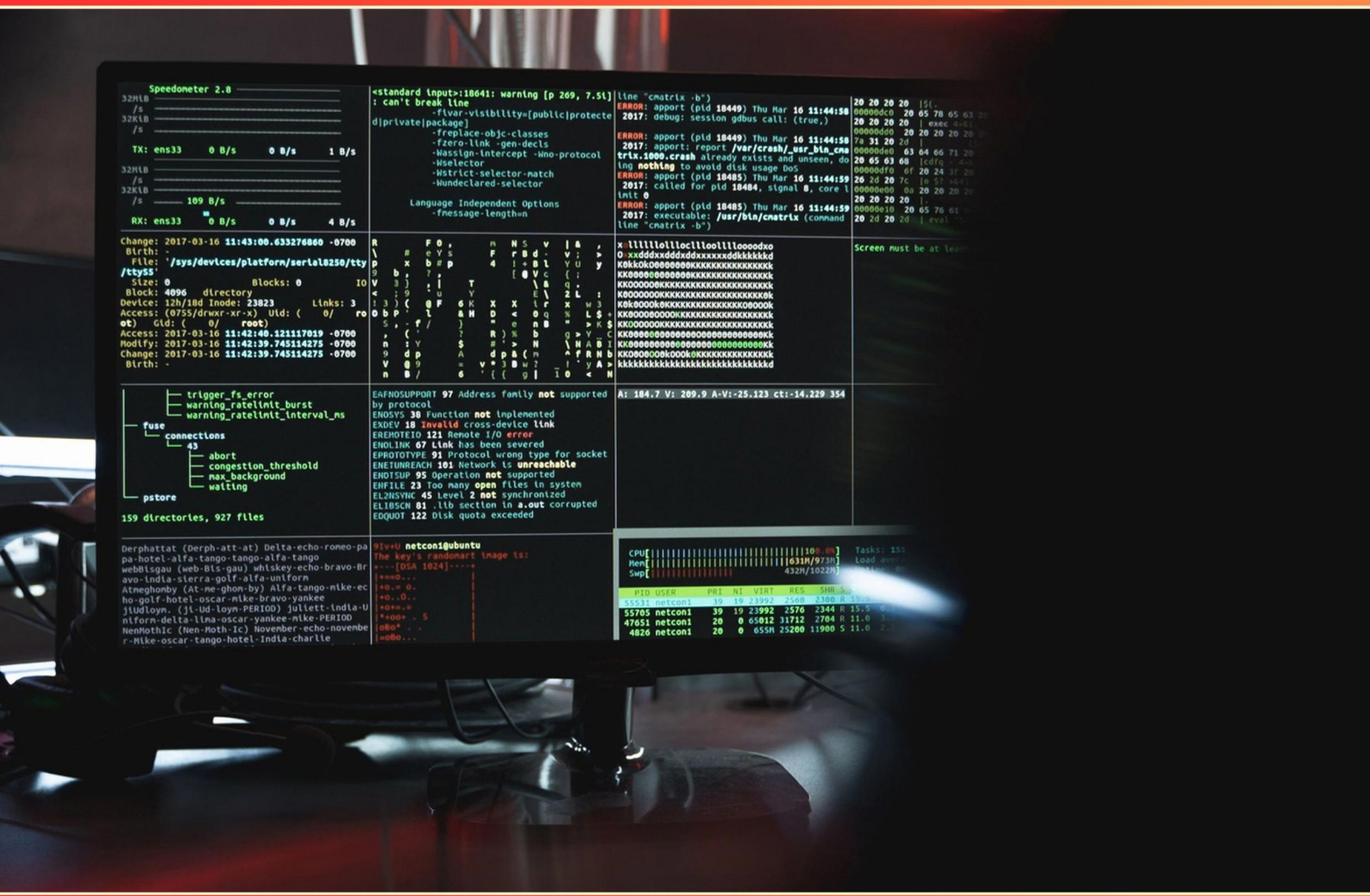
GOOGLE CLOUD DEVOPS CERTIFICATION PRACTICE TEST (SAMPLE)

