GitLab Setup

CBOM Secure · Source Code Sensor (Discover_SourceCode)

Overview

This guide configures GitLab so the CBOM Secure Source Code Discovery Sensor can discover cryptographic material in repositories — hard-coded private keys and certificates, API tokens and secrets, SSH private keys, and cryptographic configuration files. Covers GitLab SaaS (gitlab.com) and self-managed instances. All access is read-only.

Prerequisites

- A GitLab account (SaaS or self-managed) with at least Reporter role on the target group/project.
- Self-managed: GitLab 13.0+ (Personal Access Tokens) or 14.1+ (Project Access Tokens).
- The CBOM Collector has HTTP/HTTPS access to the GitLab instance.

Step-by-Step Guide

Step 1: Create a Personal Access Token

Under your avatar → Edit profile → Access Tokens, create cbom-source-discovery with the minimum scopes read_api and read_repository (no write scopes). Copy the token (shown once); it appears as glpat-xxxxxxxxxxxxxxxxxxxx.

Step 2: (Optional) Create a Project Access Token

To restrict to a single project, use Project → Settings → Access Tokens with role Reporter and scopes read_api and read_repository.

Step 3: (Self-Managed) Confirm the API Base URL

```
curl -s --header "PRIVATE-TOKEN: <your_token>" \
  "https://<your-gitlab-hostname>/api/v4/version"
```

For an internal CA, obtain the CA PEM and reference it via ssl_ca_cert in the sensor config.

Step 4: Configure the CBOM Secure Sensor

GitLab SaaS with a Personal Access Token:

```
sensor: Discover_SourceCode
platform: gitlab
gitlab_url: "https://gitlab.com"
private_token: "glpat-xxxxxxxxxxxxxxxxxxxx"
scan_groups:
  - "my-org-group"
scan_name: GitLabSourceCodeDiscovery
```

Self-managed with an internal CA, or single-project scoping:

```
# self-managed
gitlab_url: "https://gitlab.internal.example.com"
ssl_ca_cert: "/etc/cbom/certs/gitlab-ca.pem"
scan_groups: ["engineering", "security"]

# single project (project access token)
scan_projects: ["my-org-group/my-repository"]
```

Step 5: Validate

```
curl -s --header "PRIVATE-TOKEN: glpat-xxxx" \
  "https://gitlab.com/api/v4/projects?membership=true&per_page=5"
curl -s --header "PRIVATE-TOKEN: glpat-xxxx" \
  "https://gitlab.com/api/v4/projects/<project_id>/repository/tree"
```

Run the discovery task and confirm it completes without 401/403 errors; confirm the token's Last Used timestamp updates in GitLab.

Common Errors

401 Unauthorized

Cause: The token value is incorrect, revoked, or expired.

Resolution: Verify the token in the YAML matches exactly; check expiry; regenerate and update the config.

403 Forbidden

Cause: Missing read_api / read_repository scope, or a Project Access Token below Reporter.

Resolution: Recreate the token with both scopes; ensure at least Reporter access to the target namespace.

SSL certificate verify failed

Cause: Self-managed GitLab uses an internal CA not trusted by the Collector.

Resolution: Set ssl_ca_cert to the internal CA PEM and ensure the Collector can read it.

Security Recommendations

- Use a dedicated GitLab service account (e.g. cbom-scanner), not a personal account.
- Grant only read_api and read_repository — never api (full) or write scopes.
- Set token expiry and rotate on schedule; update the sensor config on rotation.
- On self-managed GitLab EE, use token IP allowlists limited to the Collector IP.
- Review GitLab audit events to confirm activity matches scan schedules.

Conclusion

Operating with only read_api and read_repository scopes, the Discover_SourceCode sensor scans GitLab SaaS and self-managed repositories read-only — with proper TLS trust for internal CAs — feeding discovered cryptographic material into CBOM Secure.