Google Professional-Cloud-Security-Engineer Mock Exam

Shared by Harrison on 17-06-2026

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

Your company plans to move most of its IT infrastructure to Google Cloud. They want to leverage their existing on-premises Active Directory as an identity provider for Google Cloud. Which two steps should you take to integrate the company's on-premises Active Directory with Google Cloud and configure access management? (Select two.)

Options:

A- Use Identity Platform to provision users and groups to Google Cloud.
B- Use Cloud Identity SAML integration to provision users and groups to Google Cloud.
C- Install Google Cloud Directory Sync and connect it to Active Directory and Cloud Identity.
D- Create Identity and Access Management (1AM) roles with permissions corresponding to each Active Directory group.
E- Create Identity and Access Management (1AM) groups with permissions corresponding to each Active Directory group.

Answer:

C, E

Explanation:

Google Cloud Directory Sync (GCDS): Install and configure GCDS to synchronize your on-premises Active Directory with Google Cloud Identity. This tool helps in maintaining consistency between your local directory and Google Cloud.

IAM Groups: Create IAM groups in Google Cloud with permissions that correspond to your Active Directory groups. This mapping ensures that users inherit the appropriate permissions based on their AD group membership.

Synchronization: Set up regular synchronization schedules to keep the user and group information up-to-date between your on-premises AD and Google Cloud.

Access Management: Use these IAM groups to manage access to Google Cloud resources, ensuring that permissions are applied consistently and securely. This approach leverages existing AD infrastructure for identity management, providing a seamless integration with Google Cloud.
Reference::

Google Cloud - Google Cloud Directory Sync

Google Cloud - IAM Groups

Question 2

Question Type: MultipleChoice

Applications often require access to ''secrets'' - small pieces of sensitive data at build or run time. The administrator managing these secrets on GCP wants to keep a track of ''who did what, where, and when?'' within their GCP projects.

Which two log streams would provide the information that the administrator is looking for? (Choose two.)

Options:

A- Admin Activity logs
B- System Event logs
C- Data Access logs
D- VPC Flow logs
E- Agent logs

Answer:

A, C

Explanation:

To keep track of 'who did what, where, and when?' within GCP projects, the administrator should focus on Admin Activity logs and Data Access logs. Here's a detailed explanation of why these two log streams are essential:

Admin Activity Logs:

These logs capture administrative actions performed in your Google Cloud resources. This includes actions like creating, modifying, or deleting resources.

Admin Activity logs provide detailed information about the user who performed the action, the resource that was affected, the action performed, and the timestamp.

Data Access Logs:

These logs capture read and write operations on data within your Google Cloud services. This includes actions like accessing or modifying data stored in databases, storage buckets, etc.

Data Access logs help track the access patterns of users and services to sensitive data, providing insights into who accessed which data and when.

Steps to Enable and Access Logs:

Navigate to the Google Cloud Console.

Go to Logging in the left-hand menu.

Enable Admin Activity and Data Access logs if not already enabled.

Use Logs Explorer to filter and view specific logs based on your requirements.

By monitoring both Admin Activity and Data Access logs, administrators can gain comprehensive visibility into the actions performed on their GCP resources and data, ensuring robust security and compliance tracking.

Google Cloud Logging Documentation

Audit Logs Overview

Question 3

Question Type: MultipleChoice

Your organization has on-premises hosts that need to access Google Cloud APIs You must enforce private connectivity between these hosts minimize costs and optimize for operational efficiency

What should you do?

Options:

A- Route all on-premises traffic to Google Cloud through an IPsec VPN tunnel to a VPC with Private Google Access enabled.
B- Set up VPC peering between the hosts on-premises and the VPC through the internet.
C- Enforce a security policy that mandates all applications to encrypt data with a Cloud Key Management. Service (KMS) key before you send it over the network.
D- Route all on-premises traffic to Google Cloud through a dedicated or Partner interconnect to a VPC with Private Google Access enabled.

Answer:

D

Explanation:

To enforce private connectivity between on-premises hosts and Google Cloud APIs while optimizing for cost and operational efficiency, using a dedicated or Partner Interconnect is the best solution. This setup ensures a reliable, high-bandwidth connection with private IP addressing.

Choose Interconnect Type: Decide between Dedicated Interconnect and Partner Interconnect based on your bandwidth needs and proximity to Google Cloud locations.

Set Up Interconnect:

For Dedicated Interconnect, order circuits through the Google Cloud Console.

