Managing alerts from secret scanning

In this article

You can view and close alerts for secrets checked in to your repository.

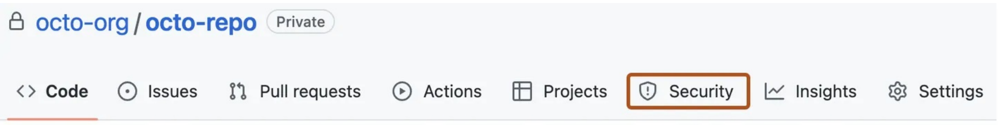
> **Who can use this feature**
> People with admin access to a repository can view and dismiss secret scanning alerts for the repository.

Secret scanning alerts for partners runs automatically on public repositories and public npm packages to notify service providers about leaked secrets on GitHub.com.
Secret scanning alerts for users are available for free on all public repositories. Organizations using GitHub Enterprise Cloud with a license for GitHub Advanced Security can also enable secret scanning alerts for users on their private and internal repositories. For more information, see "About secret scanning" and "About GitHub Advanced Security."
For information about how you can try GitHub Advanced Security for free, see "Setting up a trial of GitHub Advanced Security."

Managing secret scanning alerts

Note: Alerts are created only for repositories with secret scanning alerts for users enabled. Secrets found in public repositories and public npm packages using the free secret scanning alerts for partners service are reported directly to the partner, without creating an alert. For more information, see "Secret scanning patterns."

1. On GitHub.com, navigate to the main page of the repository.
2. Under the repository name, click **Security**. If you cannot see the "Security" tab, select the ··· dropdown menu, and then click **Security**.

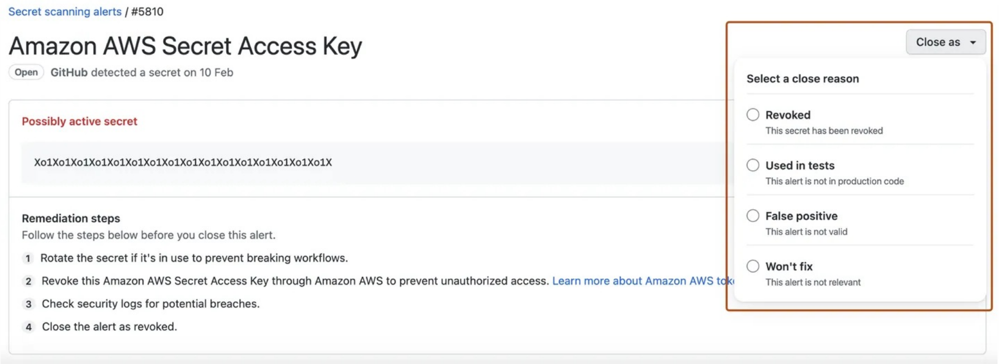


3. In the left sidebar, under "Vulnerability alerts", click **Secret scanning**.
4. Under "Secret scanning" click the alert you want to view.

5. Optionally, to perform a validity check on the token, on the top right-hand side of the alert, click **Verify secret**. For more information, see "Validating partner patterns."

> **Note:** You can only perform on-demand validity checks for patterns detected in the repository if automatic validity checks have been enabled for the repository. For more information, see "Allowing validity checks for partner patterns in a repository."

1. Optionally, if the leaked secret is a GitHub token, you can also review the token metadata. For more information on reviewing token metadata, see "Reviewing GitHub token metadata."
2. To dismiss an alert, select the "Close as" dropdown menu and click a reason for resolving an alert.



Secret scanning alerts / #5810

Amazon AWS Secret Access Key

Open GitHub detected a secret on 10 Feb

Possibly active secret

```
Xo1Xo1Xo1Xo1Xo1Xo1Xo1Xo1Xo1Xo1Xo1Xo1Xo1Xo1Xo1Xo1X
```

Remediation steps

Follow the steps below before you close this alert.

1. Rotate the secret if it's in use to prevent breaking workflows.
2. Revoke this Amazon AWS Secret Access Key through Amazon AWS to prevent unauthorized access. Learn more about Amazon AWS tok
3. Check security logs for potential breaches.
4. Close the alert as revoked.

Close as

Select a close reason

○ Revoked
This secret has been revoked

○ Used in tests
This alert is not in production code

○ False positive
This alert is not valid

○ Won't fix
This alert is not relevant

3. Optionally, in the "Comment" field, add a dismissal comment. The dismissal comment will be added to the alert timeline and can be used as justification during auditing and reporting. You can view the history of all dismissed alerts and dismissal comments in the alert timeline. You can also retrieve or set a comment by using the Secret scanning API. The comment is contained in the `resolution_comment` field. For more information, see "Secret scanning" in the REST API documentation.
4. Click **Close alert**.

Validating partner patterns

> **Note:** Validity checks for partner patterns is currently in beta and subject to change.

Validity checks for partner patterns is available on all types of repositories on GitHub.com. To use this feature, you must have a license for GitHub Advanced Security.

You can allow secret scanning to check the validity of a secret found in your repository by sending it to the relevant partner.