STUDY GUIDE



SAMPLE STUDY GUIDE WITH LIMITED QUESTIONS

VISIT US ONLINE FOR THE FULL VERSION STUDY GUIDE

https://googleclouddevops.examzify.com

Table of Contents

Copyright 1

Table of Contents 2

Introduction 3

How to Use This Guide 4

Questions 5

Answers 8

Explanations 10

Next Steps 16

Introduction

Preparing for a certification exam can feel overwhelming, but with the right tools, it becomes an opportunity to build confidence, sharpen your skills, and move one step closer to your goals. At Examzify, we believe that effective exam preparation isn't just about memorization, it's about understanding the material, identifying knowledge gaps, and building the test-taking strategies that lead to success.

This guide was designed to help you do exactly that.

Whether you're preparing for a licensing exam, professional certification, or entry-level qualification, this book offers structured practice to reinforce key concepts. You'll find a wide range of multiple-choice questions, each followed by clear explanations to help you understand not just the right answer, but why it's correct.

The content in this guide is based on real-world exam objectives and aligned with the types of questions and topics commonly found on official tests. It's ideal for learners who want to:

- Practice answering questions under realistic conditions,
- Improve accuracy and speed,
- Review explanations to strengthen weak areas, and
- Approach the exam with greater confidence.

We recommend using this book not as a stand-alone study tool, but alongside other resources like flashcards, textbooks, or hands-on training. For best results, we recommend working through each question, reflecting on the explanation provided, and revisiting the topics that challenge you most.

Remember: successful test preparation isn't about getting every question right the first time, it's about learning from your mistakes and improving over time. Stay focused, trust the process, and know that every page you turn brings you closer to success.

Let's begin.

How to Use This Guide

This guide is designed to help you study more effectively and approach your exam with confidence. Whether you're reviewing for the first time or doing a final refresh, here's how to get the most out of your Examzify study guide:

1. Start with a Diagnostic Review

Skim through the questions to get a sense of what you know and what you need to focus on. Your goal is to identify knowledge gaps early.

2. Study in Short, Focused Sessions

Break your study time into manageable blocks (e.g. 30 – 45 minutes). Review a handful of questions, reflect on the explanations.

3. Learn from the Explanations

After answering a question, always read the explanation, even if you got it right. It reinforces key points, corrects misunderstandings, and teaches subtle distinctions between similar answers.

4. Track Your Progress

Use bookmarks or notes (if reading digitally) to mark difficult questions. Revisit these regularly and track improvements over time.

5. Simulate the Real Exam

Once you're comfortable, try taking a full set of questions without pausing. Set a timer and simulate test-day conditions to build confidence and time management skills.

6. Repeat and Review

Don't just study once, repetition builds retention. Re-attempt questions after a few days and revisit explanations to reinforce learning. Pair this guide with other Examzify tools like flashcards, and digital practice tests to strengthen your preparation across formats.

There's no single right way to study, but consistent, thoughtful effort always wins. Use this guide flexibly, adapt the tips above to fit your pace and learning style. You've got this!

