About secret scanning

In this article

GitHub Enterprise Server scans repositories for known types of secrets, to prevent fraudulent use of secrets that were committed accidentally.

Secret scanning is available for organization-owned repositories in GitHub Enterprise Server if your enterprise has a license for GitHub Advanced Security. For more information, see "About secret scanning" and "About GitHub Advanced Security."

Note: Your site administrator must enable secret scanning for your GitHub Enterprise Server instance before you can use this feature. For more information, see "Configuring secret scanning for your appliance."

You may not be able to enable or disable secret scanning, if an enterprise owner has set a policy at the enterprise level. For more information, see "Enforcing policies for code security and analysis for your enterprise."

About secret scanning

If your project communicates with an external service, you might use a token or private key for authentication. Tokens and private keys are examples of secrets that a service provider can issue. If you check a secret into a repository, anyone who has read access to the repository can use the secret to access the external service with your privileges. We recommend that you store secrets in a dedicated, secure location outside of the repository for your project.

Secret scanning will scan your entire Git history on all branches present in your GitHub repository for secrets, even if the repository is archived. Secret scanning does not scan issues.

You can audit the actions taken in response to secret scanning alerts using GitHub tools. For more information, see "Auditing security alerts."

You can also enable secret scanning as a push protection for a repository or an organization. When you enable this feature, secret scanning prevents contributors from pushing code with a detected secret. To proceed, contributors must either remove the secret(s) from the push or, if needed, bypass the protection. Admins can also specify a custom link that is displayed to the contributor when a push is blocked; the link can contain resources specific to the organization to aid contributors. For more information, see "Push protection for repositories and organizations."

Note: When you fork a repository with secret scanning or push protection enabled, these features are not enabled by default on the fork. You can enable secret scanning or push protection on the fork the same way you enable them on a standalone repository.

About secret scanning on GitHub Enterprise Server

Secret scanning is available on all organization-owned repositories as part of GitHub Advanced Security. The feature is not available on user-owned repositories. When you enable secret scanning for a repository, GitHub scans the code for patterns that match secrets used by many service providers. When the scan is completed, GitHub sends an email alert to the enterprise and organization owners, even if no secrets were found.

When a supported secret is leaked, GitHub Enterprise Server generates a secret scanning alert. GitHub will also periodically run a full git history scan of existing content in GitHub Advanced Security repositories where secret scanning is enabled, and send alert notifications following the secret scanning alert notification settings. For more information, see "Secret scanning patterns."

If you're a repository administrator, you can enable secret scanning for any repository, including archived repositories. Organization owners can also enable secret scanning for all repositories or for all new repositories within an organization. For more information, see "Managing security and analysis settings for your repository" and "Managing security and analysis settings for your organization."

You can also define custom secret scanning patterns for a repository, organization, or enterprise. For more information, see "Defining custom patterns for secret scanning."

GitHub stores detected secrets using symmetric encryption, both in transit and at rest. To rotate the encryption keys used for storing the detected secrets, you can contact us by visiting GitHub Enterprise Support.

Accessing secret scanning alerts

When you enable secret scanning for a repository or push commits to a repository with secret scanning enabled, GitHub scans the contents for secrets that match patterns defined by service providers and any custom patterns defined in your enterprise, organization, or repository. GitHub also runs a scan of all historical code content in repositories with secret scanning enabled when a new partner pattern or custom pattern is added or updated.

If secret scanning detects a secret in a commit, GitHub generates an alert.

- GitHub sends an email alert to the repository administrators and organization owners. You'll receive an alert if you are watching the repository, if you have enabled notifications either for security alerts or for all the activity on the repository, and if, in your notification settings, you have selected to receive email notifications for the repositories that you are watching.
- If the person who introduced the secret in the commit isn't ignoring the repository, GitHub will also send them an email alert. The emails contains a link to the related secret scanning alert. The person who introduced the secret can then view the alert in the repository, and resolve the alert.
- GitHub displays an alert in the **Security** tab of the repository.

For more information about viewing and resolving secret scanning alerts, see "Managing alerts from secret scanning."

For more information on how to configure notifications for secret scanning alerts, see "Configuring notifications for secret scanning alerts."

Repository administrators and organization owners can grant users and teams access to secret scanning alerts. For more information, see "Managing security and analysis settings for your repository."

You can use security overview to see an organization-level view of which repositories have enabled secret scanning and the alerts found. For more information, see "About security overview."

You can also use the REST API to monitor results from secret scanning across your repositories or your organization. For more information about API endpoints, see "Secret scanning."

Further reading

- "Securing your repository"
- "Keeping your account and data secure"
- "Best practices for preventing data leaks in your organization"
- "Configuring access to private registries for Dependabot"
- "Using secrets in GitHub Actions"