PROCUREMENT & SECURITY REVIEW REFERENCE

Privian Blueprint

A practical guide for evaluating how Privian handles prompts, provider credentials, retention, masking, and data flow.

Document	Privian Blueprint
Version	v1.0
Generated	2026-06-10
Canonical	https://privian.io/blueprint
Scope	Implementation-grounded reference for security and procurement reviews

Everything in this document is grounded in the current Privian implementation. No invented certifications, customers, or metrics.

1. Executive summary

Privian is a managed gateway that sits between your application and an LLM provider. It detects supported sensitive entities in prompts, replaces them with placeholders before the prompt reaches the model, and rehydrates the placeholders in the model's response. The provider credential is yours (BYOK).

Privian helps you

- Reduce prompt-level exposure of supported sensitive entity types.
- Mask supported entities before prompts reach the model.
- Route requests to your own provider account under BYOK.
- Reduce the risk of accidental disclosure through model prompts.

Privian does not

- Provide self-hosted inference.
- Provide a governance or policy-management program.
- Provide compliance certification (HIPAA, PCI, SOC 2, ISO).
- Prevent prompt injection or jailbreak attacks.
- Control downstream provider behavior, retention, or training policies.

Who it is for

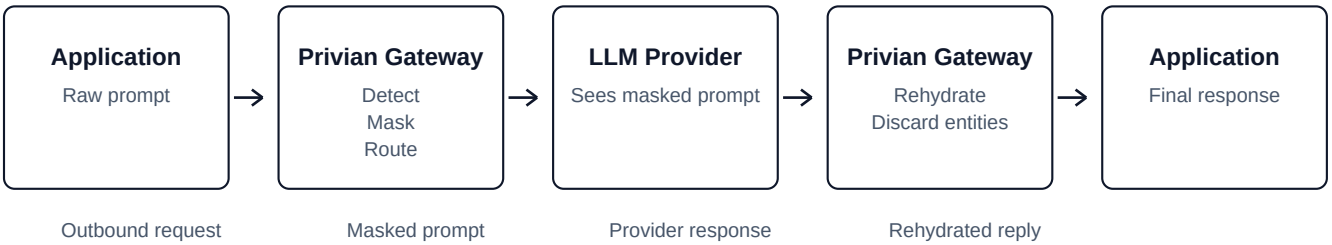
Teams that already use a managed LLM provider, want to reduce prompt-level exposure of supported entity types, and prefer to keep their own provider account and billing relationship.

Who it is not for

Teams that require fully self-hosted inference, formal compliance attestations today, or a behavioral DLP product that prevents users from sharing information they intend to share.

2. Clean data path

Supported entities are detected in-process and replaced with placeholders before any outbound provider call. The provider sees masked text only. The placeholder map exists for the lifetime of a single request and is discarded after the response is rehydrated.



In memory only (per request)	Never persisted
Entity map / placeholder bindings	Raw prompt bodies
Decrypted BYOK provider credential	Rehydrated response bodies

- Masked prompt is what reaches the model.
- Supported entities are stored in-memory only, per request.
- Raw prompt bodies are not persisted.
- Raw / rehydrated response bodies are not persisted.
- BYOK is supported across OpenAI, Anthropic, Google (Gemini), and DeepSeek.

3. What reaches the model

The table below describes which data classes cross the BYOK boundary to the LLM provider.

Data type	Reaches the model?
Masked prompt text	Yes — this is the substrate the provider receives.
Placeholder tokens (e.g. PERSON_1, EMAIL_1)	Yes — embedded inside the masked prompt.
Provider-side instructions (system, model, params)	Yes — forwarded to the provider as supplied.
Raw values of supported sensitive entities	No — replaced with placeholders before the outbound call.
Decrypted BYOK provider credential	Used to authenticate the outbound request; not sent as content.
Privian gateway API key	No — terminates at the Privian gateway.
Internal gateway secrets, signing keys	No — never sent to providers.
Entity map (placeholder → original value)	No — in-memory only; discarded after rehydration.

4. Retention model

Persistence is intentionally narrow. Operational records exist to run the service, bill it, and surface aggregate usage. Prompt and response bodies are not part of that record.

