Microsoft GH-500 Actual Test Answers | GH-500 New Braindumps Book

DOWNLOAD the newest Real4dumps GH-500 PDF dumps from Cloud Storage for free: https://drive.google.com/open?id=1fX1c0k0_3gTCEVwl9i55xzpdwIaT_1zm

Similarly, Real4dumps provides you 1 year free updates after your purchase of Microsoft GH-500 practice tests. These updates will help you prepare well if the content of the exam changes. The GitHub Advanced Security (GH-500) demo of the practice exams is totally free and it helps you in examining the GH-500 study materials.

We will refund your money if you fail to pass the exam if you buy GH-500 exam dumps from us, and no other questions will be asked. We are famous for high pass rate, with the pass rate is 98.75%, we can ensure you that you pass the exam and get the corresponding certificate successfully. In addition, GH-500 Exam Dumps of us will offer you free update for 365 days, and our system will send the latest version of GH-500 exam braindumps to your email automatically. We also have online service stuff, and if you have any questions just contact us.

>> Microsoft GH-500 Actual Test Answers <<

Microsoft GH-500 New Braindumps Book, GH-500 Latest Test Practice

All these three Real4dumps GitHub Advanced Security (GH-500) exam questions formats are easy to use and perfectly work with all devices, operating systems, and the latest web browsers. So rest assured that with the Real4dumps GH-500 Exam Dumps you will get everything that you need to learn, prepare and pass the challenging GitHub Advanced Security (GH-500) exam with good scores.

Microsoft GH-500 Exam Syllabus Topics:

Topic	Details

Topic 1	• 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 2	• 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 3	• 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 4	• 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 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 (Q30-Q35):

NEW QUESTION # 30

Why should you dismiss a code scanning alert?

- A. To prevent developers from introducing new problems
- B. If you fix the code that triggered the alert
- C. If there is a production error in your code
- D. If it includes an error in code that is used only for testing

Answer: D

Explanation:
You should dismiss a code scanning alert if the flagged code is not a true security concern, such as:
Code in test files
Code paths that are unreachable or safe by design
False positives from the scanner
Fixing the code would automatically resolve the alert - not dismiss it. Dismissing is for valid exceptions or noise reduction.

NEW QUESTION # 31
When secret scanning detects a set of credentials on a public repository, what does GitHub do?

- A. It notifies the service provider who issued the secret.
- B. It scans the contents of the commits for additional secrets.
- C. It sends a notification to repository members.
- D. It displays a public alert in the Security tab of the repository.

Answer: A

Explanation:
When a public repository contains credentials that match known secret formats, GitHub will automatically notify the service provider that issued the secret. This process is known as "secret scanning partner notification". The provider may then revoke the secret or contact the user directly.
GitHub does not publicly display the alert and does not send internal repository notifications for public detections.

NEW QUESTION # 32
Which security feature shows a vulnerable dependency in a pull request?

- A. Dependency review
- B. Dependency graph
- C. The repository's Security tab
- D. Dependabot alert

Answer: A

Explanation:
Dependency review runs as part of a pull request and shows which dependencies are being added, removed, or changed - and highlights vulnerabilities associated with any added packages.
It works in real-time and is specifically designed for use during pull request workflows.
The dependency graph is an overview, Dependabot alerts notify post-merge, and the Security tab shows the aggregated alert list.

NEW QUESTION # 33
Where in the repository can you give additional users access to secret scanning alerts?

- A. Settings
- B. Security
- C. Insights
- D. Secrets

Answer: A

Explanation:
To grant specific users access to view and manage secret scanning alerts, you do this via the Settings tab of the repository. From there, under the "Code security and analysis" section, you can add individuals or teams with roles such as security manager.
The Security tab only displays alerts; access control is handled in Settings.

NEW QUESTION # 34
Which of the following features helps to prioritize secret scanning alerts that present an immediate risk?

- A. Push protection
- B. Non-provider patterns
- C. Secret validation
- D. Custom pattern dry runs

Answer: C

Explanation:
Secret validation checks whether a secret found in your repository is still valid and active with the issuing provider (e.g., AWS, GitHub, Stripe). If a secret is confirmed to be active, the alert is marked as verified, which means it's considered a high-priority issue because it presents an immediate security risk.
This helps teams respond faster to valid, exploitable secrets rather than wasting time on expired or fake tokens.

NEW QUESTION # 35
.....

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, Disposable vapes

BTW, DOWNLOAD part of Real4dumps GH-500 dumps from Cloud Storage: https://drive.google.com/open?id=1fX1c0k0_3gTCEVwl9i55xzpdwIaT_1zm