You can enable automatic validity checks for supported partner patterns in the code security settings for your repository, organization, or enterprise. GitHub will periodically send the pattern to the relevant partner to check the secret's validity and display the validation status of the secret in the alert view.

For more information on enabling automatic validation checks for partner patterns in your repository, organization, or enterprise, see "Allowing validity checks for partner patterns in a repository," "Allowing validity checks for partner patterns in an organization," and "Managing Advanced Security features."

If your repository has validity checks enabled, you can also perform an on-demand validity check for a secret by clicking **Verify secret** in the alert view. GitHub will send the pattern to the relevant partner and display the validation status of the secret in the alert view.

You can use the validation status of a leaked secret to help prioritize the secrets needing remediation steps.

Validity	Result
Active secret	GitHub confirmed this secret is active
Active secret	GitHub checked with this secret's provider and found that the secret is active
Possibly active secret	GitHub does not support validation checks for this token type yet
Possibly active secret	GitHub could not verify this secret
Secret appears inactive	You should make sure no unauthorized access has already occurred

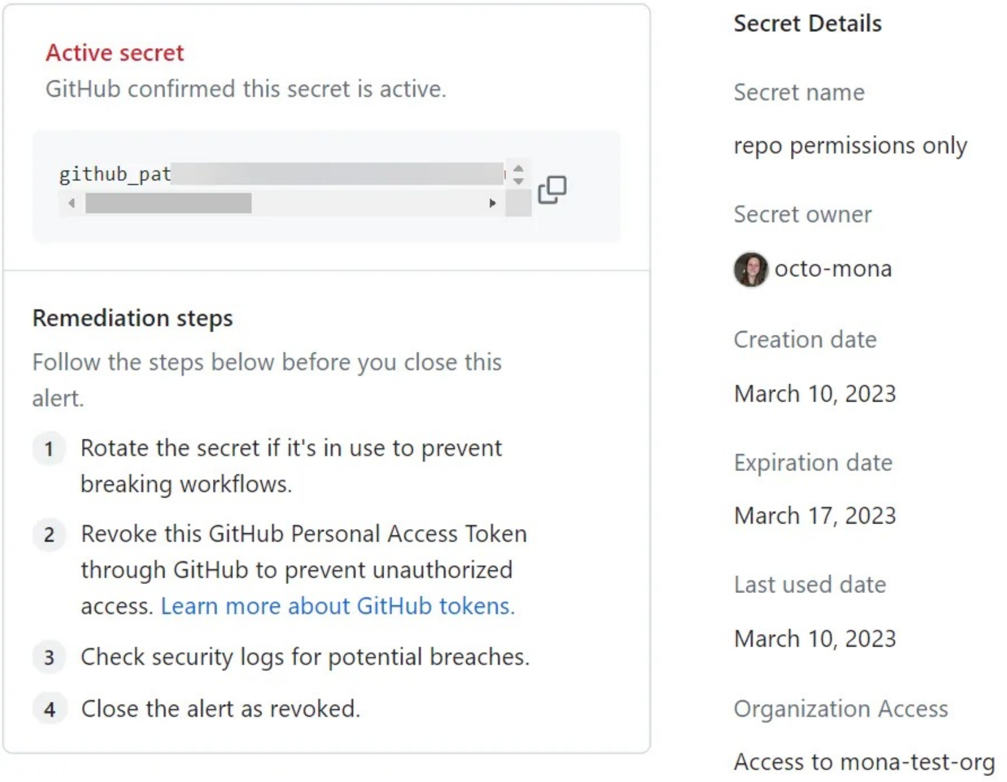
For more information on which partners support validity checks, see "Supported secrets."

Reviewing GitHub token metadata

> **Note:** Metadata for GitHub tokens is currently in public beta and subject to change.

In the view for an active GitHub token alert, you can review certain metadata about the token. This metadata may help you identify the token and decide what remediation steps to take. For more information on viewing individual alerts, see "Managing secret scanning alerts."

Tokens, like personal access token and other credentials, are considered personal information. For more information about using GitHub tokens, see GitHub's Privacy Statement and Acceptable Use Policies.



Active secret

GitHub confirmed this secret is active.

```
github_pat[redacted]
```

Remediation steps

Follow the steps below before you close this alert.

1. Rotate the secret if it's in use to prevent breaking workflows.
2. Revoke this GitHub Personal Access Token through GitHub to prevent unauthorized access. Learn more about GitHub tokens.
3. Check security logs for potential breaches.
4. Close the alert as revoked.

Secret Details

Secret name

repo permissions only

Secret owner

octo-mona

Creation date

March 10, 2023

Expiration date

March 17, 2023

Last used date

March 10, 2023

Organization Access

Access to mona-test-org

Metadata for GitHub tokens is available for active tokens in any repository with secret scanning enabled. If a token has been revoked or its status cannot be validated, metadata will not be available. GitHub auto-revokes GitHub tokens in public repositories, so metadata for GitHub tokens in public repositories is unlikely to be available. The following metadata is available for active GitHub tokens:

Metadata	Description
Secret name	The name given to the GitHub token by its creator
Secret owner	The GitHub handle of the token's owner
Created on	Date the token was created
Expired on	Date the token expired
Last used on	Date the token was last used
Access	Whether the token has organization access

Securing compromised secrets

Once a secret has been committed to a repository, you should consider the secret compromised. GitHub recommends the following actions for compromised secrets:

- For a compromised GitHub personal access token, delete the compromised token, create a new token, and update any services that use the old token. For more information, see "Managing your personal access tokens."
    - If your organization is owned by an enterprise account, identify any actions taken by the compromised token on your enterprise's resources. For more information, see "Identifying audit log events performed by an access token."
- For all other secrets, first verify that the secret committed to GitHub Enterprise Cloud is valid. If so, create a new secret, update any services that use the old secret, and then delete the old secret.

