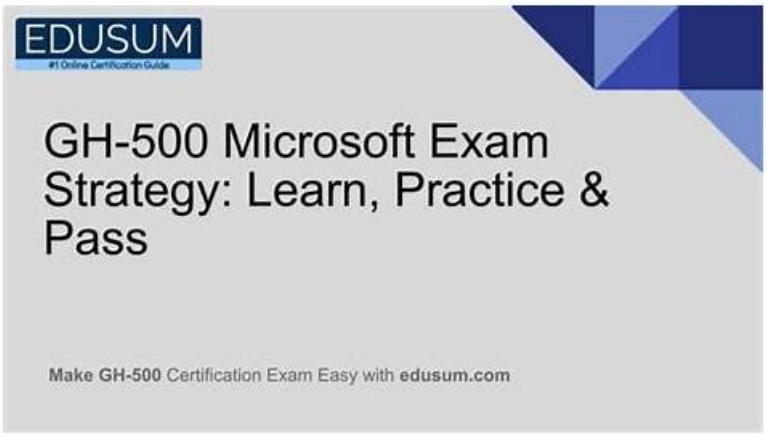
GH-500 Exam Dump - Valid GH-500 Test Pattern



DOWNLOAD the newest TrainingDump GH-500 PDF dumps from Cloud Storage for free: https://drive.google.com/open?id=1qeobjDg-Q1hLahWjx3EIg12t5ekOgOhz

Professional ability is very important both for the students and for the in-service staff because it proves their practical ability in the area they major in. Therefore choosing a certificate exam which boosts great values to attend is extremely important for them and the test GH-500 Certification is one of them. Passing the test certification can prove your outstanding major ability in some area and if you want to pass the test smoothly you'd better buy our GH-500 study materials.

TrainingDump Microsoft GH-500 exam training materials praised by the majority of candidates is not a recent thing. This shows TrainingDump Microsoft GH-500 exam training materials can indeed help the candidates to pass the exam. Compared to other questions providers, TrainingDump Microsoft GH-500 exam training materials have been far ahead. uestions broad consumer recognition and reputation, it has gained a public praise. If you want to participate in the Microsoft GH-500 Exam, quickly into TrainingDump website, I believe you will get what you want. If you miss you will regret, if you want to become a professional IT expert, then quickly add it to cart.

>> GH-500 Exam Dump <<

Valid GH-500 Test Pattern - GH-500 Premium Files

The TrainingDump Microsoft GH-500 exam dumps are ready for quick download. Just choose the right TrainingDump Microsoft GH-500 exam questions format and download it after paying an affordable TrainingDump GitHub Advanced Security (GH-500) practice questions charge and start this journey. Best of luck in Microsoft GH-500 exam and career!!!

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 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 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	• 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 5	• 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.

Microsoft GitHub Advanced Security Sample Questions (Q77-Q82):

NEW QUESTION # 77
When using CodeQL, how does extraction for compiled languages work?

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

Answer: B

Explanation:
For compiled languages, CodeQL performs extraction bymonitoring 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 # 78
Secret scanning will scan:

- A. External services.
- B. Any Git repository.
- C. The GitHub repository.
- D. A continuous integration system.

Answer: C

Explanation:
Secret scanning is a feature provided by GitHub that scans the contents of your GitHub repositories for known types of secrets, such as API keys and tokens. It operates within the GitHub environment and does not scan external systems, services, or repositories outside of GitHub. Its primary function is to prevent the accidental exposure of sensitive information within your GitHub-hosted code.

NEW QUESTION # 79
Where can you find a deleted line of code that contained a secret value?

- A. Insights
- B. Commits
- C. Dependency graph
- D. Issues

Answer: C

Explanation:
Deleted lines of code containing secrets in a GitHub repository can still be accessed through the dependency graph and other tools, even after deletion. The dependency graph analyzes package manifest files to identify dependencies, including those in deleted or private repositories. Anyone with access to the dependency graph can potentially view the list of dependencies and their transitive dependencies, potentially exposing leaked secrets if they were previously part of the codebase.

NEW QUESTION # 80
You are a maintainer of a repository and Dependabot notifies you of a vulnerability. Where could the vulnerability have been disclosed? (Each answer presents part of the solution. Choose two.)

- A. In the National Vulnerability Database
- B. In the dependency graph
- C. In manifest and lock files
- D. In security advisories reported on GitHub

Answer: A,D

Explanation:
Comprehensive and Detailed Explanation:
Dependabot alerts are generated based on data from various sources:
National Vulnerability Database (NVD): A comprehensive repository of known vulnerabilities, which GitHub integrates into its advisory database.
GitHub Docs
Security Advisories Reported on GitHub: GitHub allows maintainers and security researchers to report and discuss vulnerabilities, which are then included in the advisory database.
The dependency graph and manifest/lock files are tools used by GitHub to determine which dependencies are present in a repository but are not sources of vulnerability disclosures themselves.

NEW QUESTION # 81
By default, who will receive an e-mail when a secret has been detected in a repository? Each answer presents a complete solution. (Choose two.)

- A. users with the Admin repository role
- B. users with the Maintain repository role
- C. users with the Write repository role
- D. user who committed the secret
- E. security analyst

Answer: A,D

Explanation:
When a new secret is detected, GitHub notifies all users with access to security alerts for the repository according to their notification preferences.
These users include:
Repository administrators. [D]
Security managers.
Users with custom roles with read/write access
Organization owners and enterprise owners, if they are administrators of repositories where secrets were leaked Note Commit authors who've accidentally committed secrets will be notified, regardless of their notification preferences. [B]

NEW QUESTION # 82
.....

Looking for latest GH-500 exam questions? You can pass the certification exam easily with our GH-500 practice exam. With the help of our study guide, you will save lots of time to practice GH-500 vce pdf and boost confidence in solving the difficult questions. Our training materials can help you learn about the knowledge points of GH-500 Exam Collection and improve your technical problem-solving skills.

Disposable vapes

What's more, part of that TrainingDump GH-500 dumps now are free: https://drive.google.com/open?id=1qeobjDg-Q1hLahWjx3EIg12t5ekOgOhz