04. Secrets and API keys policy

> **Template.** Part of the open-source AI App Policy Kit, version 1.0. Replace everything in [square brackets], delete what does not apply, and have the person who runs the business approve it. A starting point, not legal advice.

Business	[Business name]
Document owner	[Name and role]
Approved by, and date	[Name], [date]
Next review	[Date, at most 12 months on]

Why this matters

A secret is anything that lets software in: a password, an API key, a token, a private key. AI coding tools often paste keys straight into code, and code is shared, copied and published. A leaked AI key can also run up a large bill in hours.

Who it covers

Everyone who writes, deploys or configures software for [Business name], and every tool that does so on their behalf.

The rules

1. **No secrets in code.** Not in source files, not in comments, not in example files, not in the app that runs in a customer's browser or phone.
2. **Keep secrets in a secrets store.** Use [the cloud provider's secrets manager, or the hosting platform's encrypted environment settings]. The app reads them when it runs.
3. **Browser and mobile code holds no secret.** Anything shipped to a user's device can be read. Calls that need a key go through your own server.
4. **Scan for secrets.** Turn on secret scanning and push protection on the code host. A commit that contains a key is blocked before it lands.
5. **Different keys for different places.** Development, test and production each have their own keys. A developer's laptop never holds a production key.
6. **Narrow keys.** Give each key only the rights it needs, and restrict it by address or service where the supplier allows.
7. **Cap the spend.** Set a budget or usage limit and an alert on every AI and cloud key.
8. **Rotate.** Replace keys every [6 or 12] months, when someone with access leaves, and at once if a key may have leaked.
9. **A leaked key is an incident.** Revoke it first, then find out what it was used for (document 06). Deleting the commit is not enough: the key is in the history and may already be copied.
10. **Never paste secrets into AI tools, chats, tickets or email.** Share them through the password manager.

Secrets register

Record that each secret exists and where it is kept. Never write the secret itself here.

Secret (name only)	What it opens	Where it is stored	Who can read it	Spend limit	Last rotated
[AI supplier production key]	[AI model calls]	[Secrets manager]	[Names]	[£ per month]	[Date]
[Database password]	[ ]	[ ]	[ ]	[n/a]	[ ]
[Payment provider key]	[ ]	[ ]	[ ]	[n/a]	[ ]

Checklist

- ☐ Secret scanning and push protection are on for every repository.
- ☐ The repository's history was scanned once for old secrets, and any found were revoked.
- ☐ No key is present in the browser or mobile app's code.
- ☐ Every AI and cloud key has a spend limit and an alert.
- ☐ Every secret in the register was rotated on schedule.

References

- OWASP, *Secrets Management Cheat Sheet*.
- NCSC, *Secure development and deployment guidance*.

AI App Policy Kit, version 1.0. © 2026 PugliseWeb LTD, licensed under CC BY 4.0: use it, change it and share it, with credit. The whole kit, and a free Software Health Check of your app against it: https://www.pugliseweb.com/ai-policy-kit