Questions

1. To decrease build time while using Cloud Build, what is a recommended action?

A. Use larger Cloud Build virtual machines by adjusting the machine-type option.

B. Run multiple Jenkins agents to parallelize the build.

C. Use multiple smaller build steps to minimize execution time.

D. Use Cloud Storage to cache intermediate artifacts.

2. How can you identify which downstream service is causing response delays in an application on GKE?

A. Analyze VPC flow logs

B. Create a data analysis pipeline

C. Use distributed tracing frameworks like OpenTelemetry

D. Investigate service liveness and readiness probes

3. Which approach should you choose to automate deployment whenever application images are updated?

A. Use Cloud Build to trigger a Spinnaker pipeline.

B. Utilize Cloud Pub/Sub to trigger a Spinnaker pipeline.

C. Create a custom builder in Cloud Build for Jenkins.

D. Trigger a custom deployment service in GKE with Pub/Sub.

4. Which tool should you use for validating and enforcing security policies on container images?

A. Cloud Build service account permissions.

B. Kritis for image scanning.

C. Cloud Security Command Center.

D. Binary Authorization in GKE clusters.

5. For a semi-annual capacity planning exercise expecting user growth, what is the appropriate first step?

A. Verify the maximum node pool size and enable horizontal pod autoscaler for load testing.

B. Assume existing configurations will handle the expected growth automatically.

C. Consider the current utilization rate sufficient for anticipated growth.

D. Reduce resource allocation to save costs until actual demand increases.

6. What is an effective way to proactively identify performance issues in dependent applications for a Node.js application on GKE?

A. Instrument all applications with Cloud Profiler

B. Instrument with Cloud Trace and review HTTP requests

C. Use Cloud Debugger for execution reviews

D. Log HTTP request times and analyze with Cloud Logging

7. To test applications under full production load before launching, what methodology should be applied?

A. Use A/B testing with blue/green deployment.

B. Employ canary testing with continuous deployment.

C. Utilize canary testing with rolling updates deployment.

D. Implement shadow testing with continuous deployment.

8. What storage solution should you use for container images and Helm charts that integrates with Google Cloud services?

A. Run an open-source container registry server in GKE with RBAC

B. Use Docker to configure a Cloud Storage driver

C. Use Artifact Registry as an OCI-based container registry

D. Configure Container Registry for both Helm charts and images

9. What action should you take to enhance network throughput performance for a service relying on n2-standard-2 Compute Engine instances?

A. Add extra Network Interface Controllers (NICs) to your VMs

B. Implement a Cloud NAT gateway and connect it to the subnet

C. Alter the machine type for your VMs to n2-standard-8

D. Deploy the Ops Agent to enable additional monitoring metrics

10. What approach would you use to implement a secure method for developers to access application logs on Google Cloud Platform?

A. Deploy the Cloud logging agent to the application servers. Give the developers the IAM Logs Viewer role to access and view logs.

B. Deploy the Cloud logging agent to the application servers. Give the developers the IAM Logs Private Logs Viewer role to access and view logs.

C. Deploy the Cloud monitoring agent to the application servers. Give the developers the IAM Monitoring Viewer role to access and view metrics.

D. Deploy logs to a separate GCP project with restricted access for developers.

Answers

1. D
2. C
3. B
4. D
5. A
6. B
7. D
8. C
9. C
10. A

Explanations

1. **To decrease build time while using Cloud Build, what is a recommended action?**

   A. Use larger Cloud Build virtual machines by adjusting the machine-type option.

   B. Run multiple Jenkins agents to parallelize the build.

   C. Use multiple smaller build steps to minimize execution time.

   D. Use Cloud Storage to cache intermediate artifacts.

Using Cloud Storage to cache intermediate artifacts is a recommended practice to decrease build time when utilizing Cloud Build. By caching intermediate artifacts, subsequent builds can skip redundant operations and reuse previously built components. This technique is particularly effective in large projects where certain dependencies or build results do not change frequently. Instead of rebuilding everything from scratch on each build, the process can leverage cached artifacts, significantly reducing the overall time spent. In contrast, the other options may not be as effective. Adjusting the machine type to a larger virtual machine could lead to faster computation, but this approach increases costs and may not always result in a linear decrease in build time. Running multiple Jenkins agents can parallelize builds, but it might complicate the build process and does not directly address the efficiency of individual builds in the Cloud Build environment. Using multiple smaller build steps could sometimes lead to improvements, but such an approach may also create overhead in managing these steps and may not guarantee a reduction in total execution time.

