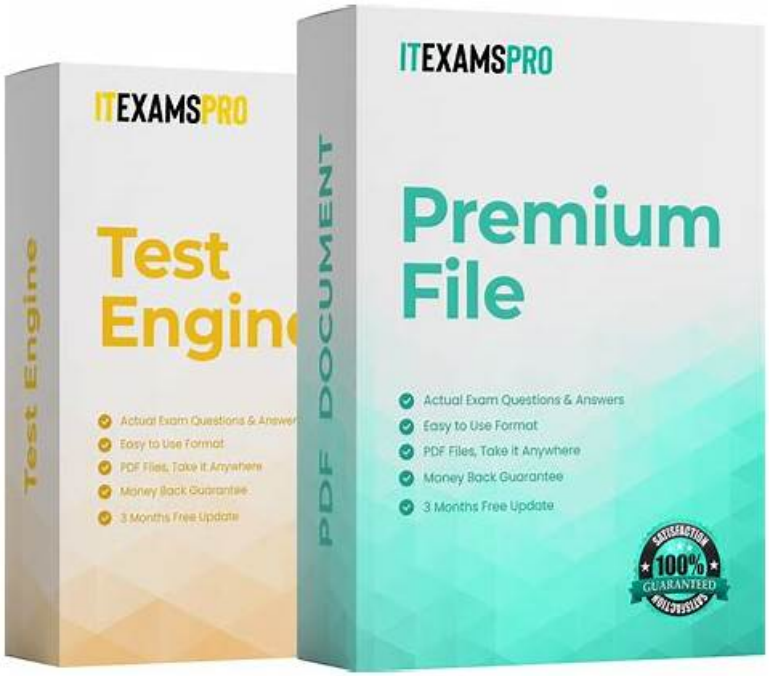
GH-500 Test Practice - Exam GH-500 PDF



Our Microsoft GH-500 PDF dumps format has actual GH-500 questions which are printable and portable. Hence, you can go through these Microsoft GH-500 questions via your smart devices like smartphones, laptops, and tablets. The GitHub Advanced Security (GH-500) dumps PDF file can be used from any location and at any time. Furthermore, you can take print of Microsoft Questions PDF to do an off-screen study.

Microsoft GH-500 exam dumps are important because they show you where you stand. After learning everything related to the GitHub Advanced Security (GH-500) certification, it is the right time to take a self-test and check whether you can clear the GH-500 certification exam or not. People who score well on the GH-500 Practice Questions are ready to give the final GitHub Advanced Security (GH-500) exam. On the other hand, those who do not score well can again try reading all the GH-500 dumps questions and then give the GH-500 exam.

>> GH-500 Test Practice <<

Three formats of Exams-boost Microsoft GH-500 Exam Preparation Material

Our Microsoft GH-500 practice exam simulator mirrors the GH-500 exam experience, so you know what to anticipate on GitHub Advanced Security (GH-500) certification exam day. Our GitHub Advanced Security practice test Exams-boost features various question styles and levels, so you can customize your Microsoft GH-500 Exam Questions preparation to meet your needs.

Microsoft GH-500 Exam Syllabus Topics:

Topic	Details

Topic 1	• Describe GitHub Advanced Security best practices, results, and how to take corrective measures: This section evaluates skills of Security Managers and Development Team Leads in effectively handling GHAS results and applying best practices. It includes using Common Vulnerabilities and Exposures (CVE) and Common Weakness Enumeration (CWE) identifiers to describe alerts and suggest remediation, decision-making processes for closing or dismissing alerts including documentation and data-based decisions, understanding default CodeQL query suites, how CodeQL analyzes compiled versus interpreted languages, the roles and responsibilities of development and security teams in workflows, adjusting severity thresholds for code scanning pull request status checks, prioritizing secret scanning remediation with filters, enforcing CodeQL and Dependency Review workflows via repository rulesets, and configuring code scanning, secret scanning, and dependency analysis to detect and remediate vulnerabilities earlier in the development lifecycle, such as during pull requests or by enabling push protection.
Topic 2	• Configure and use Dependabot and Dependency Review: Focused on Software Engineers and Vulnerability Management Specialists, this section describes tools for managing vulnerabilities in dependencies. Candidates learn about the dependency graph and how it is generated, the concept and format of the Software Bill of Materials (SBOM), definitions of dependency vulnerabilities, Dependabot alerts and security updates, and Dependency Review functionality. It covers how alerts are generated based on the dependency graph and GitHub Advisory Database, differences between Dependabot and Dependency Review, enabling and configuring these tools in private repositories and organizations, default alert settings, required permissions, creating Dependabot configuration files and rules to auto-dismiss alerts, setting up Dependency Review workflows including license checks and severity thresholds, configuring notifications, identifying vulnerabilities from alerts and pull requests, enabling security updates, and taking remediation actions including testing and merging pull requests.
Topic 3	• Configure and use secret scanning: This domain targets DevOps Engineers and Security Analysts with the skills to configure and manage secret scanning. It includes understanding what secret scanning is and its push protection capability to prevent secret leaks. Candidates differentiate secret scanning availability in public versus private repositories, enable scanning in private repos, and learn how to respond appropriately to alerts. The domain covers alert generation criteria for secrets, user role-based alert visibility and notification, customizing default scanning behavior, assigning alert recipients beyond admins, excluding files from scans, and enabling custom secret scanning within repositories.
Topic 4	• Describe the GHAS security features and functionality: This section of the exam measures skills of Security Engineers and Software Developers and covers understanding the role of GitHub Advanced Security (GHAS) features within the overall security ecosystem. Candidates learn to differentiate security features available automatically for open source projects versus those unlocked when GHAS is paired with GitHub Enterprise Cloud (GHEC) or GitHub Enterprise Server (GHES). The domain includes knowledge of Security Overview dashboards, the distinctions between secret scanning and code scanning, and how secret scanning, code scanning, and Dependabot work together to secure the software development lifecycle. It also covers scenarios contrasting isolated security reviews with integrated security throughout the development lifecycle, how vulnerable dependencies are detected using manifests and vulnerability databases, appropriate responses to alerts, the risks of ignoring alerts, developer responsibilities for alerts, access management for viewing alerts, and the placement of Dependabot alerts in the development process.
Topic 5	• Configure and use Code Scanning with CodeQL: This domain measures skills of Application Security Analysts and DevSecOps Engineers in code scanning using both CodeQL and third-party tools. It covers enabling code scanning, the role of code scanning in the development lifecycle, differences between enabling CodeQL versus third-party analysis, implementing CodeQL in GitHub Actions workflows versus other CI tools, uploading SARIF results, configuring workflow frequency and triggering events, editing workflow templates for active repositories, viewing CodeQL scan results, troubleshooting workflow failures and customizing configurations, analyzing data flows through code, interpreting code scanning alerts with linked documentation, deciding when to dismiss alerts, understanding CodeQL limitations related to compilation and language support, and defining SARIF categories.