For Partner Interconnect, select a supported service provider and order the connection through them.

Configure VPC and Private Google Access:

In your VPC, enable Private Google Access to allow on-premises hosts to access Google APIs privately.

Go to 'VPC network' -> 'Private Google Access' and enable it for your subnets.

Establish Connectivity: Work with your network team and (if applicable) your Partner Interconnect provider to set up the physical and logical connections.

Test Connectivity: Verify that on-premises hosts can reach Google Cloud services using private IP addresses.

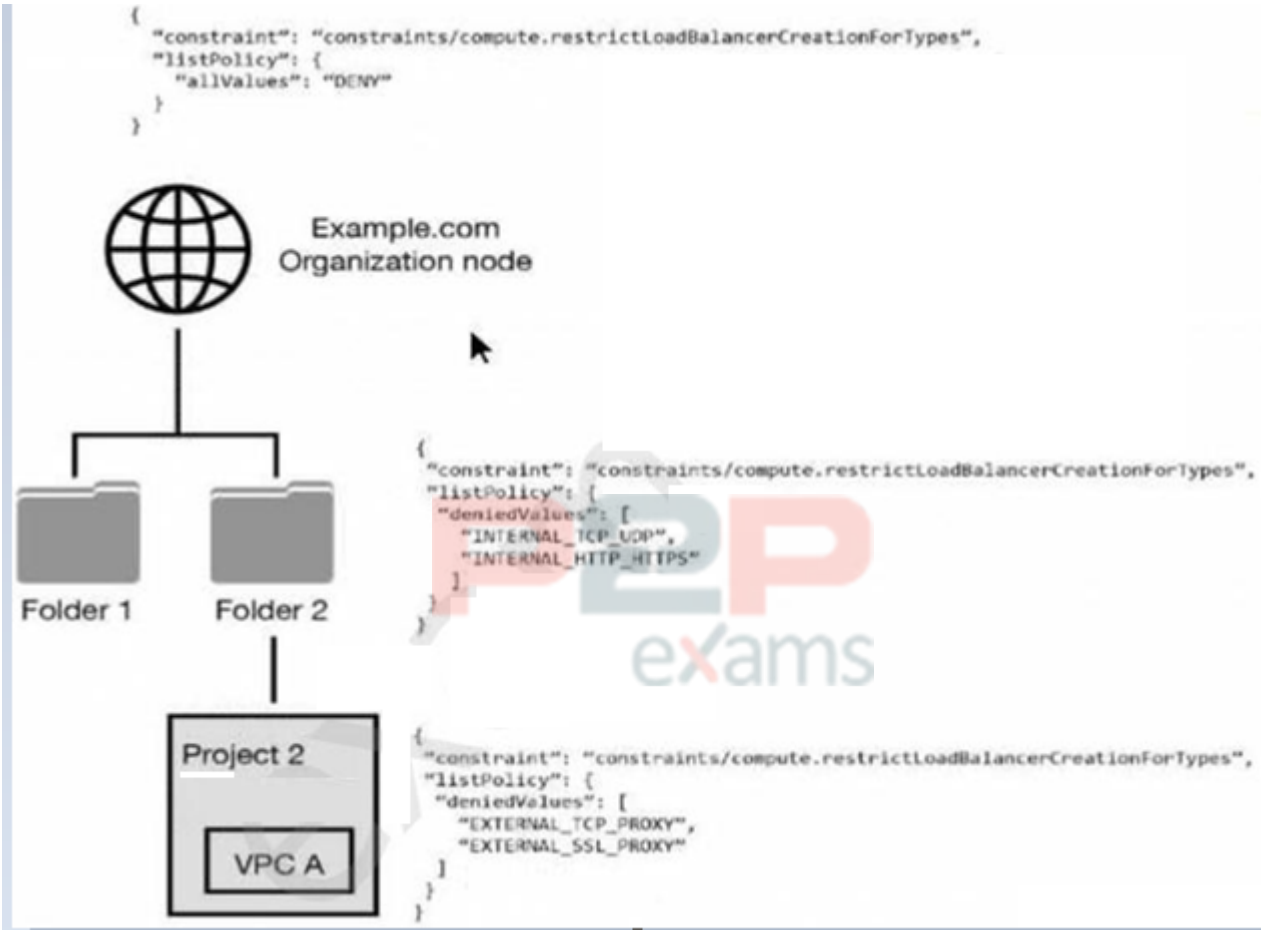
Google Cloud Interconnect Overview

Configuring Private Google Access

Question 4

Question Type: MultipleChoice

You have the following resource hierarchy. There is an organization policy at each node in the hierarchy as shown. Which load balancer types are denied in VPC A?



