innovaycree.com/blog · documento generado del artículo en línea; la versión vigente es la URL canónica.

SEGURIDAD DE IA · Nº 04 · V1.0

Cómo instalar IA segura en una empresa: 7 decisiones

> Instalar IA de forma segura en una empresa no se resuelve eligiendo modelo, sino tomando siete decisiones de arquitectura: dónde vive el dato mientras el modelo trabaja, qué se hace con lo enviado, quién ve qué, qué puede ejecutar el sistema sin permiso, qué queda registrado, cómo se corta y qué se exige por contrato. Ninguna depende del modelo y todas se pueden comprobar antes de firmar.

Por **David Pinto** — Fundador y arquitecto de la plataforma · **LinkedIn**

Revisión técnica: Innova y Cree

Publicado 29 ago 2026 · 10 min de lectura · PDF

Instalar inteligencia artificial de forma segura en una empresa no se resuelve eligiendo el modelo correcto. El modelo es la pieza que se cambia; lo que decide el riesgo es la capa que lo rodea. Son **siete decisiones de arquitectura**, todas anteriores a la elección de proveedor y todas comprobables antes de firmar: dónde vive el dato mientras el modelo trabaja, qué se hace con lo que se envía, quién ve qué, qué puede ejecutar el sistema sin permiso, qué queda registrado, cómo se corta y qué se exige por contrato. Este artículo las desarrolla una a una, con lo que hay que pedir para verificar cada una. Es el paso previo a los seis controles que exigir a un agente en producción.

Qué decide de verdad si una instalación de IA es segura

Lo que decide la seguridad de una instalación de IA es el **conjunto de límites que rodean al modelo**, no la calidad del modelo. Dos empresas pueden usar exactamente el mismo motor y tener niveles de exposición incomparables: una lo consulta desde una cuenta compartida, con acceso a la unidad de red completa y sin registro; la otra le concede permisos por herramienta, aísla el dato por área y guarda quién preguntó qué.

Esto no es una opinión de diseño: es la razón de que los catálogos de riesgo de IA describan ataques que no tocan el modelo en absoluto —envenenamiento del contexto, extracción por consulta, abuso de permisos heredados— sino la tubería que lo alimenta[2] [6]. La consecuencia práctica para un comprador es cómoda: las siete decisiones se pueden tomar sin entender de redes neuronales, y siguen valiendo cuando el proveedor cambie de modelo el año que viene.

Decisión 1 · Dónde vive el dato mientras el modelo trabaja

Hay tres respuestas posibles y la correcta la fija **el dato más sensible que el sistema va a tocar**, no la preferencia del área técnica.

En un **servicio gestionado** el dato sale de la empresa hacia la infraestructura del proveedor del modelo, se procesa y vuelve. Es lo más rápido de poner en marcha y lo que menos carga operativa deja. En una **nube privada** el modelo corre en infraestructura contratada por la empresa: el dato no se mezcla con el de terceros y la empresa controla la región. En un **servidor propio** el dato no cruza nunca el borde de la organización.

La pregunta que ordena la decisión no es «¿cuál es más seguro?» sino **«¿qué pasa si este dato concreto aparece fuera?»**. Un histórico de tickets y un expediente laboral no merecen la misma arquitectura, y casi ninguna empresa necesita la misma para todo: lo habitual y lo sensato es mezclar.

Decisión 2 · Qué se hace con lo enviado: entrenamiento, retención y revisión

Aquí se confunden tres cosas distintas y hay que exigirlas **por separado y por escrito**:

- **Entrenamiento**: si lo enviado se usa para mejorar el modelo. La respuesta que sirve es un no contractual, no una casilla en un panel.
- **Retención operativa**: cuántos días se guarda lo enviado para que el servicio funcione —reintentos, depuración, facturación—. Siempre hay alguna; lo que importa es que esté acotada en días y que se pueda pedir su borrado.
- **Revisión humana**: si personas del proveedor pueden leer contenido para detectar abuso, bajo qué condiciones y con qué trazabilidad.

