4. Security and compliance

└ 4.1 Authentication and authorization

└ 4.1.2 GitHub auth: Apps, tokens

---

1. What are the primary authentication methods available for GitHub automation?
2. What is a GitHub App and when should it be used?
3. What is a personal access token (PAT) and what is it used for?
4. What is the GITHUB_TOKEN and how is it used in GitHub Actions?
5. How do you control the permissions of a GitHub App versus a PAT?
6. When should you use GITHUB_TOKEN instead of a PAT in GitHub Actions?
7. How do you authenticate to GitHub in CI/CD pipelines securely?
8. What are the risks associated with PATs and how can you mitigate them?
9. How do you revoke access for GitHub Apps, PATs, and GITHUB_TOKEN?
10. What auditing capabilities exist for GitHub authentication methods?

---

1. **What are the primary authentication methods available for GitHub automation?**

The main methods are

- GitHub Apps,
- personal access tokens (PATs),
- and the built-in GITHUB_TOKEN used by GitHub Actions.

---

2. **What is a GitHub App and when should it be used?**

A GitHub App is an integration with granular permissions, installed directly on repositories or organizations. Use it for automation, CI/CD, and integrations needing controlled, auditable access.

---

3. **What is a personal access token (PAT) and what is it used for?**

A PAT is a user-generated token that grants API access on behalf of a user. It is used for scripting, CLI tools, and third-party integrations where user-based access is needed.

---

4. **What is the GITHUB_TOKEN and how is it used in GitHub Actions?**

GITHUB_TOKEN is an automatically generated token available to workflows, scoped to the repository and job. It allows secure automation within Actions without manual secret handling.

---

5. **How do you control the permissions of a GitHub App versus a PAT?**

- GitHub Apps have fine-grained, configurable permissions per repository and event.
- PATs inherit the full or partial permissions of the user who created them.

---

6. **When should you use GITHUB_TOKEN instead of a PAT in GitHub Actions?**

Use GITHUB_TOKEN for most automation in Actions as it is automatically created, limited in scope, and auto-rotated—minimizing risk of credential leakage.

---

7. **How do you authenticate to GitHub in CI/CD pipelines securely?**

Use GITHUB_TOKEN for native Actions, GitHub App tokens for granular access, and store any PATs in encrypted secrets—never hardcode credentials.

8. **What are the risks associated with PATs and how can you mitigate them?**
PATs are long-lived and may have broad access if leaked.

- Limit their scope,
- rotate regularly,
- use only when required,
- and store securely in GitHub Secrets.

---

9. **How do you revoke access for GitHub Apps, PATs, and GITHUB_TOKEN?**

- Uninstall the GitHub App,
- delete or regenerate the PAT from user settings,
- and GITHUB_TOKEN is invalidated automatically at job end.

---

11. **What auditing capabilities exist for GitHub authentication methods?**
Audit

- logs track PAT usage,
- GitHub App installation and actions,
- and GITHUB_TOKEN workflow invocations.

Review logs for unauthorized or unusual access.