Code security / Secret scanning / Troubleshoot secret scanning

Troubleshooting secret scanning

In this article

If you have problems with secret scanning, you can use these tips to help resolve issues.

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."

Detection of pattern pairs

Secret scanning will only detect pattern pairs, such as AWS Access Keys and Secrets, if the ID and the secret are found in the same file, and both are pushed to the repository. Pair matching helps reduce false positives since both elements of a pair (the ID and the secret) must be used together to access the provider's resource.

Pairs pushed to different files, or not pushed to the same repository, will not result in alerts. For more information about the supported pattern pairs, see the table in "Secret scanning patterns."

About legacy GitHub tokens

For GitHub tokens, we check the validity of the secret to determine whether the secret is active or inactive. This means that for legacy tokens, secret scanning won't detect a GitHub Enterprise Server personal access token on GitHub Enterprise Cloud. Similarly, a GitHub Enterprise Cloud personal access token won't be found on GitHub Enterprise Server.

Push protection limitations

If push protection did not detect a secret that you think should have been detected, then you should first check that push protection supports the secret type in the list of supported secrets. For further information, see "Secret scanning patterns."

If your secret is in the supported list, there are various reasons why push protection may not detect it.

- Push protection only blocks leaked secrets on a subset of the most identifiable user-alerted patterns. Contributors can trust security defenses when such secrets are blocked as these are the patterns that have the lowest number of false positives.
- The version of your secret may be old. Older versions of certain tokens may not be supported by push protection as these tokens may generate a higher number of false positives than their most recent version. Push protection may also not apply to legacy tokens. For tokens such as Azure Storage Keys, GitHub only supports *recently created* tokens, not tokens that match the legacy patterns.
- The push may be too large, for example, if you're trying to push thousands of large files. A push protection scan may time out and not block a user if the push is too large. GitHub will still scan and create alerts, if needed, after the push.
- If the push results in the detection of over five new secrets, we will only show you the first five (we will always show you a maximum of five secrets at one time).
- If a push contains over 1,000 existing secrets (that is, secrets for which alerts have already been created), push protection will not block the push.