EDUSUM

#1 Online Certification Guide

MICROSOFT GH-500

Microsoft GitHub Advanced Security Certification Questions & Answers

Exam Summary – Syllabus –Questions

GH-500

Microsoft GitHub Advanced Security

75 Questions Exam – 700 / 1000 Cut Score – Duration of 100 minutes

Table of Contents:

Know Your GH-500 Certification Well:

The GH-500 is best suitable for candidates who want to gain knowledge in the Microsoft GitHub. Before you start your GH-500 preparation you may struggle to get all the crucial GitHub Advanced Security materials like GH-500 syllabus, sample questions, study guide.

But don't worry the GH-500 PDF is here to help you prepare in a stress free manner.
The PDF is a combination of all your queries like-

- What is in the GH-500 syllabus?
- How many questions are there in the GH-500 exam?
- Which Practice test would help me to pass the GH-500 exam at the first attempt?

Passing the GH-500 exam makes you Microsoft GitHub Advanced Security. Having the GitHub Advanced Security certification opens multiple opportunities for you. You can grab a new job, get a higher salary or simply get recognition within your current organization.

Microsoft GH-500 GitHub Advanced Security Certification Details:

Exam Name	Microsoft GitHub Advanced Security
Exam Code	GH-500
Exam Price	$99 (USD)
Duration	100 mins
Number of Questions	75
Passing Score	700 / 1000
Books / Training	**GH-500T00-A: GitHub Advanced Security**
Schedule Exam	**Pearson VUE**
Sample Questions	**Microsoft GitHub Advanced Security Sample Questions**
Practice Exam	**Microsoft GH-500 Certification Practice Exam**

GH-500 Syllabus:

Topic	Details
Describe the GHAS security features and functionality (15%)	
Contrast GHAS features and their role in the security ecosystem	- Differentiate the security features that come automatically for open source projects, and what features are available when GHAS is paired with GHEC or GHES - Describe the features and benefits of Security Overview - Describe the differences between secret scanning and code scanning - Describe how secret scanning, code scanning, and Dependabot create a more secure software development life cycle - Contrast a security scenario with isolated security review and an advanced scenario, with security integrated into each step of the software development life cycle
Explain and use specific GHAS features	- Describe how vulnerable dependencies are identified (by looking at the manifest files and comparing with databases of known vulnerabilities) - Choose how to act on alerts from GHAS - Explain the implications of ignoring an alert - Explain the role of a developer when they discover a security alert - Describe the differences in access management to view alerts for different security features - Identify where to use Dependabot alerts in the software development lifecycle
Configure and use secret scanning (15%)	
Configure and use Secret Scanning	- Describe secret scanning - Describe push protection - Describe validity checks - Contrast secret scanning availability for public and private repositories

Topic	Details
	- Enable secret scanning for private repositories - Pick an appropriate response to a secret scanning alert - Determine if an alert is generated for a given secret, pattern, or service provider - Determine if a given user role will see secret scanning alerts and how they will be notified
Customize default secret scanning behavior	- Configure the recipients of a secret scanning alert (also includes how to provide access to members and teams other than admins) - Exclude certain files from being scanned for secrets - Enable custom secret scanning for a repository
Configure and use Dependabot and Dependency Review (35%)	
Describe tools for managing vulnerabilities in dependencies	- Define the dependency graph - Describe how the dependency graph is generated - Describe what a Software Bill of Materials (SBOM) is, and the SBOM format used by GitHub - Define a dependency vulnerability - Describe Dependabot alerts - Describe Dependabot security updates - Describe Dependency Review - Describe how alerts are generated for vulnerable dependencies (driven from the dependency graph, sourced from the GitHub Advisory Database) - Describe the difference between Dependabot and Dependency Review
Enable and configure tools for managing vulnerable dependencies	- Identify the default settings for Dependabot alerts in public and private repositories - Identify the permissions and roles required to enable Dependabot alerts - Identify the permissions and roles required to view Dependabot alerts

Topic	Details
	- Enable Dependabot alerts for private repositories - Enable Dependabot alerts for organizations - Create a valid Dependabot configuration file to group security updates - Create a Dependabot Rule to auto-dismiss low severity alerts until a patch is available - Create a Dependency Review GitHub Actions workflow - Configure license checks and custom severity thresholds in a Dependency Review workflow - Configure notifications for vulnerable dependencies
Identify and remediate vulnerable dependencies	- Identify a vulnerable dependency from a Dependabot alert - Identify vulnerable dependencies from a pull request - Enable Dependabot security updates - Remedy a vulnerability from a Dependabot alert in the Security tab (could include updating or removing the dependency) - Remedy a vulnerability from a Dependabot alert in the context of a pull request (could include updating or removing the dependency) - Take action on any Dependabot alerts by testing and merging pull requests
Configure and use Code Scanning with CodeQL (25%)	
Use code scanning with third-party tools	- Enable code scanning for use with a third-party analysis - Contrast the steps for using CodeQL versus third party analysis when enabling code scanning - Contrast how to implement CodeQL analysis in a GitHub Actions workflow versus a third-party CI tool - Upload 3rd party SARIF results via the SARIF endpoint

Topic	Details
Describe and enable code scanning	- Describe how code scanning fits in the software development life cycle - Contrast the frequency of code scanning workflows (scheduled versus triggered by events) - Choose a triggering event for a given development pattern (for example, in a pull request and for specific files) - Edit the default template for Actions workflow to fit an active, open source, production repository - Describe how to view code scanning results from CodeQL analysis - Troubleshoot a failing code scanning workflow using CodeQL, including creating or changing a custom configuration in the CodeQL workflow - Follow the data flow through code using the show paths experience - Explain the reason for a code scanning alert given documentation linked from the alert - Determine if and why a code scanning alert needs to be dismissed - Describe potential shortfalls in CodeQL via model of compilation and language support - Explain the purpose of defining a SARIF category
Describe GitHub Advanced Security best practices, results, and how to take corrective measures (10%)	
GitHub Advanced Security results & best practices	- Use a Common Vulnerabilities and Exposures (CVE) and Common Weakness Enumeration (CWE) to describe a GitHub Advanced Security alert and list potential remediation - Describe the decision-making process for closing and dismissing security alerts (documenting the dismissal, making a decision based on data) - Describe the default CodeQL query suites - Describe how CodeQL analyzes code and

