GitHub

Detection Strategy Guide

June 2025

Product Overview

GitHub, Inc. is an American multinational corporation that provides hosting for software development and version control using Git. It offers the distributed version control and source code management functionality of Git, plus its own features. Projects on GitHub.com can be accessed and managed using the standard Git command-line interface, web browsers, and/or multiple desktop clients.

GitHub.com also allows users to browse public repositories on the site. The site provides social networking-like functions such as feeds, followers, wikis, and a social network graph to show how developers work on their versions ("forks") of a repository and what fork (and branch within that fork) is the newest.

Anyone can browse and download public repositories, but only registered users can contribute content to repositories. With a registered user account, users can have discussions, manage repositories, submit contributions to others' repositories, and review changes to code.

Detection Strategy for SaaS Integrations

Detection

Our SaaS detection strategy is designed to identify and respond to suspicious user activity across cloud-based applications. By integrating directly with SaaS platforms, we continuously monitor user behavior and focus on activities such as unusual login patterns, excessive data downloads, or unauthorized access to sensitive files.

In addition to user activity, we also detect other key SaaS-related events such as changes to administrative settings, the creation or modification of privileged accounts, and unexpected data sharing with external parties. These events are processed through our detection engine, which leverages behavioral analytics and threat intelligence to flag potential risks. When anomalies are detected, our automated response bot, Ruxie, takes action by enriching evidence fields with first- and third-party threat intelligence. Additional Ruxie actions query a wide span of technologies directly to arm analysts with key pieces of investigative information and related events.

Response

Similar to identity technologies, SaaS apps can hold valuable information such as user roles, location, and other metadata that helps analysts triage Expel Alerts of all types that contain source user information.

To learn more about our overall approach to detection strategy, see About Detection Strategy in the Help Center.

What We Support for GitHub

To see a comprehensive list of the most up-to-date Expel detection rules, vendor detection rules, opt-in detections, and available DUETs (**d**id yo**u** **e**xpect **t**his) that we support for GitHub, you can visit the Detections page in Workbench or ask your Sales or Support rep for the most recent download.

Supported event log sources	■ GitHub GraphQL API Audit Log ■ GitHub REST API Audit Log
GitHub detection rules support	No.
Detection rules written by Expel	Yes.
Auto remediations	Yes. Expel supports automatic execution of some remediation actions for this integration when you follow our setup guide to update the permissions in your vendor device, and then enable the auto remediation in Workbench. The available auto remediations for this integration include: ■ Disable Accounts To enable auto remediations for your environment, see Enable an Auto Remediation in Workbench in the Help Center.
Investigative support through Workbench	Yes. We are able to take the following investigative actions to gather data for triage and investigation of events. ■ Query Logs ■ Query Raw Logs ■ SaaS App Triage Workflow
Hunting support	No. Hunting is not currently available for this integration.

Additional Details and Common Questions

Console Access

A vendor alert does not typically include all of the contextual timeline activity surrounding the event of interest. Because this integration does not allow us to get all necessary data via the API, we will ask you for a certain level of console access during onboarding.

The level of access that we require is meant to support essential triage and research activities, and to help us determine the vector and extent of attacker activity for an identified threat. At minimum, we will ask for visibility into alert data, timeline events recorded, and live response/real time response shell (if applicable).

DUET

A DUET (**d**id yo**u** **e**xpect **t**his) rule flags certain events as needing an immediate verification or notification, and bypasses the normal internal event triage process. The events subject to DUET rules contain behaviors that are not typically indicative of true security incidents, as they are related to policy violations or *potential* risk.

There are a number of workflows that a DUET may follow. When enabled, the activity will be flagged for investigation and will be routed to you (rather than to us) to take a specified first action. To see the specific DUET rules currently supported for this integration, visit the Detections page in Workbench.

Why is Expel requiring admin:org level permissions?

The requirement for admin:org comes from GitHub, it's not a requirement that Expel imposes. Expel tested the integration with only admin:read and we were unable to query the Audit Log through GraphQL API. This requirement remains the same for the REST API Audit Log as well. The requirement by GitHub isn't optional. The integration doesn't work without it. If the permissions are not authorized, Expel can't onboard the integration.

Reference:

- GraphQL API: https://docs.github.com/en/graphql/overview/about-the-graphql-api
- REST API: https://docs.GitHub.com/en/rest/reference/enterprise-admin and https://docs.GitHub.com/en/rest/reference/orgs#get-the-audit-log-for-an-organization

Why is Workbench not offering webhook support?

Webhook provides a lot of event notifications but isn't the best method to acquire the Audit Log, in particular, certain security events like Git audit events.