
Optimizing and
Operating

Course agenda

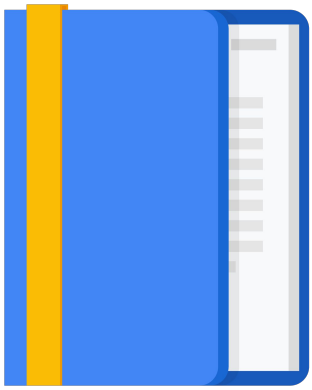
Module 1: Understanding the Professional Cloud Architect Certification

Module 2: Sample Case Studies for the Professional Cloud Architect Exam

Module 3: Designing and Implementing (Review and Preparation Tips)

Module 4: Optimizing and Operating (Review and Preparation Tips)

Module 5: Resources and Next Steps



Google Cloud

Module agenda

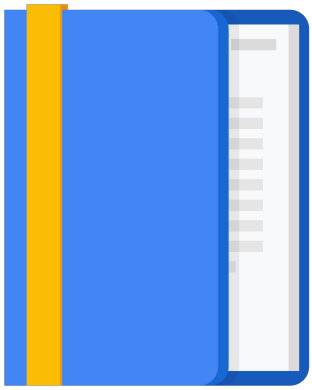
Designing for Security and Compliance

Analyzing and Optimizing Technical and Business Processes

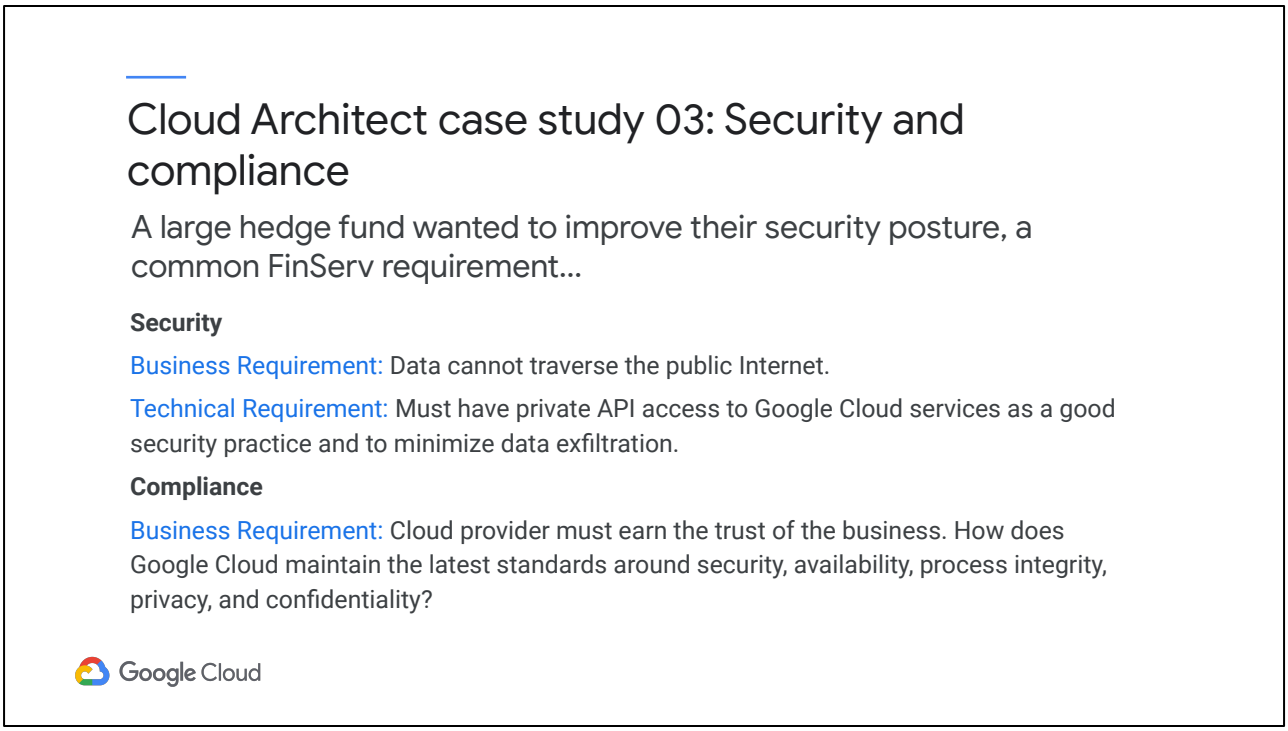
Developing Procedures to Test Resilience of Solution in Production

Managing Implementation

Ensuring Solution and Operations Reliability



Google Cloud



This customer had a common FinServ requirement. The customer did not want any data to traverse the public internet, for obvious reasons. So they had a security strategy that included a technical requirement to use private APIs to access Google Cloud resources. They saw this as fundamental to their security strategy. Additionally, they wanted to know how the Cloud Provider secured Standards Certifications, and what they did to stay current. They were concerned that the provider might lose a certification that they were relying on for business.

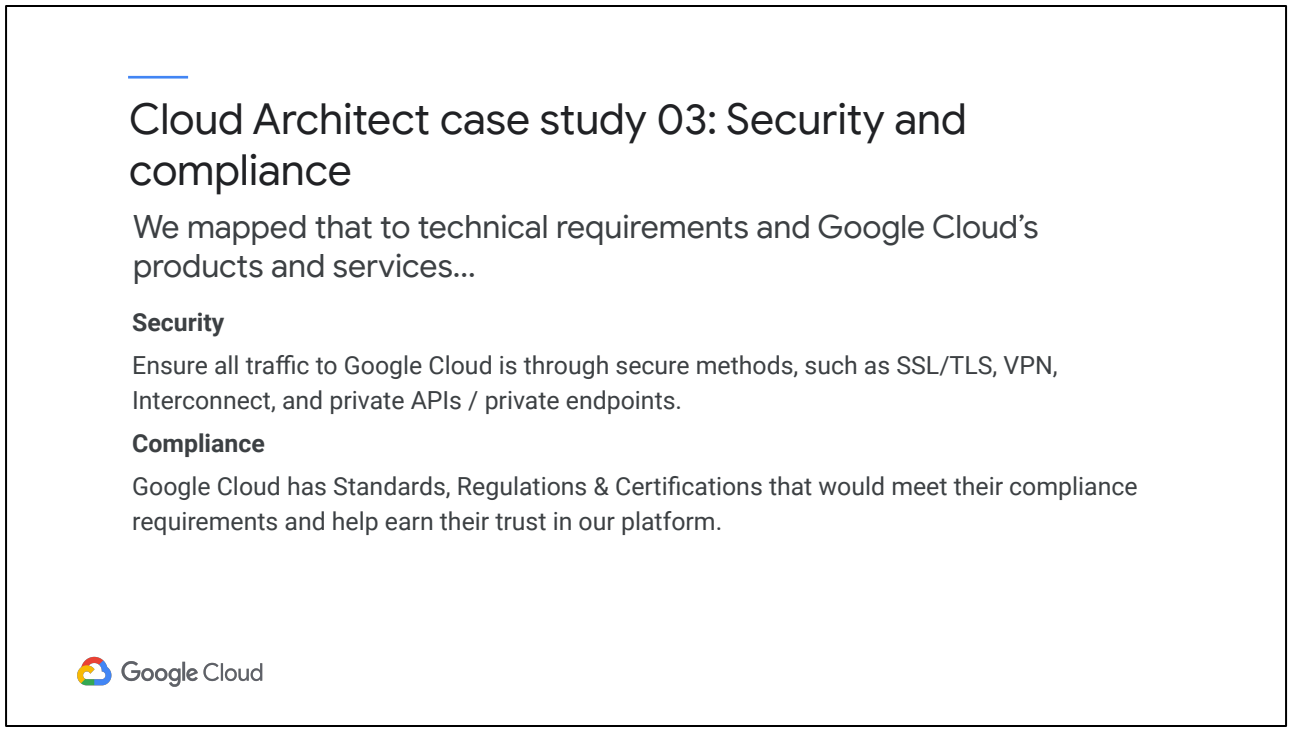
A large financial company wanted to improve their security posture, a common FinServ requirement...