Stored

Class	Detail
Account, billing, team metadata	Required to operate the service and bill correctly.
BYOK provider credentials	Encrypted at rest with AES-256-GCM; decrypted in-process per request.
Privian gateway API keys	Shown once at creation; stored as SHA-256 hash.
Usage rollups	Token counts, request counts, latency aggregates.
Sanitized observability events	Status, model, route, timing. No prompt or response bodies.

Not stored

Class	Detail
Raw prompt bodies	Discarded after the outbound provider call completes.
Rehydrated response bodies	Returned to the caller, not persisted.
Per-request entity map	Lives in process memory; cleared after rehydration.
Decrypted provider credentials	Held in memory for the duration of one request; never written to disk.

5. BYOK — bring your own provider key

BYOK is the default Privian model. You provide a credential for the provider you want to use; Privian authenticates the outbound call with that credential.

Credential ownership

Your provider credential belongs to your account at the provider. Revocation, rotation, and provider-side billing remain under your control.

Provider billing

Provider usage is billed by the provider to your provider account. Privian's invoice covers the gateway service only.

What Privian stores

- Credential ciphertext, encrypted at rest with AES-256-GCM.
- Last-4 digits and a label for identification in the dashboard.

What Privian does not store

- Plaintext provider credentials.
- Long-lived decrypted keys outside of a single request.

6. Subprocessors

Privian uses a small set of operational subprocessors (hosting and managed database, payment processor, transactional email). When you configure a BYOK provider, that provider becomes part of your data flow — but you select it and authenticate it with your own credential.

The canonical, up-to-date list is published at privian.io/resources/subprocessors. Material changes are notified in accordance with the executed DPA where applicable.

7. Security review questions

What reaches the model?

The masked prompt. Supported entities are replaced with placeholders such as PERSON_1 or EMAIL_1 before any outbound provider call.

Who owns API keys?

You. Privian gateway API keys are shown once at creation and stored only as SHA-256 hashes. BYOK provider credentials are yours and are encrypted at rest.

Do you train models?

No. Privian does not train models on customer prompts or responses. Detection uses deterministic in-process detectors.

What is retained?

Account, billing and team metadata; encrypted BYOK credentials; hashed gateway API keys; usage rollups; sanitized observability events.

Can I self-host?

Not today. Privian is operated as a managed gateway.

Can I use OpenAI directly?

Yes. Provide your OpenAI key and the outbound call is authenticated with your credential. Provider billing and terms apply between you and OpenAI.

Can I use Anthropic directly?

Yes. The BYOK model is the same for Anthropic, Google (Gemini), and DeepSeek.

Does Privian support regional residency?

Region selection is not a contracted feature today. Provider-side region behavior is governed by your relationship with the provider.

Can Privian stop employees from pasting sensitive data?

Privian masks supported entity types before prompts reach the provider, reducing prompt-level exposure. It is not a behavioral DLP product and does not prevent a user from intentionally bypassing the gateway.

What does Privian actually protect?

The prompt path. Supported sensitive values are removed from outbound prompts and restored only in the response returned to your application.

8. Limitations and trust boundaries

Privian's scope is intentionally narrow. Knowing what is out of scope is as important to a security review as knowing what is in scope.

What Privian does not claim

- No HIPAA, PCI, SOC 2, or ISO certification.
- No guarantee of detection for entities outside the supported set.
- No prevention of prompt injection or model jailbreaks.
- No control over provider-side retention, logging, or training policy.

Customer responsibilities

- Selecting which provider(s) to use and accepting their terms.
- Managing rotation and revocation of BYOK credentials.
- Governing how end-users interact with the application that calls Privian.
- Determining whether masking coverage matches the data classes you handle.

Provider responsibilities

- Processing the masked prompt under the provider's own terms.
- Provider-side logging, retention, and training opt-outs.
- Regional behavior and availability.
- Billing and rate limits on the provider account.

For the canonical web version of this document, see **privian.io/blueprint**. The web version is the source of truth; this PDF is intended for internal distribution during a security review.