Options:

A- All load balancer types are denied in accordance with the global node's policy.
B- INTERNAL_TCP_UDP, INTERNAL_HTTP_HTTPS is denied in accordance with the folder's policy.
C- EXTERNAL_TCP_PROXY, EXTERNAL_SSL_PROXY are denied in accordance with the project's policy.
D- EXTERNAL_TCP_PROXY, EXTERNAL_SSL_PROXY, INTERNAL_TCP_UDP, and INTERNAL_HTTP_HTTPS are denied in accordance with the folder and project's policies.

Answer:

D

Explanation:

Understanding Organization Policies:

Organization policies are rules that can be set at different levels of the resource hierarchy in GCP to enforce governance and compliance.

These policies can be set at the organization node, folders, and projects, and they are inherited down the hierarchy unless explicitly overridden.

Hierarchy and Policy Inheritance:

The provided resource hierarchy has an organization node (Example.com), folders (Folder 1 and Folder 2), and a project (Project 2) under Folder 2 with a specific VPC (VPC A).

Each node in the hierarchy can have its own policies, and these policies are inherited by child nodes unless overridden.

Analyzing the Policies in the Hierarchy:

Organization Node Policy:

json

Copy code

{ 'constraint': 'constraints/compute.restrictLoadBalancerCreationForTypes', 'listPolicy': { 'allValues': 'DENY' } }

This policy at the organization node denies all load balancer types.

Folder 2 Policy:

json

Copy code

{ 'constraint': 'constraints/compute.restrictLoadBalancerCreationForTypes', 'listPolicy': { 'deniedValues': ['INTERNAL_TCP_UDP', 'INTERNAL_HTTP_HTTPS'] } }

This policy at Folder 2 denies the creation of INTERNAL_TCP_UDP and INTERNAL_HTTP_HTTPS load balancers.

Project 2 Policy:

json

Copy code

{ 'constraint': 'constraints/compute.restrictLoadBalancerCreationForTypes', 'listPolicy': { 'deniedValues': ['EXTERNAL_TCP_PROXY', 'EXTERNAL_SSL_PROXY'] } }

This policy at Project 2 denies the creation of EXTERNAL_TCP_PROXY and EXTERNAL_SSL_PROXY load balancers.

Policy Application to VPC A:

Since policies are inherited, VPC A (which is within Project 2 under Folder 2) will be affected by the policies of both Folder 2 and Project 2.

Combining the denied values from both Folder 2 and Project 2:

From Folder 2: INTERNAL_TCP_UDP, INTERNAL_HTTP_HTTPS

From Project 2: EXTERNAL_TCP_PROXY, EXTERNAL_SSL_PROXY

Conclusion:

VPC A will have the following load balancer types denied: INTERNAL_TCP_UDP, INTERNAL_HTTP_HTTPS, EXTERNAL_TCP_PROXY, EXTERNAL_SSL_PROXY.

GCP Documentation on Organization Policies

GCP Documentation on Constraints and List Policies

Question 5

Question Type: MultipleChoice

In an effort for your company messaging app to comply with FIPS 140-2, a decision was made to use GCP compute and network services. The messaging app architecture includes a Managed Instance Group (MIG) that controls a cluster of Compute Engine instances. The instances use Local SSDs for data caching and UDP for instance-to-instance communications. The app development team is willing to make any changes necessary to comply with the standard

Which options should you recommend to meet the requirements?

Options:

A- Encrypt all cache storage and VM-to-VM communication using the BoringCrypto module.
B- Set Disk Encryption on the Instance Template used by the MIG to customer-managed key and use BoringSSL for all data transit between instances.
C- Change the app instance-to-instance communications from UDP to TCP and enable BoringSSL on clients' TLS connections.
D- Set Disk Encryption on the Instance Template used by the MIG to Google-managed Key and use BoringSSL library on all instance-to-instance communications.

Answer:

B

Explanation:

To comply with FIPS 140-2 for the messaging app, you need to ensure that both data at rest and data in transit are encrypted according to the standard. Using customer-managed encryption

keys (CMEK) ensures that you have control over the encryption keys, and BoringSSL is a library that meets FIPS 140-2 standards for encrypting data in transit.