Un proveedor serio declara las tres con números. Uno que responde «sus datos están seguros» a la pregunta de retención no está tranquilizando: está evitando la pregunta. Esto es materia de contrato, no de arquitectura, y por eso reaparece en la decisión 7.

> AVISO
>
> Este artículo trata la arquitectura, no la ley aplicable. La normativa de protección de datos que obliga a cada empresa depende de su país y del país donde se procese la información, y es la que fija plazos, bases de licitud y obligaciones de notificación. Antes de definir la retención conviene contrastarla con asesoría legal de la jurisdicción correspondiente.

Decisión 3 · Quién ve qué: aislamiento entre áreas y entre empresas

El aislamiento se diseña **antes** de cargar el primer documento, porque rehacerlo después obliga a reconstruir el índice entero. La regla es que el filtro de acceso se aplique **en la recuperación**, no en la respuesta: el sistema no debe recuperar un fragmento que el usuario no podría abrir y luego decidir no mostrarlo. Filtrar al final es una promesa; filtrar al recuperar es un control.

En instalaciones que sirven a varias empresas o a varias áreas, cada fragmento almacenado lleva su etiqueta de pertenencia y toda consulta la lleva también, sin excepción ni modo administrador que la desactive. La prueba de que existe es simple y hay que pedirla: **una consulta hecha con las credenciales de un área que devuelva cero resultados sobre un documento de otra**, no una explicación de cómo funciona.

Decisión 4 · Qué puede hacer el sistema sin permiso

Un sistema que solo responde texto y uno que además ejecuta acciones pertenecen a categorías de riesgo distintas. En cuanto el sistema puede escribir en una base, enviar un correo o mover un registro, los permisos dejan de ser un detalle de configuración.

Los permisos se conceden **por herramienta y por alcance de dato**, nunca «al sistema» en bloque, y los parámetros se validan fuera del modelo. Una instrucción en el prompt pidiendo que no haga algo peligroso es una preferencia y se puede sobrescribir con texto suficientemente persuasivo; el límite real vive en la capa que concede el permiso y valida la llamada [5].

Aquí entra el ataque que más sorprende a quien lo ve por primera vez: la **inyección indirecta de instrucciones**. Todo lo que el sistema lee para trabajar —un correo, un PDF adjunto, una página— es entrada no confiable, porque puede traer instrucciones

dirigidas al modelo[4]. Si además tiene permisos amplios, el atacante no necesita entrar en la red: le basta con que alguien le pase el documento.

Decisión 5 · Qué queda registrado, y durante cuánto

El registro tiene que permitir **reconstruir una decisión concreta meses después**. En la práctica eso significa guardar quién pidió qué, qué contexto se recuperó, qué herramienta se invocó con qué parámetros, qué devolvió, qué decidió el sistema y quién aprobó si hubo aprobación —todo con marca de tiempo y con **la versión del modelo y de la instrucción vigentes en ese momento**.

Esa última parte es la que casi siempre falta y la que vuelve inútil al resto: sin saber con qué versión de instrucción respondió, el registro cuenta qué pasó pero no permite explicar por qué, que es justo lo que se necesita cuando alguien reclama. Enfoques de verificación por petición como los de una arquitectura de confianza cero encajan de forma natural aquí: cada llamada se autoriza y se anota, en lugar de confiar en que quien está dentro del perímetro puede[3].

Decisión 6 · Cómo se apaga sin apagar la empresa

Un interruptor de parada no es apagar el servidor. Es poder **desactivar una herramienta o una función concreta sin derribar el servicio**, dejando el sistema en modo degradado —consulta y lectura, sin acciones— mientras se investiga.

Dos condiciones lo separan de un adorno. La primera: el corte tiene que estar **al alcance del cliente**, no solo del proveedor, porque el incidente puede ocurrir un domingo. La segunda: tiene que haberse **ejercitado en simulacro**. Un mecanismo de emergencia que nunca se probó no se sabe si funciona, y el día que hace falta no es el día de averiguarlo. Pedir la fecha del último simulacro es una de las preguntas más reveladoras que se le pueden hacer a un proveedor.