Microsoft GitHub Advanced Security Sample Questions (Q13-Q18):

NEW QUESTION # 13

Which of the following secret scanning features can verify whether a secret is still active?

- A. Branch protection
- B. Validity checks
- C. Push protection
- D. Custom patterns

Answer: B

Explanation:
Validity checks, also called secret validation, allow GitHub to check if a detected secret is still active. If verified as live, the alert is marked as "valid", allowing security teams to prioritize the most critical leaks.
Push protection blocks secrets but does not check their validity. Custom patterns are user-defined and do not include live checks.

NEW QUESTION # 14
What should you do after receiving an alert about a dependency added in a pull request?

- A. Fork the branch and deploy the new fork
- B. Disable Dependabot alerts for all repositories owned by your organization
- C. Update the vulnerable dependencies before the branch is merged
- D. Deploy the code to your default branch

Answer: C

Explanation:
If an alert is raised on a pull request dependency, best practice is to update the dependency to a secure version before merging the PR. This prevents the vulnerable version from entering the main codebase.
Merging or deploying the PR without fixing the issue exposes your production environment to known risks.

NEW QUESTION # 15
What role is required to change a repository's code scanning severity threshold that fails a pull request status check?

- A. Admin
- B. Triage
- C. Write
- D. Maintain

Answer: A

Explanation:
To change the threshold that defines whether a pull request fails due to code scanning alerts (such as blocking merges based on severity), the user must have Admin access on the repository. This is because modifying these settings falls under repository configuration privileges.
Users with Write, Maintain, or Triage roles do not have the required access to modify rulesets or status check policies.

NEW QUESTION # 16
You are managing code scanning alerts for your repository. You receive an alert highlighting a problem with data flow. What do you click for additional context on the alert?

- A. Code scanning alerts
- B. Security
- C. Show paths

Answer: C

Explanation:
When dealing with a data flow issue in a code scanning alert, clicking on "Show paths" provides a detailed view of the data's journey through the code. This includes the source of the data, the path it takes, and where it ends up (the sink). This information is crucial

for understanding how untrusted data might reach sensitive parts of your application and helps in identifying where to implement proper validation or sanitization.

NEW QUESTION # 17
Which Dependabot configuration fields are required? (Each answer presents part of the solution. Choose three.)

- A. package-ecosystem
- B. directory
- C. milestone
- D. schedule.interval
- E. allow

Answer: A,B,D

Explanation:
Comprehensive and Detailed Explanation:
When configuring Dependabot via the dependabot.yml file, the following fields are mandatory for each update configuration:
directory: Specifies the location of the package manifest within the repository. This tells Dependabot where to look for dependency files.
package-ecosystem: Indicates the type of package manager (e.g., npm, pip, maven) used in the specified directory.
schedule.interval: Defines how frequently Dependabot checks for updates (e.g., daily, weekly). This ensures regular scanning for outdated or vulnerable dependencies.
The milestone field is optional and used for associating pull requests with milestones. The allow field is also optional and used to specify which dependencies to update.
GitLab

NEW QUESTION # 18
.....

500 " and download it for free on ➽ www.pdfdumps.com □ website □GH-500 Reliable Study Guide

- www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, in.ecomsolutionservices.com, lms.ait.edu.za, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, Disposable vapes