Free Questions for Professional-Cloud-DevOps-Engineer by vceexamstest

Shared by Blackwell on 29-01-2024

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

You are leading a DevOps project for your organization. The DevOps team is responsible for managing the service infrastructure and being on-call for incidents. The Software Development team is responsible for writing, submitting, and reviewing code. Neither team has any published SLOs. You want to design a new joint-ownership model for a service between the DevOps team and the Software Development team. Which responsibilities should be assigned to each team in the new joint-ownership model?

A.

DevOps team responsibilities	Software Development team responsibilities
• Manage the service infrastructure • Be on-call for incidents • Perform code reviews	• Submit code to be reviewed by the DevOps team • Publish the SLOs that the DevOps team must meet

B.

DevOps team responsibilities	Software Development team responsibilities
• Manage the service infrastructure • Perform code reviews	• Submit code to be reviewed by the DevOps team • Be on-call for incidents • Publish the SLOs that the DevOps team must meet

C.

DevOps team responsibilities	Shared responsibilities	Software Development team responsibilities
• Manage the service infrastructure	• Perform code reviews • Be on-call for incidents on a rotation basis • Adopt and publish SLOs for the service	• Submit code to be reviewed

D.

DevOps team responsibilities	Shared responsibilities	Software Development team responsibilities
• Manage the service infrastructure • Be on-call for incidents	• Adopt and publish SLOs for the service	• Submit code to be reviewed • Perform code reviews

Options:

A- Option A

B- Option B

C- Option C

D- Option D

Answer:

D

Explanation:

The correct answer is D. Option D)

According to the DevOps best practices, a joint-ownership model for a service between the DevOps team and the Software Development team should follow these principles12:

The DevOps team and the Software Development team should share the responsibility and collaboration for managing the service infrastructure, performing code reviews, and adopting and sharing SLOs for the service.

The DevOps team and the Software Development team should have end-to-end ownership of the service, from design to development to deployment to operation to maintenance.

The DevOps team and the Software Development team should use common tools and processes to facilitate communication, coordination, and feedback.

The DevOps team and the Software Development team should align their goals and incentives with the business outcomes and customer satisfaction.

Option D is the only option that reflects these principles. Option D assigns both teams the responsibilities of managing the service infrastructure, performing code reviews, and adopting and sharing SLOs for the service. Option D also implies that both teams have end-to-end ownership of the service, as they are involved in every stage of the service lifecycle. Option D also encourages both teams to use common tools and processes, such as GitLab3, to collaborate and communicate effectively. Option D also aligns both teams with the business outcomes and customer satisfaction, as they use SLOs to measure and improve the service quality.

The other options are incorrect because they do not follow the DevOps best practices. Option A is incorrect because it assigns only the DevOps team the responsibility of managing the service infrastructure, which creates a silo between the two teams and reduces their collaboration. Option A also does not assign any responsibility for adopting and sharing SLOs for the service, which means that both teams lack a common metric for measuring and improving the service quality. Option B is incorrect because it assigns only the Software Development team the responsibility of performing code reviews, which creates a gap between the two teams and reduces their feedback. Option B also does not assign any responsibility for adopting and sharing SLOs for the service, which means that both teams lack a common metric for measuring and improving the service quality. Option C is incorrect because it assigns both teams the same responsibilities as option A and option B, which combines their drawbacks.

5 key organizational models for DevOps teams | GitLab, 5 key organizational models for DevOps teams | GitLab. Building a Culture of Full-Service Ownership - DevOps.com, Building a Culture of Full-Service Ownership - DevOps.com. GitLab, GitLab.

Question 2

Question Type: MultipleChoice

You are investigating issues in your production application that runs on Google Kubernetes Engine (GKE). You determined that the source Of the issue is a recently updated container image, although the exact change in code was not identified. The deployment is currently pointing to the latest tag. You need to update your cluster to run a version of the container that functions as intended. What should you do?

Options:

A- Create a new tag called stable that points to the previously working container, and change the deployment to point to the new tag.

B- Apply the latest tag to the previous container image, and do a rolling update on the deployment.

C- Build a new container from a previous Git tag, and do a rolling update on the deployment to the new container.