Security

- Business Requirement: Data cannot traverse the public Internet.
- Technical Requirement: Must have private API access to Google Cloud services as a good security practice and to minimize data exfiltration.

Compliance

- Business Requirement: Cloud provider must earn the trust of the business. How does Google Cloud maintain the latest standards around security, availability, process integrity, privacy, and confidentiality?



The first thing we did was made sure all access to Google Cloud was through secure methods, including SSL, VPN, Interconnect, and private API.

We decided to use a new feature that was in alpha, called VPC Service control. https://cloud.google.com/vpc-service-controls/ This enables a security perimeter. For example, BigQuery could be placed inside a security perimeter, and then could only be accessed at a private endpoint. And then there were standards and compliance such as ISO and SOC. We provided these to the customer - and they needed to sign agreements to be covered by Google's guarantees about these standards.

We mapped that to technical requirements and Google Cloud's products and services…

Security

- Ensure all traffic to Google Cloud is through secure methods, such as SSL/TLS, VPN, Interconnect, and private APIs / private endpoints.

Compliance

- Google Cloud has Standards, Regulations & Certifications that would meet their compliance requirements and help earn their trust in our platform.

Cloud Architect case study 03: Security and compliance

And this is how we implemented that technical requirement.

VPC Service Controls / Secure Google Cloud API

- Restrict access to user's Google Cloud resources based on the Google Cloud Virtual Network or IP range.
- Restrict the set of Google APIs and Google Cloud resources accessible from user's Google Cloud Virtual Network.

Standards, Regulations & Certifications

- Products regularly undergo independent verification of:
  - Security / Privacy / Compliance Controls
  - Certifications
    - ISO 27001, 27017, and 27018 and SOC 1, 2, and 3 certifications.

Google Cloud

Currently in Alpha, VPC Service Controls is likely to be on future versions of the Cloud Architect certification exam.

All services are managed through a secured global API gateway infrastructure.

An interesting point about both security and compliance, is that it is a "shared responsibility" model. So although we provided secure access and layered protection, the customer needed to use IAM to manage access to its employees and implement secure practices in its procedures. Also, the standards compliance covers the cloud resources, but not the customer's application. So they may need to take extra steps to ensure that the overall solution is compliant.

Designing for security

Exam outline	Tips
Identity and Access Management (IAM)	Use groups.
Data security (key management, encryption)	Principle of least privilege.
Resource hierarchy (organizations, folders, projects)	Separation of responsibilities to match organization structure.
Penetration testing	Shape the scope and the objectives of the test.
Separation of duties	Isolate key roles. Always have an alternate/secondary who can take over if needed. Service accounts.
Security controls	How are you monitoring security? What logs and reports are available to you? What responses will you take?
Managing customer-supplied encryption keys with Cloud KMS	Key management, key rotation, standards and policy compliance.

TIP

Can you apply the "principle of least privilege" in case examples?

Google Cloud

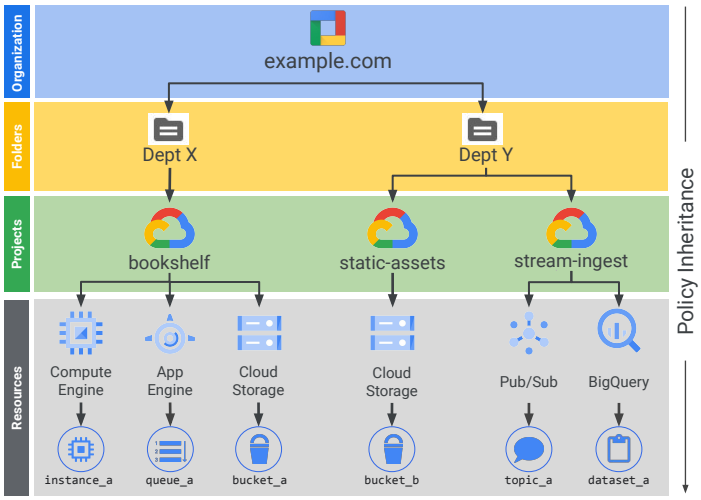
One key to securing access is to request and establish groups that represent roles. Then apply the permissions to the groups. And allow the people in the organization who manage identity to assign membership to the groups. This creates a clean interface between permission management on the cloud side, and group membership on the personnel/IT side.

Another key to security is to craft security permissions. The standard roles are defined for the most common use cases. But you might want to derive more granular and restricted roles by customizing them.

Service accounts are a great way to separate system components and establish secure communications between components.

A bastion host is a way to leverage a service account. For risky and uncommon actions, make the user/admin start up and log into a bastion host. From there they can "borrow" the service account assigned to the host to perform restricted functions. One benefit is that the login process generates logs for accountability.