Decisión 7 · Qué se exige por contrato

Lo que no está por escrito no es exigible, por muy convincente que fuera la reunión. El contrato debe recoger, como mínimo: la **no utilización para entrenamiento**; el **plazo de retención** en días y el procedimiento de borrado; la **región de procesamiento**; el régimen de **subencargados** —quién más toca el dato, porque el proveedor del modelo rara vez es el único—; la **notificación de incidentes** con plazo; el **derecho a auditar** o, en su defecto, a recibir evidencia periódica; y la **portabilidad**: qué se lleva la empresa si se va, en qué formato y en cuánto tiempo.

La cláusula que más se olvida es la última. Una instalación de la que no se puede salir con los datos y los índices propios no es un proveedor: es una dependencia. Estas exigencias son las mismas que ordenan la conversación en cómo evaluar a un proveedor de IA.

Las tres arquitecturas, comparadas

	SERVICIO GESTIONADO	NUBE PRIVADA	SERVIDOR PROPIO
Dónde se procesa	Infraestructura del proveedor	Infraestructura contratada por la empresa	Dentro de la organización
El dato cruza el borde	Sí	Sí, a un entorno dedicado	No
Puesta en marcha	Días	Semanas	Semanas o meses
Carga operativa continua	Del proveedor	Compartida	De la empresa: parcheo, copias, vigilancia

	SERVICIO GESTIONADO	NUBE PRIVADA	SERVIDOR PROPIO
Control de región	Según el contrato	Alto	Total
Encaja bien con	Datos internos de sensibilidad media, volumen variable	Datos regulados con volumen sostenido	Secreto industrial, restricción de salida del dato
Riesgo principal	Contrato débil o subencargados no declarados	Configuración de aislamiento mal hecha	Mantenimiento que se abandona con el tiempo

las tres formas de instalar IA en una empresa, con el riesgo característico de cada una.

Cuándo la IA en servidor propio compensa de verdad

El servidor propio compensa cuando **el dato no puede salir** —por secreto industrial, por una obligación específica del sector o por una decisión de riesgo tomada arriba— y cuando existe alguien que se hará cargo del mantenimiento durante años, no solo de la instalación.

Cuando la razón para elegirlo es otra —desconfianza genérica, o la sensación de que «dentro» es sinónimo de seguro— suele salir mal por una vía predecible: el sistema se instala, funciona, y dieciocho meses después nadie ha aplicado un parche, las copias no se han restaurado nunca y el aislamiento entre áreas quedó pendiente para la fase dos. Un servidor propio desatendido concentra el riesgo en lugar de repartirlo.

La decisión honesta es por dato, no por empresa: lo sensible dentro, lo demás fuera, con el mismo conjunto de siete decisiones aplicado a ambos.

Los tres errores que más vemos

El primero es **elegir el modelo antes que la arquitectura**. La conversación empieza por cuál es mejor y termina seis semanas después descubriendo que el dato no podía salir del país. Las siete decisiones son anteriores y no cambian al cambiar de motor.

El segundo es **confundir el piloto con la instalación**. Un piloto con tres personas y datos de prueba no ejercita el aislamiento, ni los permisos, ni el corte. Todo lo que este artículo describe empieza a existir cuando entra el primer dato real de un tercero.

El tercero es **prohibir en vez de ordenar**. Cuando una empresa prohíbe la IA sin ofrecer una vía autorizada, el uso no desaparece: se vuelve invisible, en cuentas personales y fuera de todo registro. El inventario honesto de lo que ya se usa suele ser el primer entregable útil de cualquier instalación seria, y es el mismo ejercicio que ordena el inventario de componentes de IA.