D- Alter the deployment to point to the sha2 56 digest of the previously working container.

Answer:

D

Question 3

Question Type: MultipleChoice

You have deployed a fleet Of Compute Engine instances in Google Cloud. You need to ensure that monitoring metrics and logs for the instances are visible in Cloud Logging and Cloud Monitoring by your company's operations and cyber security teams. You need to grant the required roles for the Compute Engine service account by using Identity and Access Management (IAM) while following the principle of least privilege. What should you do?

Options:

A- Grant the logging.editor and monitoring.metricwriter roles to the Compute Engine service accounts.

B- Grant the Logging. admin and monitoring . editor roles to the Compute Engine service accounts.

C- Grant the logging. logwriter and monitoring. editor roles to the Compute Engine service accounts.

D- Grant the logging. logWriter and monitoring. metricWriter roles to the Compute Engine service accounts.

Answer:

A

Explanation:

The correct answer is D. Grant the logging.logWriter and monitoring.metricWriter roles to the Compute Engine service accounts.

According to the Google Cloud documentation, the Compute Engine service account is a Google-managed service account that is automatically created when you enable the Compute Engine API1. This service account is used by default to run your Compute Engine instances and access other Google Cloud services on your behalf1. To ensure that monitoring metrics and logs for the instances are visible in Cloud Logging and Cloud Monitoring, you need to grant the following IAM roles to the Compute Engine service account23:

The logging.logWriter role allows the service account to write log entries to Cloud Logging4.

The monitoring.metricWriter role allows the service account to write custom metrics to Cloud Monitoring5.

These roles grant the minimum permissions that are needed for logging and monitoring, following the principle of least privilege. The other roles are either unnecessary or too broad for this purpose. For example, the logging.editor role grants permissions to create and update logs, log sinks, and log exclusions, which are not required for writing log entries6. The logging.admin role grants permissions to delete logs, log sinks, and log exclusions, which are not required for writing log entries and may pose a security risk if misused. The monitoring.editor role grants permissions to create and update alerting policies, uptime checks, notification channels, dashboards, and groups, which are not required for writing custom metrics.

Service accounts, Service accounts. Setting up Stackdriver Logging for Compute Engine, Setting up Stackdriver Logging for Compute Engine. Setting up Stackdriver Monitoring for Compute Engine, Setting up Stackdriver Monitoring for Compute Engine. Predefined roles, Predefined roles. Predefined roles, Predefined roles. Predefined roles, Predefined roles. [Predefined roles], Predefined roles. [Predefined roles], Predefined roles.

Question 4

Question Type: **MultipleChoice**

You are designing a new Google Cloud organization for a client. Your client is concerned with the risks associated with long-lived credentials created in Google Cloud. You need to design a solution to completely eliminate the risks associated with the use of JSON service account keys while minimizing operational overhead. What should you do?

Options:

A- Use custom versions of predefined roles to exclude all iam.serviceAccountKeys. * service account role permissions.

B- Apply the constraints/iam.disableserviceAccountKeycreation constraint to the organization.

C- Apply the constraints/iam. disableServiceAccountKeyUp10ad constraint to the organization.

D- Grant the roles/ iam.serviceAccountKeyAdmin IAM role to organization administrators only.

Answer:

B

Explanation:

The correct answer is B, Apply the constraints/iam.disableServiceAccountKeyCreation constraint to the organization.

According to the Google Cloud documentation, the constraints/iam.disableServiceAccountKeyCreation constraint is an organization policy constraint that prevents the creation of user-managed service account keys1. User-managed service account keys are long-lived credentials that can be downloaded as JSON or P12 files and used to authenticate as a service account2. These keys pose severe

security risks if they are leaked, stolen, or misused by unauthorized entities34. By applying this constraint to the organization, you can completely eliminate the risks associated with the use of JSON service account keys and enforce a more secure alternative for authentication, such as Workload Identity or short-lived access tokens12. This also minimizes operational overhead by avoiding the need to manage, rotate, or revoke user-managed service account keys.

