta.

   El mismo camino se aplica tanto si Cortex se consume desde un copilot conversacional, desde un agente en background o desde un cliente programatico de API. No existe una ruta rapida que evite el acotado por tenant o la auditoria.

Modelo de aislamiento por tenant

Cortex se organiza alrededor de la Unidad de Computo Privada (PCU): un limite logico y

operativo que envuelve la memoria, la politica, las claves y la ejecucion de un tenant.

- Cada tenant tiene su propio espacio de memoria. Embeddings, documentos, estado previo de agentes e historial de conversacion estan particionados por identificador de tenant en la capa de almacenamiento, no por filtros a nivel de aplicacion.
- Cada tenant tiene sus propias claves de cifrado. Los datos en reposo se cifran con claves acotadas a ese tenant, asi que una clave filtrada no puede descifrar material de otro tenant.
- La recuperacion esta restringida en la capa de plataforma. Una consulta lanzada dentro de la PCU del tenant A no puede alcanzar el almacenamiento, los indices o las cachees del tenant B, incluso si un prompt o una herramienta intenta pedirlo.
- Las operaciones cruzadas entre tenants no son una capacidad de runtime. No existe una superficie de tipo "busca en todos los tenants" en el producto. Las agregaciones para operadores se calculan solo sobre metadatos, nunca sobre el contenido del tenant.

  Es el mismo patron productizado en la vertical legal, donde cada cliente o asunto recibe su propia PCU. Cortex generaliza ese patron mas alla de legal.

Limites de identidad y acceso

Cortex se mapea a los sistemas de identidad que las organizaciones ya operan, en lugar de introducir un directorio paralelo.

- Autenticacion: SSO contra el proveedor de identidad del cliente (OIDC o SAML), con soporte para cuentas de servicio e identidades de maquina para agentes y llamantes de backend.
- Autorizacion: acceso basado en roles y atributos, aplicado en el limite de la PCU. Los roles se mapean a roles organizativos existentes (por ejemplo, equipo de asunto, equipo de cliente, usuario interno de copilot, operador, auditor).
- Identidad del agente: cada agente y cada llamada de herramienta lleva una identidad verificable. Las acciones del agente son atribuibles al agente, a la persona en cuyo nombre actua y al tenant al que esta acotado.
- Acceso de operadores: los operadores de la plataforma no tienen acceso permanente al contenido del tenant. Existen procedimientos break-glass para respuesta a incidentes, requieren aprobacion explicita y quedan registrados en el registro de auditoria.
- Limite de red: las PCUs pueden exponerse solo a la red del cliente, a un conjunto definido de llamantes en lista blanca o a internet bajo la politica del cliente.

Controles de retencion y eliminacion

La retencion es una politica por tenant, no un valor por defecto global.

-

Ventanas de retencion configurables: memoria, transcripciones, fragmentos recuperados y salidas de herramientas pueden retenerse durante duraciones distintas, definidas por tenant y por flujo.

- Eliminacion fuerte: las solicitudes de eliminacion purgan los registros subyacentes, incluidos embeddings e indices derivados. La eliminacion es verificable en el registro de auditoria.
- Exportacion: los tenants pueden exportar su memoria, su registro de auditoria y su configuracion en un formato documentado, asi que nunca quedan atrapados en el producto.
- Retencion legal: cuando un tenant lo requiere, ciertos datos pueden quedar bajo retencion legal y exentos de la eliminacion por politica hasta que se libere la retencion.
- Copias de seguridad: las copias de seguridad siguen el mismo acotado por tenant y la misma politica de retencion que los datos en vivo, y se eliminan con el mismo calendario.

Registro de auditoria

El registro de auditoria es una parte de primer orden del producto, no un anadido.

- Alcance: cada prompt, recuperacion, accion de agente, invocacion de herramienta, llamada a modelo, cambio de configuracion y evento de acceso queda registrado.
- Granularidad: cada registro incluye quien actuo, en nombre de que tenant, contra que memoria, con que modelo, y con que politica vigente en ese momento.
- Evidencia de manipulacion: los registros son append only y estan encadenados, de modo que la modificacion posterior es detectable.
- Acceso: los tenants pueden consultar su propio registro desde el producto y via API. El registro es exportable para revision offline por legal, cumplimiento o auditoria interna.
- Defensibilidad: el registro esta disenado para responder a las preguntas que aparecen bajo revision regulatoria o en litigio, es decir, que sabia el agente, cuando lo supo y que hizo con ese conocimiento.

Opciones de gobierno de modelos

Cortex no ata al comprador a un unico modelo ni a un unico proveedor.

- Open weight dentro del limite: ejecuta modelos open weight dentro de la PCU del tenant, de manera que prompts y contexto nunca salen del limite. Adecuado para las cargas mas sensibles.
- Proveedores frontier bajo contrato empresarial: enruta a modelos frontier comerciales bajo contratos que incluyen retencion cero o retencion contractualmente acotada, sin entrenamiento con datos del cliente y con compromisos de procesamiento regional.
- Enrutamiento mixto por flujo: asigna modelos distintos a flujos distintos o a niveles de

sensibilidad distintos dentro del mismo tenant. La politica de enrutamiento forma parte de la configuracion del tenant y queda registrada en el registro de auditoria.

- Cambio de proveedor: cambiar de proveedor no requiere re-arquitectar el tenant. Memoria, politica y registro de auditoria son independientes del modelo en uso.

Modos de despliegue

Cortex soporta los modos de despliegue que los compradores regulados y soberanos esperan.

- Despliegue publico: Cortex multi tenant sobre infraestructura compartida, con las garantias de aislamiento por tenant descritas arriba. Adecuado para equipos sin requisitos duros de residencia o de tenant unico.
- Despliegue privado: Cortex de tenant unico sobre infraestructura dedicada gestionada por GREENPOW. Adecuado para organizaciones que necesitan un plano de control y un plano de datos dedicados.
- Region soberana: despliegue anclado a una jurisdiccion especifica, con procesamiento, almacenamiento y gestion de claves dentro de esa jurisdiccion. Adecuado para sector publico, finanzas reguladas, salud y otros mercados con requisitos duros de residencia.
- On premises o nube del cliente: Cortex desplegado dentro del propio centro de datos o cuenta de nube del cliente, con GREENPOW aportando el software y la guia operativa.

El modo de despliegue se elige por tenant. Un mismo cliente puede ejecutar tenants distintos en modos distintos si su cartera lo requiere.

Que es esta pagina, y que no es

Esta pagina describe la arquitectura sobre la que esta disenado Cortex. Es un punto de partida para una revision de proveedor, no un sustituto. Los compradores en alcance para un despliegue de Cortex reciben:

- Una revision detallada de arquitectura y seguridad bajo NDA.
- Un recorrido del modo de despliegue relevante y de los controles que se le aplican.
- Confirmacion de las certificaciones, terminos contractuales y compromisos operativos que aplican al proyecto especifico.

Para iniciar esa conversacion, visita la pagina de producto de Cortex o contacta directamente con GREENPOW.

Fuente: greenpow.com/es/products/cortex/architecture

Fuente: greenpow.com/es/products/cortex/architecture

Fuente: greenpow.com/es/products/cortex/architecture

Fuente: greenpow.com/es/products/cortex/architecture

Fuente: greenpow.com/es/products/cortex/architecture