2. **How can you identify which downstream service is causing response delays in an application on GKE?**

   A. Analyze VPC flow logs

   B. Create a data analysis pipeline

   C. Use distributed tracing frameworks like OpenTelemetry

   D. Investigate service liveness and readiness probes

Using distributed tracing frameworks like OpenTelemetry is an effective method for identifying which downstream service is causing response delays in an application on Google Kubernetes Engine (GKE). Distributed tracing allows you to track individual requests as they flow through the various services that make up your application. It helps in capturing the timing information for each segment of the request lifecycle across multiple microservices. By utilizing OpenTelemetry, you can visualize the entire flow of a request, see where the bottlenecks are occurring, and pinpoint which specific downstream service is introducing latency. This detailed view of the interaction between services enables you to diagnose performance issues more accurately and optimize your application accordingly. In contrast, analyzing VPC flow logs provides network-level visibility but does not give detailed insights into how different services are interacting or where delays may be occurring in the request handling process. Creating a data analysis pipeline may assist with data collection and analysis but doesn't directly address tracing service interactions. Investigating service liveness and readiness probes is crucial for ensuring service availability, but this approach does not provide information regarding response times or delays between services.

3. Which approach should you choose to automate deployment whenever application images are updated?

A. Use Cloud Build to trigger a Spinnaker pipeline.

B. Utilize Cloud Pub/Sub to trigger a Spinnaker pipeline.

C. Create a custom builder in Cloud Build for Jenkins.

D. Trigger a custom deployment service in GKE with Pub/Sub.

Utilizing Cloud Pub/Sub to trigger a Spinnaker pipeline is an effective approach for automating deployment whenever application images are updated. This method allows for a decoupled and event-driven architecture, where Cloud Pub/Sub acts as a messaging service that can notify Spinnaker of changes in application images. When a new image is built and pushed to a container registry, a message can be published to a Pub/Sub topic. Spinnaker can be configured to listen to this topic and initiate a deployment pipeline in response to those notifications. This process ensures that deployments are automated and can be streamlined to respond quickly to changes, improving efficiency and reducing manual intervention. The other options provided may include methods that involve additional complexity or less direct integration with automated deployment processes. For instance, while Cloud Build can trigger pipelines, it typically requires a specific build configuration, and directly using a custom builder for Jenkins involves managing an additional tool rather than leveraging Spinnaker directly. Triggering a custom deployment service in GKE through Pub/Sub could work, but it introduces another layer of service rather than making full use of Spinnaker's robust deployment capabilities. Hence, using Cloud Pub/Sub directly with Spinnaker is the most straightforward and effective method for triggering deployments based on image

4. Which tool should you use for validating and enforcing security policies on container images?

A. Cloud Build service account permissions.

B. Kritis for image scanning.

C. Cloud Security Command Center.

D. Binary Authorization in GKE clusters.

Binary Authorization in GKE clusters is a powerful tool specifically designed to validate and enforce security policies on container images. It acts as a deployment safety mechanism that ensures only trusted container images are deployed to your Google Kubernetes Engine (GKE) clusters. By defining policies that require certain criteria to be met—such as having undergone security scans, being signed by trusted authorities, or meeting compliance standards—Binary Authorization helps prevent vulnerabilities introduced by malicious or unverified images. The use of Binary Authorization offers a flexible yet robust approach to maintaining a secure deployment process within Kubernetes, as it integrates seamlessly with the CI/CD pipelines. This tool effectively enhances the security posture of your applications running in GKE by ensuring that only those images that have passed predefined security checks are allowed to be deployed. Other choices, while important in their own capacities, do not specifically enforce security policies during the deployment phase of container images. For instance, the Cloud Build service account permissions focus more on access control rather than on validation of images themselves. Kritis, which provides image scanning and compliance checks, is relevant to image security but does not enforce deployment policies directly. Meanwhile, Cloud Security Command Center is a broader security management tool designed for monitoring and managing security across Google Cloud resources rather than specifically validating container images