En Innova y Cree operamos 12 plataformas con IA en producción, y estas siete decisiones son las que tomamos en cada una antes de escribir la primera línea. Lo que publicamos sobre nuestra propia tasa de error y cómo la medimos está en el benchmark público, con su método y sus límites.

Puntos clave

01 Instalar IA segura es una decisión de arquitectura, no de modelo: las siete decisiones se toman antes de elegir proveedor y sobreviven a cambiarlo.

02 Dónde vive el dato admite tres respuestas —servicio gestionado, nube privada y servidor propio— y la correcta depende del dato más sensible que va a tocar, no de la preferencia del área técnica.

03 Entrenamiento, retención operativa y revisión humana son tres cosas distintas: se exigen por escrito y por separado, con plazos en días.

04 Los permisos se conceden por herramienta y por alcance de dato, y se validan fuera del modelo: una instrucción en el prompt no es un control de acceso.

05 Lo que no se puede enseñar en una sesión técnica —un registro reconstruible y un corte ejercitado— no está instalado, está prometido.

Preguntas frecuentes

¿Instalar IA segura es elegir el modelo correcto?

¿Los datos que enviamos a un modelo entrenan al modelo?

¿Es más seguro instalar la IA en un servidor propio?

¿Qué es la inyección indirecta de instrucciones y por qué afecta a la seguridad del dato?

¿Se puede comprobar todo esto antes de firmar?

¿Por dónde se empieza si la empresa ya está usando IA sin control?

Referencias

[1] NIST AI Risk Management Framework 1.0 — funciones GOVERN, MAP, MEASURE y MANAGE. https://www.nist.gov/itl/ai-risk-management-framework. Consultado el 29 de agosto de 2026. NORMA

[2] NIST AI 100-2 — taxonomía de ataques y mitigaciones en aprendizaje automático adversario. https://csrc.nist.gov/pubs/ai/100/2/e2025/final. Consultado el 29 de agosto de 2026. NORMA

[3] NIST SP 800-207 — Zero Trust Architecture (verificación por petición, no por perímetro). https://csrc.nist.gov/pubs/sp/800/207/final. Consultado el 29 de agosto de 2026. NORMA

[4] OWASP GenAI — LLM Top 10 (edición 2026): inyección de instrucciones, fuga de datos y cadena de suministro. https://genai.owasp.org/resource/owasp-genai-llm-top-10-2026/. Consultado el 29 de agosto de 2026. NORMA

[5] OWASP GenAI — Top 10 for Agentic Applications (2026). https://genai.owasp.org/resource/owasp-top-10-for-agentic-applications-for-2026/. Consultado el 29 de agosto de 2026. NORMA

[6] MITRE ATLAS — matriz de tácticas y técnicas adversarias contra sistemas de inteligencia artificial. https://atlas.mitre.org/. Consultado el 29 de agosto de 2026. NORMA

Citar este artículo

APA

```
Pinto, D. (2026, 29 de agosto). Cómo instalar IA segura en una empresa: 7
decisiones (v1.0). Blog técnico de Innova y Cree.
https://innovaycree.com/blog/seguridad-ia/instalar-ia-segura-empresa.html
```

BIBTEX

```
@misc{pinto2026instalar,
  author       = {Pinto, David},
  title        = {Cómo instalar IA segura en una empresa: 7 decisiones},
  howpublished = {Blog técnico de Innova y Cree},
  year         = {2026},
  month        = {ago},
  note         = {v1.0, revisado 2026-08-29},
  url          = {https://innovaycree.com/blog/seguridad-ia/instalar-ia-segura-
empresa.html}
}
```

v1.0 · 29 ago 2026

▶ HISTORIAL DE VERSIONES

SOBRE EL AUTOR

David Pinto · Fundador y arquitecto de la plataforma

Diseña la arquitectura de IA verificable de Innova y Cree: RAG con cita verificada contra la fuente oficial, benchmark anti-alucinación público y plataformas en producción para sectores regulados de habla hispana.

LinkedIn · Todos sus artículos