Free PDF 2026 Microsoft GH-500 Newest Latest Real Exam

DROUMT
$50 Unlimited Access Pack Included! >
+ Android, Offline Off Online Testing Engine- No restrictions-
+ Oven Axam Printalle PDFS,
+ Free updates
+ 100% passing Guarantee / Money-Back
+ Bok Assurance
>50% OFF!
Microsofa
CISCO
ORACLE
SAP
VIBM
HP
juniper

P.S. Free & New GH-500 dumps are available on Google Drive shared by TestInsides: https://drive.google.com/open?id=1EkxllAXSkmldxqE-btlEdYFMACVRw1oi

The cost of registering a GH-500 Certification is quite expensive, ranging between $100 and $1000. After paying such an amount, the candidate is sure to be on a tight budget. TestInsides provides Microsoft GH-500 preparation material at very low prices compared to other platforms. We also assure you that the amount will not be wasted and you will not have to pay for the certification a second time. For added reassurance, we also provide up to 1 year of free updates. Free demo version of the actual product is also available so that you can verify its validity before purchasing.

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 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.

Topic 3	- 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 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 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.

>> Latest Real GH-500 Exam <<

Microsoft GH-500 Detailed Study Dumps & GH-500 New Soft Simulations

You can study GH-500 exam engine anytime and anyplace for the convenience our three versions of our GH-500 study questions bring. What is more, it is our mission to help you pass the exam. Our study materials will provide you with 100% assurance of passing the professional qualification GH-500 Exam. We are very confident in the quality of GH-500 guide dumps. Our pass rate is high as 98% to 100%. You can totally rely on us.

Microsoft GitHub Advanced Security Sample Questions (Q36-Q41):

NEW QUESTION # 36
Which of the following steps should you follow to integrate CodeQL into a third-party continuous integration system? (Each answer presents part of the solution. Choose three.)

- A. Upload scan results
- B. Process alerts
- C. Write queries
- D. Install the CLI
- E. Analyze code

Answer: A,D,E

Explanation:

When integrating CodeQL outside of GitHub Actions (e.g., in Jenkins, CircleCI):
Install the CLI: Needed to run CodeQL commands.
Analyze code: Perform the CodeQL analysis on your project with the CLI.
Upload scan results: Export the results in SARIF format and use GitHub's API to upload them to your repo's security tab.
You don't need to write custom queries unless extending functionality. "Processing alerts" happens after GitHub receives the results.

NEW QUESTION # 37
What step is required to run a SARIF-compatible (Static Analysis Results Interchange Format) tool on GitHub Actions?

- A. Update the workflow to include a final step that uploads the results.
- B. The CodeQL action uploads the SARIF file automatically when it completes analysis.
- C. Use the CLI to upload results to GitHub.
- D. By default, the CodeQL runner automatically uploads results to GitHub on completion.

Answer: A

Explanation:
When using a SARIF-compatible tool within GitHub Actions, it's necessary to explicitly add a step in your workflow to upload the analysis results. This is typically done using the upload-sarif action, which takes the SARIF file generated by your tool and uploads it to GitHub for processing and display in the Security tab. Without this step, the results won't be available in GitHub's code scanning interface.

NEW QUESTION # 38
What do you need to do before you can define a custom pattern for a repository?

- A. Provide match requirements for the secret format.
- B. Enable secret scanning on the repository.
- C. Add a secret scanning custom pattern.
- D. Provide a regular expression for the format of your secret pattern.

Answer: B

Explanation:
Stack Overflow
Explanation:
Comprehensive and Detailed Explanation:
Before defining a custom pattern for secret scanning in a repository, you must enable secret scanning for that repository. Secret scanning must be active to utilize custom patterns, which allow you to define specific formats (using regular expressions) for secrets unique to your organization.
Once secret scanning is enabled, you can add custom patterns to detect and prevent the exposure of sensitive information tailored to your needs.

NEW QUESTION # 39
When using CodeQL, how does extraction for compiled languages work?

- A. By running directly on the source code
- B. By resolving dependencies to give an accurate representation of the codebase
- C. By generating one language at a time
- D. By monitoring the normal build process

Answer: D

Explanation:
For compiled languages, CodeQL performs extraction by monitoring the normal build process. This means it watches your usual build commands (like make, javac, or dotnet build) and extracts the relevant data from the actual build steps being executed.
CodeQL uses this information to construct a semantic database of the application.
This approach ensures that CodeQL captures a precise, real-world representation of the code and its behavior as it is compiled, including platform-specific configurations or conditional logic used during build.

NEW QUESTION # 40
In the pull request, how can developers avoid adding new dependencies with known vulnerabilities?

- A. Add Dependabot rules.
- B. Enable Dependabot alerts.
- C. Enable Dependabot security updates.
- D. Add a workflow with the dependency review action.

Answer: D

Explanation:
To detect and block vulnerable dependencies before merge, developers should use the Dependency Review GitHub Action in their pull request workflows. It scans all proposed dependency changes and flags any packages with known vulnerabilities.
This is a preventative measure during development, unlike Dependabot, which reacts after the fact.

NEW QUESTION # 41
......

For candidates who want to get the certificate of the exam, choosing a proper GH-500 learning material is important. We will provide you the GH-500 learning with high accuracy and high quality. If you fail to pass the exam, money back guarantee and it will returning to your account, and if you have any questions about the GH-500 Exam Dumps, our online service staff will help to solve any problem you have, just contact us without any hesitation.

GH-500 Detailed Study Dumps: https://www.testinsides.top/GH-500-dumps-review.html

BTW, DOWNLOAD part of TestInsides GH-500 dumps from Cloud Storage: https://drive.google.com/open?id=1EkxllAXSkmldxqE-btlEdYFMACVRw1oi