mitiga

GitHub Logs

A Now You See Me cheat sheet

GitHub

“Now You See Me” is a Mitiga Labs series that demystifies SaaS security log by log with the why, what, and how of SaaS threat hunting.

Why GitHub Belongs in Your Security Scope

GitHub is a DevOps platform that controls automation, tokens, permissions, and workflows.

Attackers can use it to gain persistence, pivot into CI/CD, and deploy malicious changes. That makes its logs relevant to every security team.

What’s Actually in GitHub Logs

GitHub provides multiple log types useful for investigations:

- **Audit Logs:** Changes to access, repos, tokens, and org-level settings
- **User Security Logs:** Logins, MFA changes, password resets
- **Actions Workflow Logs:** Triggered workflows, jobs, runners, and artifacts
- **App and OAuth Events:** Token use, scope changes, GitHub App installs and removals

What You May Not Know About These Logs

- **Missing attribution.** Events from tokens and GitHub Apps may lack IPs, which weakens geo- and velocity-based detections.
- **Token ownership isn’t obvious.** A long-lived PAT used today might not tie back to a known user.
- **Repo deletions cut your timeline.** You might see the delete event, but not who accessed the repo beforehand if you aren’t streaming the logs.
- **Forks are easy to miss.** Exfiltration via repo forks can blend into the noise.
- **GitHub Actions logs don’t show full execution.** You get metadata, not the commands run inside runners.

GitHub Threat Hunting Tips & Investigation Tricks

Skilled threat hunters and detection engineers should make use of these kinds of tactics with GitHub logs.

Watch for new GitHub App installs.

High-impact change with broad permissions implications.

Track token behavior.

Look for unusual activity, geo shifts, re-use of old credentials.

Correlate access with repo sensitivity.

Pay attention when access patterns break from the norm.

Monitor workflow file changes.

New or edited workflows may precede privilege escalation/credential harvesting.

Look for MFA and auth changes.

Disabling MFA, adding SSH keys, or odd SSO logins should trigger review.

Treat GitHub as a security-critical system. Export and retain your logs, normalize identity activity, and monitor for permission drift and token misuse.

Even small gaps—like missed App installs or untracked token use—can become critical blind spots during an incident.

Want to see how Mitiga helps you uncover what others miss?

Learn more about our **Zero-Impact Breach Prevention** platform or request a live demo.

www.mitiga.io/demo