4. Security and compliance
└ 4.4 Security and compliance scanning
└ 4.4.4 CodeQL, Container Scanning

---

1. What is CodeQL and how does it work in GitHub Advanced Security?
2. How do you configure CodeQL analysis in a GitHub repository?
3. What languages are supported by CodeQL?
4. How does CodeQL identify vulnerabilities?
5. How do you run CodeQL in a containerized environment?
6. What tools are commonly used for container image scanning?
7. How can GitHub Actions be used to scan container images?
8. How do you automate container scanning in CI/CD pipelines?
9. How are container scan results surfaced and consumed?
10. What best practices ensure secure use of CodeQL and container scanning in DevOps?

---

1. What is CodeQL and how does it work in GitHub Advanced Security?
CodeQL performs semantic analysis on source code using a query language to detect vulnerabilities. It analyzes code on PRs, pushes, and schedules.

---

2. How do you configure CodeQL analysis in a GitHub repository?
Add the `github/codeql-action/init` and analyze actions in a `.github/workflows/codeql.yml` file. Select target languages and triggers (e.g., push, PR).

---

3. What languages are supported by CodeQL?
CodeQL supports C/C++, C#, Go, Java, JavaScript/TypeScript, Python, Ruby, and Swift.

---

4. How does CodeQL identify vulnerabilities?
It uses a query engine to detect security patterns (e.g., SQL injection, XSS).
Queries are run against the code database built during analysis.

---

5. How do you run CodeQL in a containerized environment?
Use GitHub-hosted or self-hosted runners with container support.
CodeQL can analyze code inside Docker containers by configuring the `container: option` in workflows.

---

6. What tools are commonly used for container image scanning?
Microsoft Defender for Cloud, *Trivy*, *Aqua*, and *Anchore* are popular tools.
They detect OS/package-level vulnerabilities and policy violations.

---

7. How can GitHub Actions be used to scan container images?
Use actions like `aquasecurity/trivy-action` or `anchore/scan-action` in a workflow to scan images after build and fail the job on high-severity findings.

---

8. How do you automate container scanning in CI/CD pipelines?
Integrate scanning steps post-image-build in *GitHub Actions* or *Azure Pipelines*.
Scan images before pushing to registry and gate releases on scan results.

9. How are container scan results surfaced and consumed?
Results appear in the *GitHub* Security tab (if supported), job logs, or are exported to external tools. *Defender* shows results in *Azure Security Center*.

---

10. What best practices ensure secure use of CodeQL and container scanning in DevOps?

- Scan on every PR,
- enforce status checks,
- update scan tools regularly,
- exclude false positives via configuration,
- and act on critical alerts before deploy.