Cloud IAM resource hierarchy



Google Cloud

A policy is set on a resource, and each policy contains a set of:

- Roles
- Role members
- Resources inherit policies from parent:
- Resource policies are a union of parent and resource.
- If parent policy is less restrictive, it overrides a more restrictive resource policy.

https://cloud.google.com/iam/

https://cloud.google.com/iam/docs/

https://cloud.google.com/iam/docs/concepts

https://cloud.google.com/iam/docs/understanding-roles

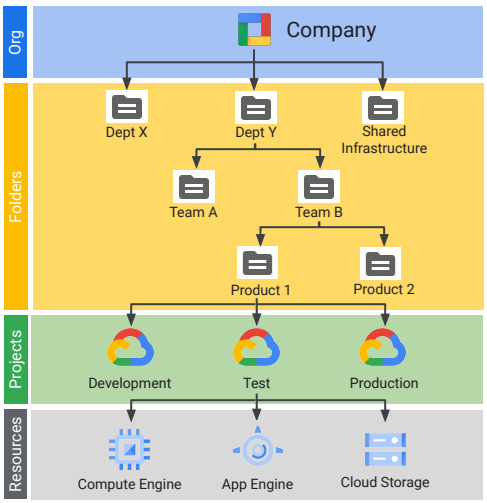
https://cloud.google.com/iam/docs/service-accounts

Folders

Additional grouping mechanism and isolation boundaries between projects:

- Different legal entities
- Departments
- Teams

Folders allow delegation of administration rights.



Google Cloud

Folder map well to organization structure. It is a way to isolate organizations or users or products while still having them share billing and corporate resources.

https://cloud.google.com/resource-manager/docs/managing-multiple-orgs

https://cloud.google.com/resource-manager/docs/access-control-org

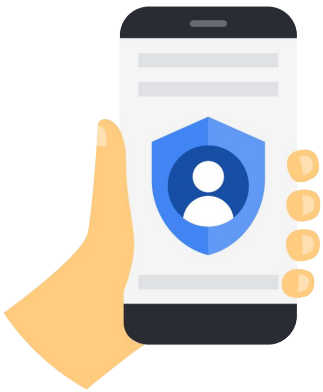
https://cloud.google.com/resource-manager/docs/creating-managing-projects

https://cloud.google.com/resource-manager/docs/access-control-proj

https://cloud.google.com/resource-manager/docs/creating-managing-folders

https://cloud.google.com/resource-manager/docs/access-control-folders

Identity and access



- Separate responsibilities.
- Always have a backup or alternative in case the responsible person is unreachable.
- Have a separate maintenance path when the normal paths aren't working (e.g., bastion host).
- Use groups to allocate permissions, then separately manage group membership.
- Customize roles for greater granularity of permissions.
- Give each group only the permissions they need to perform that job or task.
- Place critical functions on service machines to create accountability trail (login log, activity monitoring).
- Backup/spare logs and records; have a review, analysis, and monitoring strategy (ex: monthly reports).

Google Cloud

TIP: Commit a security checklist to memory. Sometimes just running down a list will rapidly identify a solution.

Why do you bother locking doors?

Google Cloud

The principal illustrated in this image is very important and will help you understand the next several slides about security.

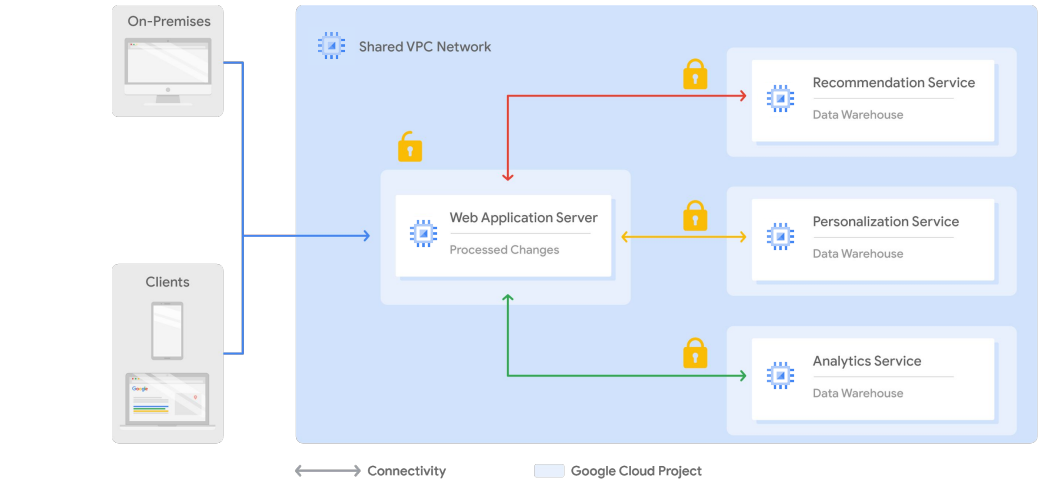
This is a very simple door lock. But it is on the **inside** of this door. The person or people inside can unlock this very easily at any time. For them it does nothing but puts a step in the way of using the door. But the lock is not supposed to do anything *for* them. The purpose of the lock is to keep *other people out*. So to do that, you need to lock your own people *in*.

This principal is on display in the way networking contributes to security.

Consider an internal firewall between two VMs. Why would you want to do that?

If you think about it functionally it makes no sense. Because the VMs can already communicate. So adding a firewall between them does nothing for them. It only makes it inconvenient if they want to communicate using a different protocol. Now you have to go change the firewall rule to allow that protocol. And it seems like invented work. Because if you just didn't put the firewall rule in place to begin with you wouldn't have to modify it later. However, the firewall rule is not for those VMs. It doesn't do anything for them. It is to keep others out of their communications. To prevent someone from spoofing or injecting bad traffic as part of an attack to either violate privacy or deny service.

Shared VPC: Keep others out by locking you in



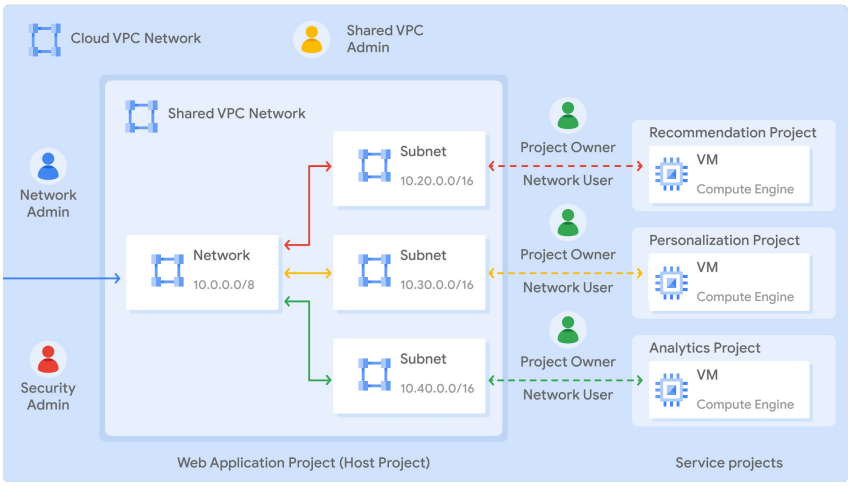
Google Cloud

Shared VPC allows an organization to connect resources from multiple projects to a common VPC network. This allows the resources to communicate with each other securely and efficiently using internal IPs from that network.

