This article is also available in French.

Managing security and analysis settings for your organization

You can control features that secure and analyze the code in your organization's projects on GitHub.

Who can use this feature?

Organization owners can manage security and analysis settings for repositories in the organization.

About management of security and analysis settings

GitHub can help you to secure the repositories in your organization. You can manage the security and analysis features for all existing or new repositories that members create in your organization. Organizations that use GitHub Enterprise Cloud with a license for GitHub Advanced Security can also manage access to these features. For more information, see the GitHub Enterprise Cloud documentation.

> **Note**: You can't disable some security and analysis features that are enabled by default for public repositories.

You can quickly enable security features at scale with the GitHub-recommended security configuration, a collection of security enablement settings you can apply to repositories in an organization. You can then further customize GitHub Advanced Security features at the organization level with global settings. See "About enabling security features at scale."

If you enable security and analysis features, GitHub performs read-only analysis on your repository.

Allowing Dependabot to access private dependencies

Dependabot can check for outdated dependency references in a project and automatically generate a pull request to update them. To do this, Dependabot must have access to all of the targeted dependency files. Typically, version updates will fail if one or more dependencies are inaccessible. For more information, see "About Dependabot version updates."

By default, Dependabot can't update dependencies that are located in private repositories, or private package registries. However, if a dependency is in a private GitHub repository within the same organization as the project that uses that dependency, you can allow Dependabot to update the version successfully by giving it access to the host repository.

If your code depends on packages in a private registry, you can allow Dependabot to update the versions of these dependencies by configuring this at the repository level. You do this by adding authentication details to the `dependabot.yml` file for the repository. For more information, see "Configuration options for the dependabot.yml file."

For more information on how to grant Dependabot access to private dependencies, see "Configuring global security settings for your organization."

Removing access to GitHub Advanced Security from individual repositories in an organization

You can use security configurations to remove access to GitHub Advanced Security from individual repositories in an organization. For more information, see "Managing your GitHub Advanced Security license usage."

Further reading

- "Quickstart for securing your repository"
- "About the dependency graph"
- "About supply chain security"

Help and support

Did you find what you needed?

Yes No

Privacy policy

Help us make these docs great!

All GitHub docs are open source. See something that's wrong or unclear? Submit a pull request.

Make a contribution

Learn how to contribute

Still need help?

Ask the GitHub community

Contact support

Legal

© 2024 GitHub, Inc. Terms Privacy Status Pricing Expert services Blog