CloudWinks

CLOUDWINKS · FOUNDATION APPS

KeyKook

Technical brief

KeyKook brings sign-in, access permissions and protected credentials into one shared service. It gives applications a controlled way to obtain the access they need, reducing scattered key stores and making security responsibilities clearer.

1. Architecture in brief

KeyKook has two related responsibilities: identity/access and a server-secret vault. Server-only secrets remain separate from user credential records. KeyKookSecretProtector uses AES-256-GCM encryption; KeyKookSecretResolver applies application/company grants, expiry checks and audit handling around secret resolution.

2. Service boundary and intake

Apps request server secrets through the common resolver rather than maintain their own stores. Password sign-in issues a platform JWT for protected platform calls. Provider OAuth code exchange and refresh run server-side, keeping provider client secrets out of Flutter. Provider access tokens and platform JWTs have distinct purposes.

3. Data handled

The service holds encrypted secret values and version metadata, application/company grants, audit context, user credential tokens and access/session records. Bootstrap encryption material stays in host configuration. Server-secret resolution is a service capability, not an ordinary client list/read operation. The vault also holds personal identifiers by type (such as SSN, RUC and CI), bank numbers and tax ids, and the field-encryption keys used for Zibit underwriting fields and DreamDrafter personal financial statements; apps store a vault reference and a masked last four.

4. Integrations and ownership

BigTalk resolves messaging-provider credentials through KeyKook. CosmosCall resolves signature-verification secrets through the same service. KeyKook connected accounts are the only connection record for outside services: CosmosCall points to them, each belongs to a person or a company, and other people and companies cannot see them. Crux supplies base identity, while the platform uses KeyKook access services. The standalone product design extends this common credential responsibility beyond the surrounding business apps.

5. Security and access

Encryption protects stored values; authorization and grant checks control their use. The server separates provider identity credentials from its own authenticated platform sessions. Key recovery, rotation, revocation timing and audit-failure handling are important parts of operating the vault. The product’s documented rotation policy calls for a 45-day credential cycle. Reading an unmasked vault value requires a KeyKook role and is written to the calling app’s append-only reveal log. Payroll roles require MFA, and step-up MFA protects Zibit payments above a threshold.

Competitive context

ALTERNATIVE	ITS FOCUS	CLOUDWINKS PRODUCT FOCUS
HashiCorp Vault	Secret storage, access policies and audit capabilities.	Credential access closely connected to CloudWinks app permissions.
Auth0	Sign-in and identity-access capabilities.	A shared identity/access and credential-service role across the app family.

© 2026 CloudWinks.