For example, in this diagram there is one network that belongs to the Web Application Server's project. This network is shared with three other projects, namely the Recommendation Service, the Personalization Service, and the Analytics Service. Each of those service projects has instances that are in the same network as the Web Application Server, allowing for private communication to that server, using internal IP addresses. The Web Application Server communicates with clients and on-premises using the server's external IP address. The backend services, on the other hand, cannot be reached externally because they only communicate using internal IP addresses.

When you use shared VPC, you designate a project as a host project and attach one or more other service projects to it. In this case, the Web Application Server's project is the host project, and the three other projects are the service projects. The overall VPC network is called the shared VPC network.

An example of Shared VPC keeping a system safe



Google Cloud

This example has one host project and 3 service projects:

In this diagram, the Shared VPC Admin, which was nominated by an organization admin, configured the Web Application Project to be a host project with subnet-level permissions. Doing so allowed the Shared VPC Admin to selectively share subnets from the VPC network.

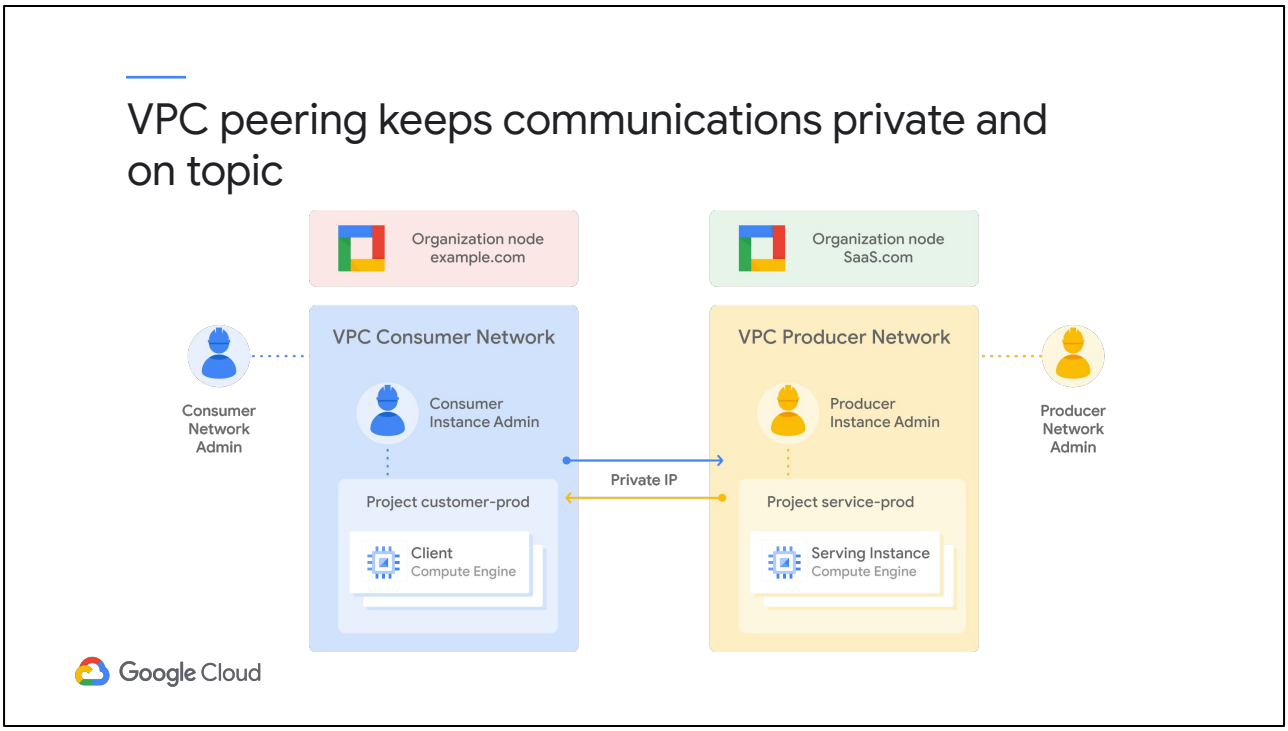
Next, the Shared VPC Admin attached the three service projects to the host project and gave each project owner the Network User role for the corresponding subnets. Each project owner then created VM instances from their service projects in the shared subnets.

The billing for those VM instances is attributed to the project where the resources are created, which are the service projects.

Shared VPC Admins have full control over the resources in the host project, including administration of the shared VPC network. They can optionally delegate the Network Admin and Security Admin roles for the host project. Overall, shared VPC is a centralized approach to multi-project networking because security and network policy occurs in a single designated VPC network.

For a demo on how to create VM instances in a Shared VPC network, please refer here:
https://storage.googleapis.com/cloud-training/gcpnet/student/M3%20-%20Shared%20

VPC.mp4



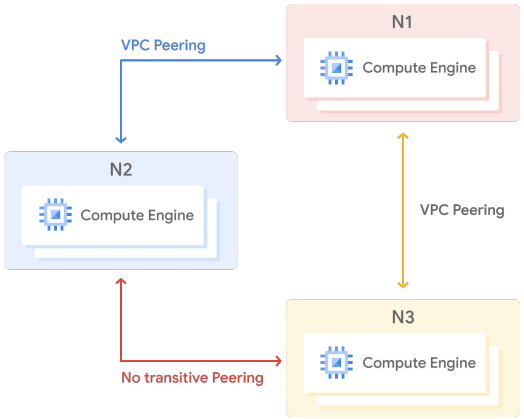
VPC Network Peering allows private RFC 1918 connectivity across two VPC networks, regardless of whether they belong to the same project or the same organization. Now, remember that each VPC network will have firewall rules that define what traffic is allowed or denied between the networks.

For example, in this diagram there are two organizations that represent a consumer and a producer, respectively. Each organization has its own organization node, VPC network, VM instances, Network Admin and Instance Admin. In order for VPC Network Peering to be established successfully, the Producer Network Admin needs to peer the Producer Network with the Consumer Network, and the Consumer Network Admin needs to peer the Consumer Network with the Producer Network. When both peering connections are created, the VPC Network Peering session becomes Active and routes are exchanged This allows the VM instances to communicate privately, using their internal IP addresses.

VPC Network Peering is a decentralized or distributed approach to multi-project networking, because each VPC network may remain under the control of separate administrator groups and maintains its own global firewall and routing tables. Historically, such projects would consider external IP addresses or VPNs to facilitate private communication between VPC networks. However, VPC Network Peering does not incur the network latency, security, and cost drawbacks that are present when using external IP addresses or VPNs.

