

KEEP SECRETS OUT OF YOUR CODE

Detection helps. A clean workflow starts before the commit.

GitHub's purpose-built model uses surrounding code to detect likely credentials. Availability and billing differ across alerts, push protection and security review; check the linked announcement before enabling a feature.

Before you commit

1. Keep credentials in a secret manager or local environment configuration. Put example names and fake values in shared templates.
2. Review the staged diff and check logs, screenshots, notebooks and generated files for accidental credentials.
3. Use your repository's available secret scanning and push protection. Check the alerts and access settings with the repository owner.
4. Treat AI detections as findings to review. An empty scan is not evidence that every possible secret is absent.

If a credential is exposed

1. Revoke or rotate it with the provider first; removing a line from the latest file does not invalidate the credential.
2. Identify where it was used and check access logs. Follow the provider's recovery procedure.
3. Coordinate any repository history cleanup with collaborators and recheck automation that used the old credential.

Copy this review request

Review this staged diff for hardcoded credentials, accidentally tracked environment files and sensitive values in logs. Report file locations and the type of issue. Do not repeat actual secret values in your response.

ORIGINAL RESOURCE

[GitHub's leaked-secret detection announcement](https://github.blog/changelog/2026-10-07-purpose-built-model-for-leaked-secret-detection/)

<https://github.blog/changelog/2026-10-07-purpose-built-model-for-leaked-secret-detection/>

An independent companion by Huzaifa Shoukat. Check the original project for current setup and availability. Prepared 12 October 2026.