5. For a semi-annual capacity planning exercise expecting user growth, what is the appropriate first step?

A. Verify the maximum node pool size and enable horizontal pod autoscaler for load testing.

B. Assume existing configurations will handle the expected growth automatically.

C. Consider the current utilization rate sufficient for anticipated growth.

D. Reduce resource allocation to save costs until actual demand increases.

The first step in a semi-annual capacity planning exercise anticipating user growth should focus on verifying the maximum node pool size and enabling horizontal pod autoscaler for load testing. This approach is key because it ensures that the infrastructure is capable of scaling to accommodate the projected increase in users. By checking the node pool size, you can determine the upper limits of your current infrastructure. Enabling a horizontal pod autoscaler allows for the dynamic adjustment of the number of pods based on real-time load, ensuring that your application can handle increased traffic efficiently. Conducting load testing in this context helps simulate the expected user growth and validates whether the current architecture can support that growth without service degradation. This initial step is critical to identifying potential bottlenecks and understanding resource needs before actual user growth occurs, allowing for any necessary adjustments in advance. It lays the groundwork for informed decision-making in scaling resources effectively to meet future demand.

6. What is an effective way to proactively identify performance issues in dependent applications for a Node.js application on GKE?

A. Instrument all applications with Cloud Profiler

B. Instrument with Cloud Trace and review HTTP requests

C. Use Cloud Debugger for execution reviews

D. Log HTTP request times and analyze with Cloud Logging

Instrumenting with Cloud Trace and reviewing HTTP requests is particularly effective for proactively identifying performance issues in dependent applications for a Node.js application on Google Kubernetes Engine (GKE) because it provides end-to-end tracking of requests as they propagate through various services. Cloud Trace captures latency data by allowing you to visualize the path of requests through your system, making it easier to detect where delays are occurring. This visibility into the performance of each component helps pinpoint which parts of your application may be causing bottlenecks or slowdowns in response times. By analyzing the traces collected, you can identify patterns, such as particular endpoints that may be slowing down interactions with dependent services. This proactive approach can lead you to optimize not only the Node.js application but also its dependent microservices. Monitoring the latency and understanding how various services interact provides insight into potential areas for improvement, allowing for timely adjustments before issues significantly impact users. Other methods, while valuable, don't provide the same level of insight into the interdependencies and performance metrics across services. For example, while Cloud Profiler can help in profiling resource usage at a granular code level, it does not focus on the performance of requests across services. Cloud Debugger is primarily used for inspecting the state of an application during execution, which

7. To test applications under full production load before launching, what methodology should be applied?

A. Use A/B testing with blue/green deployment.

B. Employ canary testing with continuous deployment.

C. Utilize canary testing with rolling updates deployment.

D. Implement shadow testing with continuous deployment.

The methodology that best fits the requirement of testing applications under full production load before launching is shadow testing with continuous deployment. Shadow testing allows you to run a new version of an application in parallel to the current production version without affecting the actual user experience. This means you can monitor the performance of the new version under real traffic conditions while the existing version continues to serve users. This approach is particularly advantageous because it simulates production load and gives you the opportunity to identify any issues that might only arise under full production conditions. You can analyze metrics and logs generated by the shadow version, gain insights into how it behaves under pressure, and ensure it can handle the expected user load before the actual launch. Continuous deployment supports this methodology by allowing you to release updates to production systems automatically, maintaining a fast and efficient pipeline for your application without downtime. By using shadow testing in conjunction with continuous deployment, you can ensure that any potential issues are addressed before the application goes live to all users, which enhances reliability and user satisfaction.

8. What storage solution should you use for container images and Helm charts that integrates with Google Cloud services?

A. Run an open-source container registry server in GKE with RBAC

B. Use Docker to configure a Cloud Storage driver

C. Use Artifact Registry as an OCI-based container registry

D. Configure Container Registry for both Helm charts and images

Using Artifact Registry as an OCI-based container registry is the optimal choice for storing container images and Helm charts within Google Cloud services. Artifact Registry provides a fully integrated solution specifically designed for managing and storing container images as well as various package formats like Helm charts. It supports the OCI (Open Container Initiative) image format, ensuring compatibility and adherence to open standards for container distribution. Furthermore, Artifact Registry offers enhanced security features, such as vulnerability scanning and fine-grained IAM (Identity and Access Management) controls, allowing organizations to manage permissions effectively. This integration with Google Cloud's security and identity services makes it easier to enforce policies across projects and teams. Additionally, Artifact Registry simplifies workflows by providing features such as automatic versioning and management of container images, which are critical for DevOps practices where consistent and reliable deployments are essential. Its ability to integrate seamlessly with other Google Cloud services enhances operational efficiency, making it the preferred choice compared to other options. In contrast, running an open-source container registry server in Google Kubernetes Engine (GKE) would require more management overhead as you would need to handle the deployment, scaling, and maintenance of the server. Using Docker with a Cloud Storage driver doesn't provide the same level of integration or feature set specific to container management that Artifact

9. What action should you take to enhance network throughput performance for a service relying on n2-standard-2 Compute Engine instances?