Tips on designing solutions that use VPC peering

- Compute Engine, Google Kubernetes Engine, and App Engine flexible environments
- Peered VPC networks remain administratively separate
- Each side of a peering association is set up independently
- No subnet IP range overlap across peered VPC networks
- Transitive peering is not supported

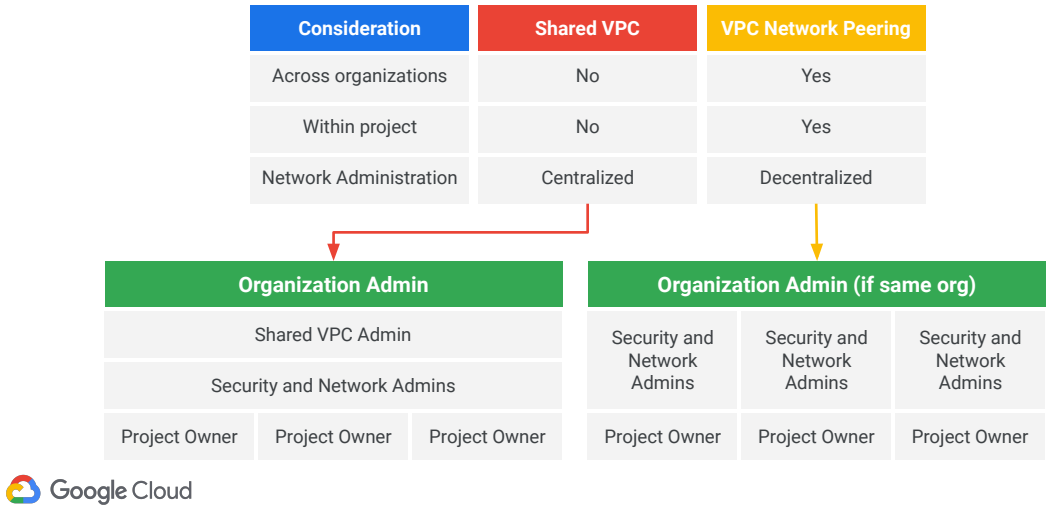


Google Cloud

Now, there are some things that we want you to remember when using VPC Network Peering:

- First of all, VPC Network Peering works with Compute Engine, Google Kubernetes Engine, and App Engine flexible environments.
- Peered VPC networks remain administratively separate. This means that routes, firewalls, VPNs, and other traffic management tools are administered and applied separately in each of the VPC networks.
- Each side of a peering association is set up independently. Peering will be active only when the configuration from both sides matches. This allows either side to delete the peering association at any time.
- A subnet CIDR prefix in one peered VPC network cannot overlap with a subnet CIDR prefix in another peered network. This means that two auto mode VPC networks that only have the default subnets cannot peer.
- Only directly peered networks can communicate, meaning that transitive peering is not supported. In other words, if VPC network N1 is peered with N2 and N3, but N2 and N3 are not directly connected, VPC network N2 cannot communicate with VPC network N3 over the peering. This is critical if N1 is a SaaS organization offering services to N2 and N3.

Should you use Shared VPC or VPC peering ?



Google Cloud

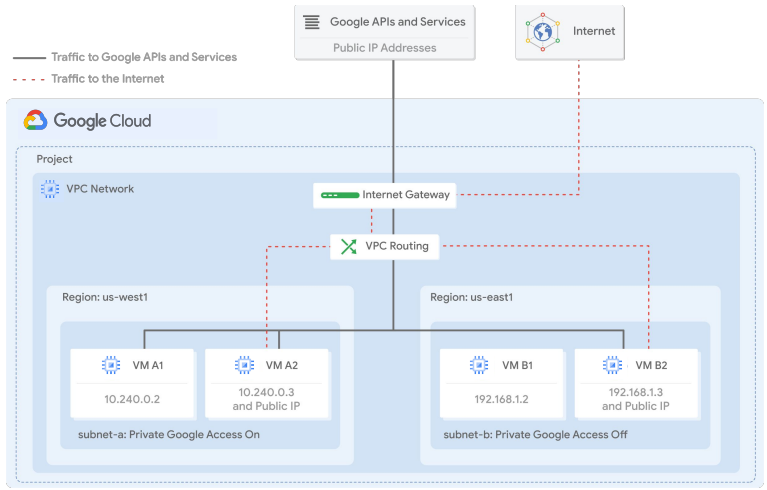
Should you use Shared VPC or VPC peering for a particular situation? That depends on the business administration requirements.

The biggest difference between the two configurations is the network administration models. Shared VPC is a centralized approach to multi-project networking, because security and network policy occurs in a single designated VPC network. In contrast, VPC Network Peering is a decentralized approach, because each VPC network can remain under the control of separate administrator groups and maintains its own global firewall and routing tables.

Also, consider the limits of VM instances per network and total among peered networks: https://cloud.google.com/vpc/docs/quota#per_network

Now that we've compared both of these configurations for sharing VPC networks across Google Cloud projects, let's look at one last use case.

Remove external IPs using Private Google Access



Google Cloud

The goal of Private Google Access is to remove external IPs from your VMs. Every time you remove an external IP you reduce the number of opportunities for an attacker to gain entry or try to deny service.

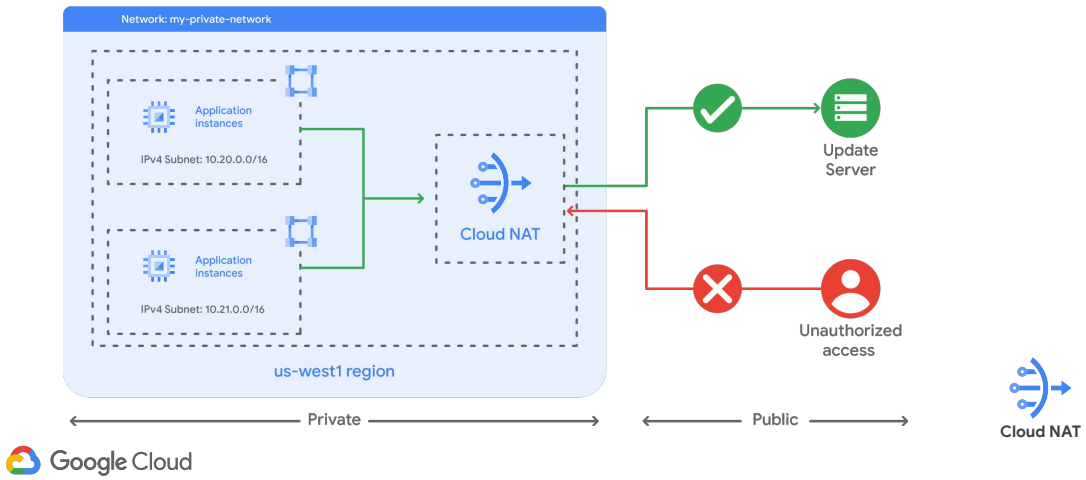
Private Google Access allows VM instances that only have internal IP addresses to reach the external IP addresses of Google APIs and services. For example, if your private VM instance needs to access a Cloud Storage bucket, you need to enable Private Google Access.