Steps:

Encrypt Local SSDs: Modify the instance template for the Managed Instance Group (MIG) to use customer-managed encryption keys (CMEK) for encrypting Local SSDs.

Enable BoringSSL: Update the application to use the BoringSSL library for all instance-to-instance communication to ensure that all data in transit is encrypted according to FIPS 140-2 standards.

Google Cloud: Customer-managed encryption keys (CMEK)

BoringSSL documentation

Question 6

Question Type: MultipleChoice

You are the security admin of your company. Your development team creates multiple GCP projects under the "implementation" folder for several dev, staging, and production workloads. You want to prevent data exfiltration by malicious insiders or compromised code by setting up a security perimeter. However, you do not want to restrict communication between the projects.

What should you do?

Options:

A- Use a Shared VPC to enable communication between all projects, and use firewall rules to prevent data exfiltration.
B- Create access levels in Access Context Manager to prevent data exfiltration, and use a shared VPC for communication between projects.
C- Use an infrastructure-as-code software tool to set up a single service perimeter and to deploy a Cloud Function that monitors the 'implementation' folder via Stackdriver and Cloud Pub/Sub. When the function notices that a new project is added to the folder, it executes Terraform to add the new project to the associated perimeter.
D- Use an infrastructure-as-code software tool to set up three different service perimeters for dev, staging, and prod and to deploy a Cloud Function that monitors the 'implementation' folder via Stackdriver and Cloud Pub/Sub. When the function notices that a new project is added to the folder, it executes Terraform to add the new project to the respective perimeter.

Answer:

D

Explanation:

Setting up separate service perimeters for dev, staging, and prod environments allows for more granular control and monitoring. Automating the addition of new projects to the respective perimeters ensures that all projects are consistently secured without manual intervention.

Steps:

Set Up Service Perimeters: Use Access Context Manager to define and configure three separate service perimeters for dev, staging, and prod.

Deploy Monitoring Function: Create a Cloud Function that monitors the 'implementation' folder for new projects using Stackdriver (Cloud Monitoring) and Cloud Pub/Sub.

Automate Perimeter Updates: Configure the Cloud Function to execute Terraform scripts that automatically add new projects to the appropriate service perimeter.

Google Cloud: Access Context Manager

Service perimeter automation

Question 7

Question Type: MultipleChoice

You are a Security Administrator at your organization. You need to restrict service account creation capability within production environments. You want to accomplish this centrally across the organization. What should you do?

Options:

A- Use Identity and Access Management (IAM) to restrict access of all users and service accounts that have access to the production environment.
B- Use organization policy constraints/iam.disableServiceAccountKeyCreation boolean to disable the creation of new service accounts.
C- Use organization policy constraints/iam.disableServiceAccountKeyUpload boolean to disable the creation of new service accounts.
D- Use organization policy constraints/iam.disableServiceAccountCreation boolean to disable the creation of new service accounts.

Answer:

D

Explanation:

You can use the iam.disableServiceAccountCreation boolean constraint to disable the creation of new service accounts. This allows you to centralize management of service accounts while not restricting the other permissions your developers have on projects.
https://cloud.google.com/resource-manager/docs/organization-policy/restricting-service-accounts#disable_service_account_creation

Question 8

Question Type: MultipleChoice

An engineering team is launching a web application that will be public on the internet. The web application is hosted in multiple GCP regions and will be directed to the respective backend based on the URL request.

Your team wants to avoid exposing the application directly on the internet and wants to deny traffic from a specific list of malicious IP addresses

Which solution should your team implement to meet these requirements?

Options:

A- Cloud Armor
B- Network Load Balancing
C- SSL Proxy Load Balancing
D- NAT Gateway

Answer:

A

Explanation:

Google Cloud Armor provides protection against DDoS attacks and allows you to define security policies to control access to your application. It enables you to block traffic from specific IP addresses or ranges, making it suitable for denying traffic from a list of malicious IP addresses

while protecting your application from being directly exposed to the internet.

Steps:

Set Up Cloud Armor: Enable Cloud Armor in your Google Cloud Console.

Create Security Policies: Define security policies that specify the rules for allowing or denying traffic based on IP addresses.

Attach Policies to Backend Services: Apply these security policies to the backend services of your web application.

Google Cloud Armor documentation

Creating and managing security policies

P2P exams