A. Add extra Network Interface Controllers (NICs) to your VMs

B. Implement a Cloud NAT gateway and connect it to the subnet

C. Alter the machine type for your VMs to n2-standard-8

D. Deploy the Ops Agent to enable additional monitoring metrics

To enhance network throughput performance for a service relying on n2-standard-2 Compute Engine instances, altering the machine type to n2-standard-8 is the most effective approach. This is because Google Cloud's Compute Engine assigns resources such as CPU and memory based on the selected machine type. The n2-standard-2 machine type provides a defined level of network resources and throughput, which may not be sufficient for high-demand applications. Switching to a n2-standard-8 machine type increases the number of virtual CPUs and memory available to the instance, which can significantly boost not only overall compute performance but also the network throughput capabilities associated with that instance. Increasing the machine type takes advantage of Google Cloud's infrastructure, which can offer more network bandwidth as the resources increase, thereby improving the throughput for your service. This is particularly important for workloads that are sensitive to network performance, such as data-intensive applications or microservices requiring substantial communication across instances. Other strategies, such as adding additional Network Interface Controllers or implementing Cloud NAT, may help under certain conditions but do not fundamentally change the resource limitations of a specific VM's capability. Similarly, deploying the Ops Agent would provide better monitoring insights but wouldn't inherently improve throughput performance for network traffic. Therefore, changing the machine

10. What approach would you use to implement a secure method for developers to access application logs on Google Cloud Platform?

A. Deploy the Cloud logging agent to the application servers. Give the developers the IAM Logs Viewer role to access and view logs.

B. Deploy the Cloud logging agent to the application servers. Give the developers the IAM Logs Private Logs Viewer role to access and view logs.

C. Deploy the Cloud monitoring agent to the application servers. Give the developers the IAM Monitoring Viewer role to access and view metrics.

D. Deploy logs to a separate GCP project with restricted access for developers.

Implementing a secure method for developers to access application logs on Google Cloud Platform involves ensuring that they have the appropriate permissions while also following the principle of least privilege. The chosen approach of deploying the Cloud logging agent to the application servers and granting developers the IAM Logs Viewer role is effective because it provides the necessary access to view logs without over-permissioning. The IAM Logs Viewer role allows developers to read logs from Cloud Logging. This role specifically enables them to see the logs associated with the resources managed in the same project where the logging agent is deployed. By using this role, you ensure that developers can access essential log information to troubleshoot applications or monitor performance without granting them broader permissions that could pose security risks. Other approaches listed may not align with best practices for security and management of logs. For instance, using the IAM Logs Private Logs Viewer role, while it might seem to provide an enhanced security layer, can limit log visibility and may not be necessary if developers need basic log access. Additionally, deploying the Cloud monitoring agent and granting the IAM Monitoring Viewer role focuses on metrics rather than logs, which does not address the requirement for log access. Finally, moving logs to a separate GCP project with restricted access might complicate access management and impede developers from efficiently working

Next Steps

Congratulations on reaching the final section of this guide. You've taken a meaningful step toward passing your certification exam and advancing your career.

As you continue preparing, remember that consistent practice, review, and self-reflection are key to success. Make time to revisit difficult topics, simulate exam conditions, and track your progress along the way.

If you need help, have suggestions, or want to share feedback, we'd love to hear from you. Reach out to our team at hello@examzify.com.

Or visit your dedicated course page for more study tools and resources:

https://googleclouddevops.examzify.com

We wish you the very best on your exam journey. You've got this!