You enable Private Google Access on a subnet-by-subnet basis. As you can see in this diagram, subnet-a has Private Google Access enabled, and subnet-b has it disabled. This allows VM A1 to access Google APIs and services, even though it has no external IP address.

Private Google Access has no effect on instances that have external IP addresses. That's why VMs A2 and B2 can access Google APIs and services. The only VM that can't access those APIs and services is VM B1. This VM has no public IP address, and it is in a subnet where Google Private Access is disabled.

For a list of the services supported by Private Google Access, see: https://cloud.google.com/vpc/docs/private-access-options#pga-supported

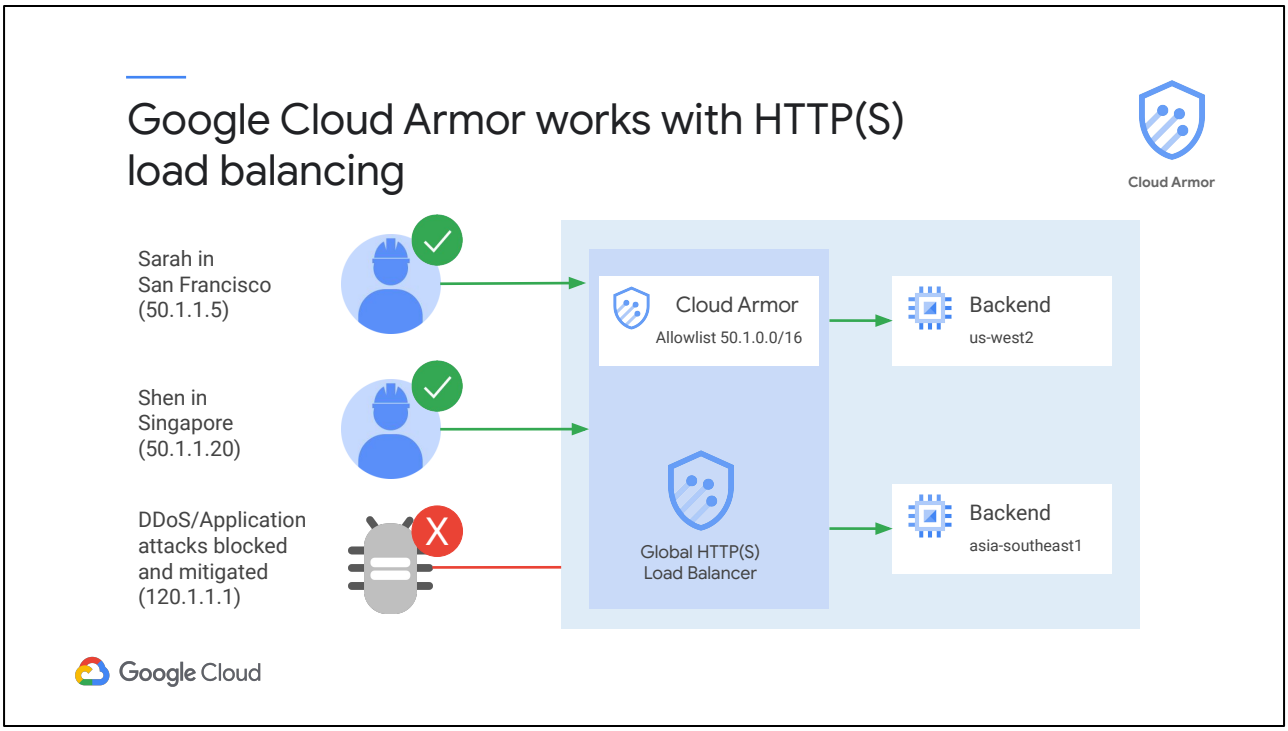
Cloud NAT provides internet access to private instances



Network Address Translation enables you to provide limited internet access without adding an external IP to your VM, and without exposing your internal IP to the world.

Cloud NAT is Google's managed network address translation service. It lets you provision your application instances without public IP addresses, while also allowing them to access the internet in a controlled and efficient manner. This means that your private instances can access the internet for updates, patching, configuration management, and more.

In this diagram, Cloud NAT enables two private instances to access an update server on the internet, which is referred to as outbound NAT. However, Cloud NAT does not implement inbound NAT. In other words, hosts outside your VPC network cannot directly access any of the private instances behind the Cloud NAT gateway. This helps you keep your VPC networks isolated and secure.



Physical hardware load balancers are a target for denial of service attacks. That is because they have physical hardware network interfaces that have a high but limited capacity. If an attacker can overrun an interface with traffic, or trigger an overrun from the internal service out to the internet, they can cause congestion that will slow or halt traffic.

Google Cloud Armor works with global HTTP(S) load balancing to provide built-in defenses against Infrastructure Distributed Denial of Service or DDoS attacks.Google Cloud Armor benefits from over a decade of experience protecting some of the world's largest internet properties like Google Search, Gmail, and YouTube.

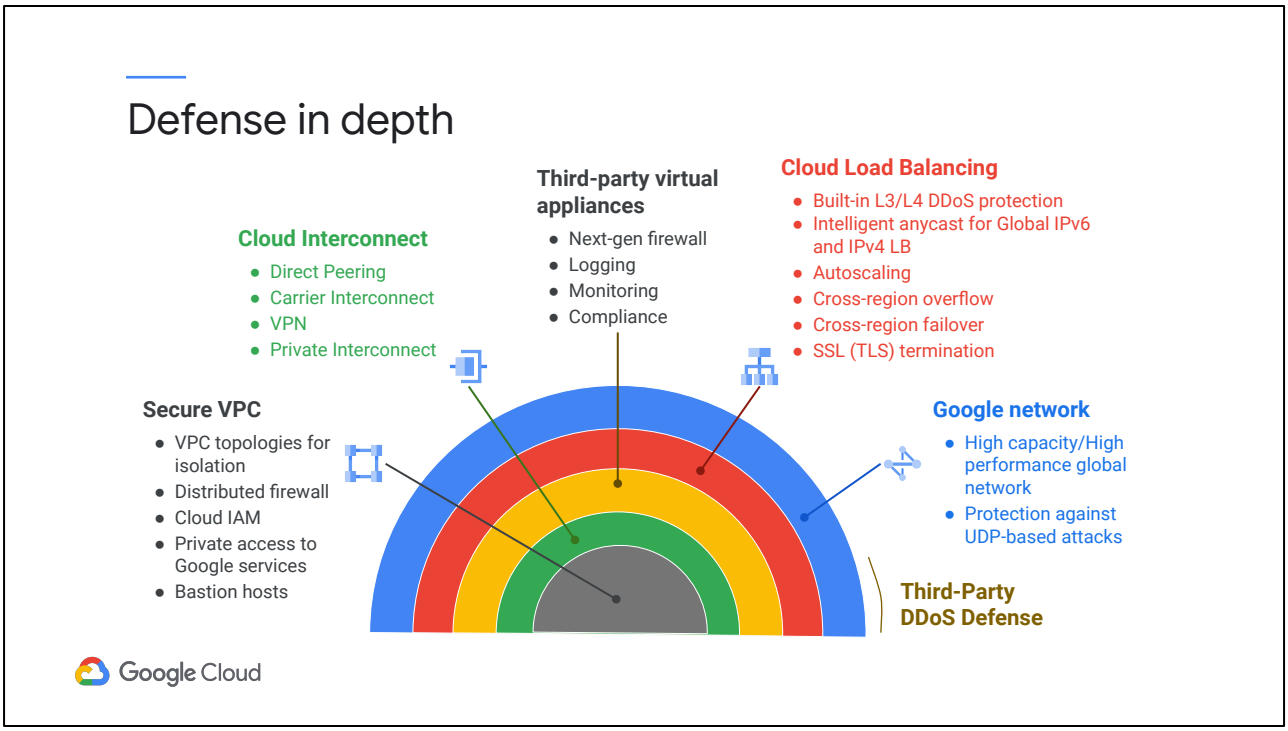
Google Cloud Armor enables you to restrict or allow access to your HTTP(S) load balancer at the edge of the Google Cloud network, meaning as close as possible to the user and to malicious traffic. For example, in this diagram, Sarah in San Francisco and Shen in Singapore are two employees who are allowed to access your HTTP load balancer. Therefore, their traffic will be forwarded to the backend in the us-west2 region and the backend in the asia-southeast1 region, respectively. A DDoS attack, on the other hand, can be blocked directly at the edge without consuming resources or entering your VPC network.