Topic	Details
	produces results, including differences between compiled and interpreted language - Determine the roles and responsibilities of development and security teams on a software development workflow - Describe how the severity threshold for code scanning pull request status checks can be changed - Explain how filters and sorting can be used to prioritize secret scanning remediation (validity:active) - Explain how CodeQL & Dependency Review workflows can be enforced with Repository Rulesets - Describe how code scanning can be configured to identify and remediate vulnerabilities earlier (scanning upon pull request) - Describe how secret scanning can be configured to identify and remediate vulnerabilities earlier (enabling push protection) - Describe how dependency analysis can be configured to identify and remediate vulnerabilities earlier (enable dependency review to scan upon pull request)

Microsoft GH-500 Sample Questions:

Question: 1

What is the difference between scheduled versus triggered events in code scanning?

a) Scheduled events are more difficult to configure than triggered events.
b) Scheduled events run based on a specified schedule and triggered events run on code events such as a push.
c) Triggered events run less frequently than scheduled events.
d) Scheduled events can only be set up by administrators.

Answer: b

Question: 2

How does GitHub Advanced Security (GHAS) help integrate security into each step of the software development life cycle?

a) By providing a comprehensive dashboard summarizing the security status of the repository.
b) By automating security checks with every pull request, surfacing issues in the context of the development workflow.
c) By generating alerts for outdated dependencies in a project.
d) By providing access to curated security intelligence from millions of developers and security researchers around the world.

Answer: b

Question: 3

Which two pieces of information should be included in a security advisory?

a) Product affected and severity.
b) Severity and exposure list.
c) Administrator name and severity.
d) Exposures list and administrator name.

Answer: a

Question: 4

How does Dependabot use the dependency graph in GitHub Advanced Security (GHAS)?

a) To identify and address security vulnerabilities in the codebase.
b) To automatically update project dependencies to their latest, secure versions.
c) To generate alerts for potential security vulnerabilities in project dependencies.
d) To cross-reference dependency data with the GitHub Advisory Database.

Answer: d

Question: 5

When code scanning is enabled, what is one default event that triggers a scan?

a) Creating a new branch.
b) Deleting a branch.
c) Pushing a change.
d) Merging a branch.

Answer: c

Question: 6

Which of the following is NOT an action a user can take when they receive an alert from GitHub Advanced Security (GHAS)?

a) Ignore the alert.
b) Dismiss the alert.
c) Report the alert to GitHub.
d) Investigate the alert and take appropriate action.

Answer: c

Question: 7

What are the permissions and roles required to enable Dependabot alerts on GitHub?

a) Only users with admin access to a repository can enable Dependabot alerts.
b) Only repository maintainers can enable Dependabot alerts.
c) Only users with write access to a repository can enable Dependabot alerts.
d) Any user with access to a repository can enable Dependabot alerts.

Answer: a

Question: 8

What are the default settings for Dependabot alerts in public and private repositories on GitHub?

a) Dependabot alerts are enabled by default for public repositories and disabled by default for private repositories.
b) Dependabot alerts are disabled by default for both public and private repositories.
c) Dependabot alerts are enabled by default for both public and private repositories.
d) Dependabot alerts are disabled by default for public repositories and enabled by default for private repositories.

Answer: a

Question: 9

How does secret scanning availability differ for public and private repositories on GitHub?

a) Secret scanning is only available for public repositories.
b) Secret scanning is only available for private repositories.
c) Secret scanning is available for both public and private repositories, but the configuration options may differ.
d) Secret scanning is not available for either public or private repositories.

Answer: c

Question: 10

What is the exportable SBOM format created by the dependency graph on GitHub?

a) CycloneDX.
b) SPDX.
c) SWID.
d) All of the above.

Answer: d

Study Guide to Crack Microsoft GitHub Advanced Security GH-500 Exam:

- Getting details of the GH-500 syllabus, is the first step of a study plan. This pdf is going to be of ultimate help. Completion of the syllabus is must to pass the GH-500 exam.
- Making a schedule is vital. A structured method of preparation leads to success. A candidate must plan his schedule and follow it rigorously to attain success.
- Joining the Microsoft provided training for GH-500 exam could be of much help. If there is specific training for the exam, you can discover it from the link above.
- Read from the GH-500 sample questions to gain your idea about the actual exam questions. In this PDF useful sample questions are provided to make your exam preparation easy.
- Practicing on GH-500 practice tests is must. Continuous practice will make you an expert in all syllabus areas.

Reliable Online Practice Test for GH-500 Certification

Make EduSum.com your best friend during your Microsoft GitHub Advanced Security exam preparation. We provide authentic practice tests for the GH-500 exam. Experts design these online practice tests, so we can offer you an exclusive experience of taking the actual GH-500 exam. We guarantee you 100% success in your first exam attempt if you continue practicing regularly. Don't bother if you don't get 100% marks in initial practice exam attempts. Just utilize the result section to know your strengths and weaknesses and prepare according to that until you get 100% with our practice tests. Our evaluation makes you confident, and you can score high in the GH-500 exam.

Start Online practice of GH-500 Exam by visiting URL
https://www.edusum.com/microsoft/gh-500-microsoft-github-advanced-security