The other options are incorrect because they do not completely eliminate the risks associated with the use of JSON service account keys. Option A is incorrect because it only restricts the IAM permissions to create, list, get, delete, or sign service account keys, but it does not prevent existing keys from being used or leaked. Option C is incorrect because it only disables the upload of user-managed service account keys, but it does not prevent the creation or download of such keys. Option D is incorrect because it only limits the IAM role that can create and manage service account keys, but it does not prevent the keys from being distributed or exposed to unauthorized entities.

Disable user-managed service account key creation, Disable user-managed service account key creation. Service accounts, User-managed service accounts. Help keep your Google Cloud service account keys safe, Help keep your Google Cloud service account keys safe. Stop Downloading Google Cloud Service Account Keys!, Stop Downloading Google Cloud Service Account Keys! [Service Account Keys], Service Account Keys. [Disable user-managed service account key upload], Disable user-managed service account key upload. [Granting roles to service accounts], Granting roles to service accounts.

Question 5

Question Type: MultipleChoice

You are monitoring a service that uses n2-standard-2 Compute Engine instances that serve large files. Users have reported that downloads are slow. Your Cloud Monitoring dashboard shows that your VMS are running at peak network throughput. You want to improve the network throughput performance. What should you do?

Options:

A- Deploy a Cloud NAT gateway and attach the gateway to the subnet of the VMS.

B- Add additional network interface controllers (NICs) to your VMS.

C- Change the machine type for your VMS to n2-standard-8.

D- Deploy the Ops Agent to export additional monitoring metrics.

Answer:

C

Explanation:

The correct answer is C, Change the machine type for your VMs to n2-standard-8.

According to the Google Cloud documentation, the network throughput performance of a Compute Engine VM depends on its machine type1. The n2-standard-2 machine type has a maximum egress bandwidth of 4 Gbps, which can be a bottleneck for serving large files. By changing the machine type to n2-standard-8, you can increase the maximum egress bandwidth to 16 Gbps, which can improve the network throughput performance and reduce the download time for users. You also need to enable per VM Tier_1 networking

performance, which is a feature that allows VMs to achieve higher network performance than the default settings2.

The other options are incorrect because they do not improve the network throughput performance of your VMs. Option A is incorrect because Cloud NAT is a service that allows private IP addresses to access the internet, but it does not increase the network bandwidth or speed3. Option B is incorrect because adding additional network interfaces (NICs) or IP addresses per NIC does not increase ingress or egress bandwidth for a VM1. Option D is incorrect because deploying the Ops Agent can help you monitor and troubleshoot your VMs, but it does not affect the network throughput performance4.

Cloud NAT overview, Cloud NAT overview. Network bandwidth, Bandwidth summary. Installing the Ops Agent, Installing the Ops Agent. Configure per VM Tier_1 networking performance, Configure per VM Tier_1 networking performance.

Question 6

Question Type: MultipleChoice

You recently noticed that one Of your services has exceeded the error budget for the current rolling window period. Your company's product team is about to launch a new feature. You want to follow Site Reliability Engineering (SRE) practices.

What should you do?

Options:

A- Notify the team that their error budget is used up. Negotiate with the team for a launch freeze or tolerate a slightly worse user experience.

B- Look through other metrics related to the product and find SLOs with remaining error budget. Reallocate the error budgets and allow the feature launch.

C- Escalate the situation and request additional error budget.

D- Notify the team about the lack of error budget and ensure that all their tests are successful so the launch will not further risk the error budget.

Answer:

A

Explanation:

The correct answer is

A, Notify the team that their error budget is used up. Negotiate with the team for a launch freeze or tolerate a slightly worse user experience.

According to the Site Reliability Engineering (SRE) practices, an error budget is the amount of unreliability that a service can tolerate without harming user satisfaction1. An error budget is derived from the service-level objectives (SLOs), which are the measurable goals for the service quality2. When a service exceeds its error budget, it means that it has violated its SLOs and may have negatively impacted the users. In this case, the SRE team should notify the product team that their error budget is used up and negotiate with them for a launch freeze or a lower SLO3. A launch freeze means that no new features are deployed until the service reliability is restored. A

lower SLO means that the product team accepts a slightly worse user experience in exchange for launching new features. Both options require a trade-off between reliability and innovation, and should be agreed upon by both teams.