Designing for legal compliance

Exam outline	Tips
Legislation (e.g., Health Insurance Portability and Accountability Act (HIPAA), Children's Online Privacy Protection Act (COPPA), etc.)	PII - personally identifiable information
Audits (including logs)	Do your policies conform to audit requirements?
Certification (e.g., Information Technology Infrastructure Library (ITIL) framework)	Firewall, IAM, and Keys. CSEK? Make sure that growth doesn't break compliance

Google Cloud

What are the two most common compliance areas?
(1) privacy regulations (e.g., HIPAA, GDPR); (2) commercial and line-of-business standards (e.g., PCI-DSS).



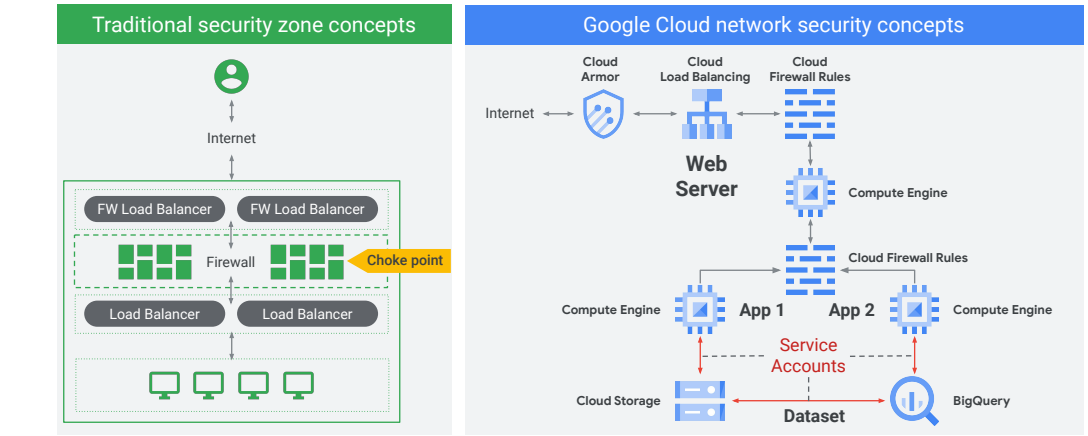
Layers of protection. Each layer protects and complements the next internal layer.

- **Secure VPC:** Identify optimal VPC topology. Deploy distributed firewalls. Control access with IAM permissions. Avoid adding Public IPs to instances. Access Google services internally.
- **Cloud Interconnect:** Connect securely to on-prem or other-cloud deployments. Private Interconnect. Carrier Interconnect. Direct Peering. VPN.
- **3rd party virtual appliances:** Enhance VPC security with 3rd party appliances. Next-gen firewalls. IDS/IPS (intrusion detection). Logging. Monitoring. Scale 3rd party appliances using internal load balancing so you don't create choke points in your VPC.
- **Cloud Load Balancing** provides edge protection and global infrastructure protection for IPv4 and IPv6. Layer 3 and 4 DDoS protection. Anycast IP - even if backends are in multiple regions. To absorb attacks for resiliency - autoscaling, cross-region overflow, and cross-region failover.
- **Google network:** High capacity. High performance. SDN network virtualization Global networks with subnets. Organization, folders, cross-project networking, peering.
- **3rd Party DDOS:** You can complement this infrastructure with additional DDoS security from 3rd party providers.

https://cloud.google.com/security/

https://cloud.google.com/docs/enterprise/best-practices-for-enterprise-organizations

Map from traditional security to cloud security concepts



Key concepts: Cloud Armor, Cloud Load Balancing, Firewall Rules, Service Accounts, separation into front-end and back-end, isolation of resources using separate service accounts between services.

TIP: Because of pervasive availability of firewall rules, you don't have to install a router in the network at a particular location to get firewall protection. That means you can layer the firewalls as shown in this example.

