

Dependabot Alerts & Workflows — Hands-On Cheatsheet

Heads-up: This document uses `minted` for syntax highlighting. If you want actual GraphQL highlighting, install the Pygments lexer plugin `pygments-graphql` and switch the block back to `graphql`. Compile with: `pdflatex -shell-escape dependabot-cheatsheet.tex` (or `xelatex` with `-shell-escape`).

1 Core Concepts

- **Dependency types:** Direct dependencies are packages your code imports/uses. *Transitive* dependencies are packages required by your direct dependencies.
- **Where alerts appear:** Security tab (repository overview), Pull Requests (as comments/checks), and via API/CLI.
- **Security updates:** Dependabot can automatically open PRs to bump vulnerable versions.

2 Enable Security Updates

Add or adjust `dependabot.yml` at repo/org level to scan ecosystems and open PRs automatically.

Minimal `dependabot.yml`

```
1 version: 2
2
3 updates:
4   - package-ecosystem: "pip"          # npm, maven, gradle, gomod, etc.
5     directory: "/"                   # location of manifest
6     schedule:
7       interval: "daily"            # daily, weekly, monthly
8     open-pull-requests-limit: 5
9     rebase-strategy: auto
10    reviewers: ["team/security"]
11    labels: ["dependencies", "security"]
```

3 Triage & Remediation Workflow

1. Review new alerts in **Security → Alerts** or in the PR that raised them.
2. Validate severity, affected manifests, and suggested upgrade.
3. Run tests locally and/or in CI (unit, integration, regression).
4. Merge the Dependabot PR (or create one via *Update now*) and monitor post-merge checks.

4 Query Alerts Programmatically

4.1 GraphQL API (curl)

```
----- Query via GraphQL (replace TOKEN/owner/repo) -----
1 curl -s -H "Authorization: bearer TOKEN" \
2   -X POST https://api.github.com/graphql \
3   -d '{"query":'
4 query {
5   repository(owner:\"OWNER\", name:\"REPO\") {
6     vulnerabilityAlerts(first: 20) {
7       nodes {
8         securityVulnerability { severity package { name } }
9         vulnerableManifestFilename
10        vulnerableRequirements
11        createdAt
12        state
13      }
14    }
15  }
16 }"}' | jq .
```

4.2 GraphQL Shape

```
----- GraphQL selection set sketch -----
1 query {
2   repository(owner: "OWNER", name: "REPO") {
3     vulnerabilityAlerts(first: 20) {
4       nodes {
5         securityVulnerability { severity package { name } }
6         vulnerableManifestFilename
7         vulnerableRequirements
8         createdAt state
9       }
10     }
11   }
12 }
```

4.3 REST via GitHub CLI

```
----- Fetch Dependabot alerts with gh api -----
1 # Authenticate once
2 gh auth login --web --hostname github.com
3
4 # List alerts (REST): /repos/{owner}/{repo}/dependabot/alerts
5 gh api repos/OWNER/REPO/dependabot/alerts --paginate > alerts.json
6
7 # Convert to CSV (example jq shape)
8 jq -r '.[] | [
9   .number,
10  .state,
11  .dependency.package.name,
12  .security_advisory.severity,
13  .security_vulnerability.severity,
14  .created_at
15 ] | @csv' alerts.json > alerts.csv
```

5 Working From the Security Tab or PR

- **Security tab:** Review alert details and use *Update now* to spawn a security update PR.
- **Pull Request:** Dependabot comments highlight the vulnerable dependency and the safe version.
- **Change control:** PR discussion and reviews record decisions and provide traceability.

6 Advisory Intelligence

- Browse the **GitHub Advisory Database** (filter by ecosystem, severity, CWE, newest).
- Use it to proactively check dependencies and plan upgrade windows.

7 Dependabot CLI and Core

- **Dependabot Core:** Core logic is open source; backend services are proprietary to GitHub.
- **CLI:** Useful for local testing/debugging of update jobs.

8 Automation Tips

- Ensure unit/integration tests run on Dependabot PRs.
- Consider grouping updates to reduce PR noise when appropriate.
- Label, auto-assign reviewers, and protect main branches with required checks.

9 Quick Commands

```
1 # Authenticate GitHub CLI (web flow)
2 gh auth login --web --hostname github.com
3
4 # List Dependabot alerts (REST)
5 gh api repos/OWNER/REPO/dependabot/alerts
6
7 # Example: narrow by severity (jq)
8 gh api repos/OWNER/REPO/dependabot/alerts \
  | jq '.[] | select(.security_advisory.severity=="critical")'
```