The other options are incorrect because they do not follow the SRE practices. Option B is incorrect because it violates the principle of error budget autonomy, which means that each service should have its own error budget and SLOs, and should not borrow or reallocate them from other services4. Option C is incorrect because it does not address the root cause of the error budget overspend, and may create unrealistic expectations for the service reliability. Option D is incorrect because it does not prevent the possibility of introducing new errors or bugs with the feature launch, which may further degrade the service quality and user satisfaction.

Error Budgets, Error Budgets. Service Level Objectives, Service Level Objectives. Error Budget Policies, Error Budget Policies. Error Budget Autonomy, Error Budget Autonomy.

Question 7

Question Type: **MultipleChoice**

You are configuring your CI/CD pipeline natively on Google Cloud. You want builds in a pre-production Google Kubernetes Engine (GKE) environment to be automatically load-tested before being promoted to the production GKE environment. You need to ensure that only builds that have passed this test are deployed to production. You want to follow Google-recommended practices. How should you configure this pipeline with Binary Authorization?

Options:

A- Create an attestation for the builds that pass the load test by requiring the lead quality assurance engineer to sign the attestation by using a key stored in Cloud Key Management Service (Cloud KMS).

B- Create an attestation for the builds that pass the load test by using a private key stored in Cloud Key Management Service (Cloud KMS) authenticated through Workload Identity.

C- Create an attestation for the builds that pass the load test by using a private key stored in Cloud Key Management Service (Cloud KMS) with a service account JSON key stored as a Kubernetes Secret.

D- Create an attestation for the builds that pass the load test by requiring the lead quality assurance engineer to sign the attestation by using their personal private key.

Answer:

B

Explanation:

The correct answer is B, Create an attestation for the builds that pass the load test by using a private key stored in Cloud Key Management Service (Cloud KMS) authenticated through Workload Identity.

According to the Google Cloud documentation, Binary Authorization is a deploy-time security control that ensures only trusted container images are deployed on Google Kubernetes Engine (GKE) or Cloud Run1. Binary Authorization uses attestations to certify that a specific image has completed a previous stage in the CI/CD pipeline, such as passing a load test2. Attestations are signed by private keys that are associated with attestors, which are entities that verify the attestations3. To follow Google-recommended practices, you

should store your private keys in Cloud Key Management Service (Cloud KMS), which is a secure and scalable service for managing cryptographic keys4. You should also use Workload Identity, which is a feature that allows Kubernetes service accounts to act as Google service accounts, to authenticate to Cloud KMS and sign attestations without having to manage or expose service account keys5.

The other options are incorrect because they do not follow Google-recommended practices. Option A and option D require human intervention to sign the attestations, which is not scalable or automated. Option C exposes the service account JSON key as a Kubernetes Secret, which is less secure than using Workload Identity.

Creating an attestor, Creating an attestor. Cloud Key Management Service Documentation, Overview. Attestations overview, Attestations overview. Using Workload Identity with Binary Authorization, Using Workload Identity with Binary Authorization. Binary Authorization, Binary Authorization.

Question 8

Question Type: MultipleChoice

Your company runs applications in Google Kubernetes Engine (GKE). Several applications rely on ephemeral volumes. You noticed some applications were unstable due to the DiskPressure node condition on the worker nodes. You need to identify which Pods are causing the issue, but you do not have execute access to workloads and nodes. What should you do?

Options:

A- Check the node/ephemeral_storage/used_bytes metric by using Metrics Explorer.

B- Check the metric by using Metrics Explorer.

C- Locate all the Pods with emptyDir volumes. use the df-h command to measure volume disk usage.

D- Locate all the Pods with emptyDir volumes. Use the du -sh * command to measure volume disk usage.

Answer:

A

Explanation:

The correct answer is A, Check the node/ephemeral_storage/used_bytes metric by using Metrics Explorer.

The node/ephemeral_storage/used_bytes metric reports the total amount of ephemeral storage used by Pods on each node1. You can use Metrics Explorer to query and visualize this metric and filter it by node name, namespace, or Pod name2. This way, you can identify which Pods are consuming the most ephemeral storage and causing disk pressure on the nodes. You do not need to have execute access to the workloads or nodes to use Metrics Explorer.