Note: If a secret is detected in a public repository on GitHub.com and the secret also matches a partner pattern, an alert is generated and the potential secret is reported to the service provider. For details of partner patterns, see "Secret scanning patterns."

Configuring notifications for secret scanning alerts

Notifications are different for incremental scans and historical scans.

Incremental scans

When a new secret is detected, GitHub Enterprise Cloud notifies all users with access to security alerts for the repository according to their notification preferences. These users include:

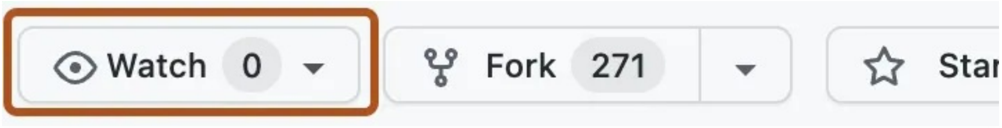
- Repository administrators
- Security managers
- Users with with custom roles with read/write access
- Organization owners and enterprise owners, if they are administrators of repositories where secrets were leaked

Note: Commit authors who've accidentally committed secrets will be notified, regardless of their notification preferences.

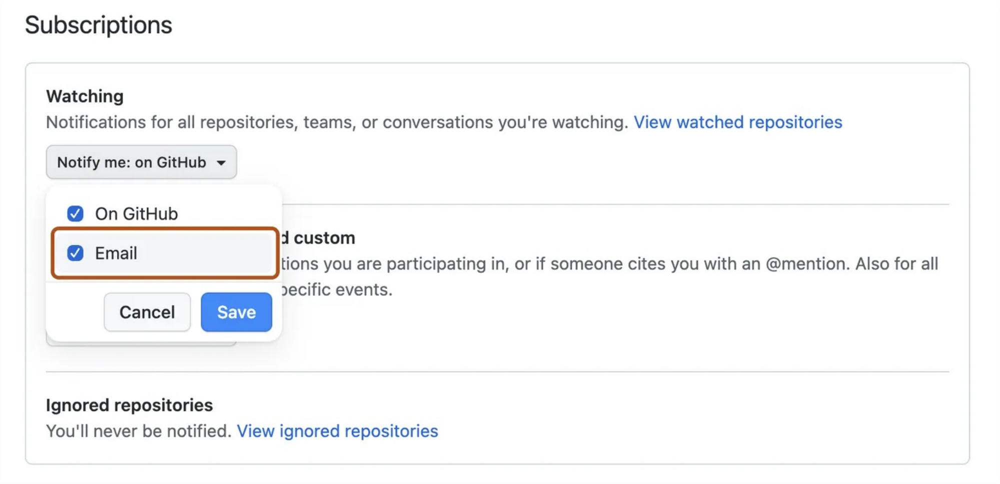
You will receive an email notification if:

- You are watching the repository.
- You have enabled notifications for "All Activity", or for custom "Security alerts" on the repository.
- In your notification settings, under "Subscriptions", then under "Watching", you have selected to receive notifications by email.

1. On GitHub.com, navigate to the main page of the repository.
2. To start watching the repository, select **Watch**.



3. In the dropdown menu, click **All Activity**. Alternatively, to only subscribe to security alerts, click **Custom**, then click **Security alerts**.
4. Navigate to the notification settings for your personal account. These are available at https://github.com/settings/notifications.
5. On your notification settings page, under "Subscriptions", then under "Watching", select the **Notify me** dropdown.
6. Select "Email" as a notification option, then click **Save**.



Subscriptions

Watching

Notifications for all repositories, teams, or conversations you're watching. View watched repositories

Notify me: on GitHub

☑ On GitHub

☑ Email

d custom

tions you are participating in, or if someone cites you with an @mention. Also for all

pecific events.

Ignored repositories

You'll never be notified. View ignored repositories

For more information about setting up notification preferences, see "Managing security and analysis settings for your repository" and "Configuring your watch settings for an individual repository."

Historical scans

For historical scans, GitHub Enterprise Cloud notifies the following users:

- Organization owners, enterprise owners, and security managers—whenever a historical scan is complete, even if no secrets are found.
- Repository administrators, security managers, and users with custom roles with read/write access—whenever a historical scan detects a secret, and according to their notification preferences.

We do *not* notify commit authors.

For more information about setting up notification preferences, see "Managing security and analysis settings for your repository" and "Configuring your watch settings for an individual repository."

Auditing responses to secret scanning alerts

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