The other options are incorrect because they require execute access to the workloads or nodes, which you do not have. The df -h and du -sh * commands are Linux commands that can measure disk usage, but you need to run them inside the Pods or on the nodes, which is not possible in your scenario34.

Monitoring metrics for Kubernetes system components, Node metrics, node/ephemeral_storage/used_bytes. Using Metrics Explorer, Querying metrics. How do I find out disk space utilization information using Linux command line?, df command. How to check disk space

in Linux from the command line, du command.

Question 9

Question Type: MultipleChoice

You need to create a Cloud Monitoring SLO for a service that will be published soon. You want to verify that requests to the service will be addressed in fewer than 300 ms at least 90% Of the time per calendar month. You need to identify the metric and evaluation method to use. What should you do?

Options:

A- Select a latency metric for a request-based method of evaluation.

B- Select a latency metric for a window-based method of evaluation.

C- Select an availability metric for a request-based method of evaluation.

D- Select an availability metric for a window-based method Of evaluation.

Answer:

A

Explanation:

The correct answer is

A, Select a latency metric for a request-based method of evaluation.

A latency metric measures how responsive your service is to users. For example, you can use the cloud.googleapis.com/http/server/response_latencies metric to measure the latency of HTTP requests to your service1. A request-based method of evaluation counts the number of successful requests that meet a certain criterion, such as being below a latency threshold, and compares it to the number of all requests. For example, you can define an SLI as the ratio of requests with latency below 300 ms to all requests2. A request-based method of evaluation is suitable for measuring performance over time, such as per calendar month. You can set an SLO for the SLI to be at least 90%, which means that you expect 90% of the requests to have latency below 300 ms in a month3.

Creating an SLO | Operations Suite | Google Cloud, Choosing a metric, Latency metric. Concepts in service monitoring | Operations Suite | Google Cloud, Service-level indicators, Request-based SLIs. Learn how to set SLOs -- SRE tips | Google Cloud Blog, Setting SLOs.

Question 10

Question Type: MultipleChoice

As a Site Reliability Engineer, you support an application written in GO that runs on Google Kubernetes Engine (GKE) in production. After releasing a new version Of the application, you notice the application runs for about 15 minutes and then restarts. You decide to add Cloud Profiler to your application and now notice that the heap usage grows constantly until the application restarts. What should you do?

Options:

A- Add high memory compute nodes to the cluster.

B- Increase the memory limit in the application deployment.

C- Add Cloud Trace to the application, and redeploy.

D- Increase the CPU limit in the application deployment.

Answer:

B

Explanation:

The correct answer is B, Increase the memory limit in the application deployment.

The application is experiencing a memory leak, which means that it is allocating memory that is not freed or reused. This causes the heap usage to grow constantly until it reaches the memory limit of the pod, which triggers a restart by Kubernetes. Increasing the memory limit in the application deployment can help mitigate the problem by allowing the application to run longer before reaching the

limit. However, this is not a permanent solution, as the memory leak will still occur and eventually exhaust the available memory. The best solution is to identify and fix the source of the memory leak in the application code, using tools like Cloud Profiler and pprof12.

Using Cloud Profiler with Go, Troubleshooting memory leaks. Profiling Go Programs, Heap profiles.

Question 11

Question Type: MultipleChoice

You are designing a deployment technique for your applications on Google Cloud. As part Of your deployment planning, you want to use live traffic to gather performance metrics for new versions Of your applications. You need to test against the full production load before your applications are launched. What should you do?

Options:

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

B- Use shadow testing with continuous deployment.

C- Use canary testing with continuous deployment.

D- Use canary testing with rolling updates deployment,

Answer:

B

Explanation:

The correct answer is B, Use shadow testing with continuous deployment.

Shadow testing is a deployment technique that involves routing a copy of the live traffic to a new version of the application, without affecting the production environment. This way, you can gather performance metrics and compare them with the current version, without exposing the new version to the users. Shadow testing can help you test against the full production load and identify any issues or bottlenecks before launching the new version. You can use continuous deployment to automate the process of deploying the new version after it passes the shadow testing.

Application deployment and testing strategies, Testing strategies, Shadow test pattern.