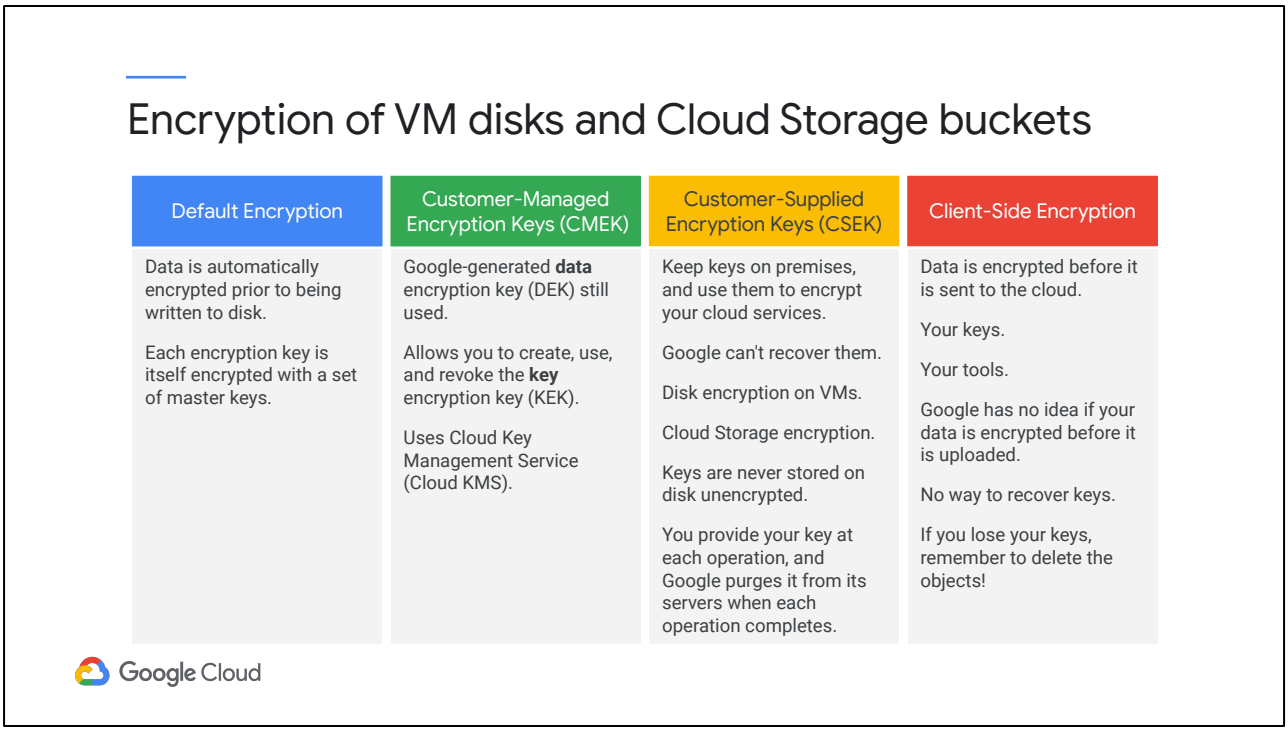
TIP: Because if pervasive support for Service accounts, you can "lock down" connections between components.

Security

1. Security is an "umbrella" term for many specific services. Access. Authorization. Accountability. Privacy. Authentication. Encryption.
2. Layering security measures provides a greater deterrent due to the synergy of multiple methods working together.
3. In some cases, practical security involves raising the effort required to bypass the security above the value of the item being protected. Sometimes called "castle logic". The walls of the castle (and depth of the moat) should be higher (and deeper) than the value of the treasure in the castle.

When faced with a security question on an exam (or in practice), determine which of the specific technologies/services is being discussed (authentication, encryption) for example. Then determine exactly what the goals are for sufficient security. Is it

deterrence? Is it meeting a standard for compliance? Is the goal to eliminate a particular risk or vulnerability? This will help you define the scope of a solution, whether on an exam or in application.



Encryption options
https://cloud.google.com/security/encryption-at-rest/

Customer Managed Encryption Keys (CMEK) *using Cloud KMS*
When you use Dataproc, cluster and job data is stored on Persistent Disks (PDs) associated with the Compute Engine VMs in your cluster and in a Cloud Storage bucket. This PD and bucket data is encrypted using a Google-generated data encryption key (DEK) and key encryption key (KEK). The CMEK feature allows you to create, use, and revoke the key encryption key (KEK). Google still controls the data encryption key (DEK).

Default Encryption
Encryption at rest uses the Key Management System (KMS) to generate KEKs and DEKs.
https://cloud.google.com/security/encryption-at-rest/default-encryption/#key_management

https://pixabay.com/en/key-old-skeleton-lock-metal-door-30417/

Key Management Service (KMS): https://cloud.google.com/kms/
AES256 keys
Generate keys
Usage includes off-cloud
Key rotation

When a key is destroyed, there is a 24-hour delay
API support
Envelope DEK/KEK

Client-Side Encryption
https://cloud.google.com/storage/docs/encryption/client-side-keys

Scenario #1

Question

Which IAM roles apply to security auditors requiring visibility across all projects?

A. Org viewer, project owner

B. Org viewer, project viewer

C. Org admin, project browser

D. Project owner, network admin

Google Cloud

Origin: CAPE

Scenario #1

Answer

Which IAM roles apply to security auditors requiring visibility across all projects?

A. Org viewer, project owner

B. Org viewer, project viewer

C. Org admin, project browser

D. Project owner, network admin

Google Cloud

Origin: CAPE

Scenario #1

Rationale

B - Gives read-only access across the company.

A, C, D - The other options allow them to make changes. They should not be able to make changes.

https://cloud.google.com/iam/docs/understanding-roles

https://cloud.google.com/iam/docs/granting-changing-revoking-access

Scenario #2

Question

A company's security team has decided to standardize on AES256 for storage device encryption. Which strategy should be used with Compute Engine instances?

A. Select SSDs rather than HDDs to ensure AES256 encryption.

B. Use the linux dm-crypt tool for whole-disk encryption.

C. Use Customer Supplied Encryption Keys (CSEK).

D. Use openSSL for AES256 file encryption.

Origin: Case study

Scenario #2

Answer

A company's security team has decided to standardize on AES256 for storage device encryption. Which strategy should be used with Compute Engine instances?

A. Select SSDs rather than HDDs to ensure AES256 encryption.

B. Use the linux dm-crypt tool for whole-disk encryption.

C. Use Customer Supplied Encryption Keys (CSEK).

D. Use openSSL for AES256 file encryption.

Google Cloud

Origin: Case study

Scenario #2

Rationale

A - Selection of disk type determines the default method for whole-disk encryption. HDDs use AES128 and SDDs use AES256.

B - This would be redundant with Compute Engine disk encryption.

C - Who manages the keys has nothing to do with whether it is AES128 or AES256.

D - File encryption is a different layer. The standard is for device encryption.

https://cloud.google.com/compute/docs/disks/customer-supplied-encryption
https://cloud.google.com/security/encryption-at-rest/default-encryption/

Google Cloud

https://cloud.google.com/compute/docs/disks/customer-supplied-encryption
https://cloud.google.com/security/encryption-at-rest/default-encryption/

"In addition to the storage system level encryption described above, in most cases data is also encrypted at the storage device level, with at least AES128 for hard disks (HDD) and AES256 for new solid state drives (SSD), using a separate device-level key (which is different than the key used to encrypt the data at the storage level). As older devices are replaced, solely AES256 will be used for device-level encryption."

Module agenda

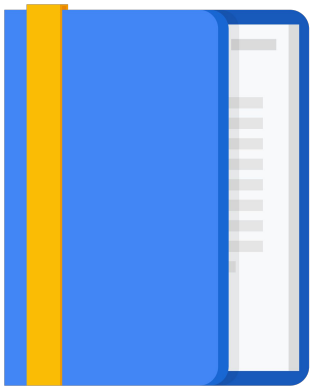
Designing for Security and Compliance

Analyzing and Optimizing Technical and Business Processes

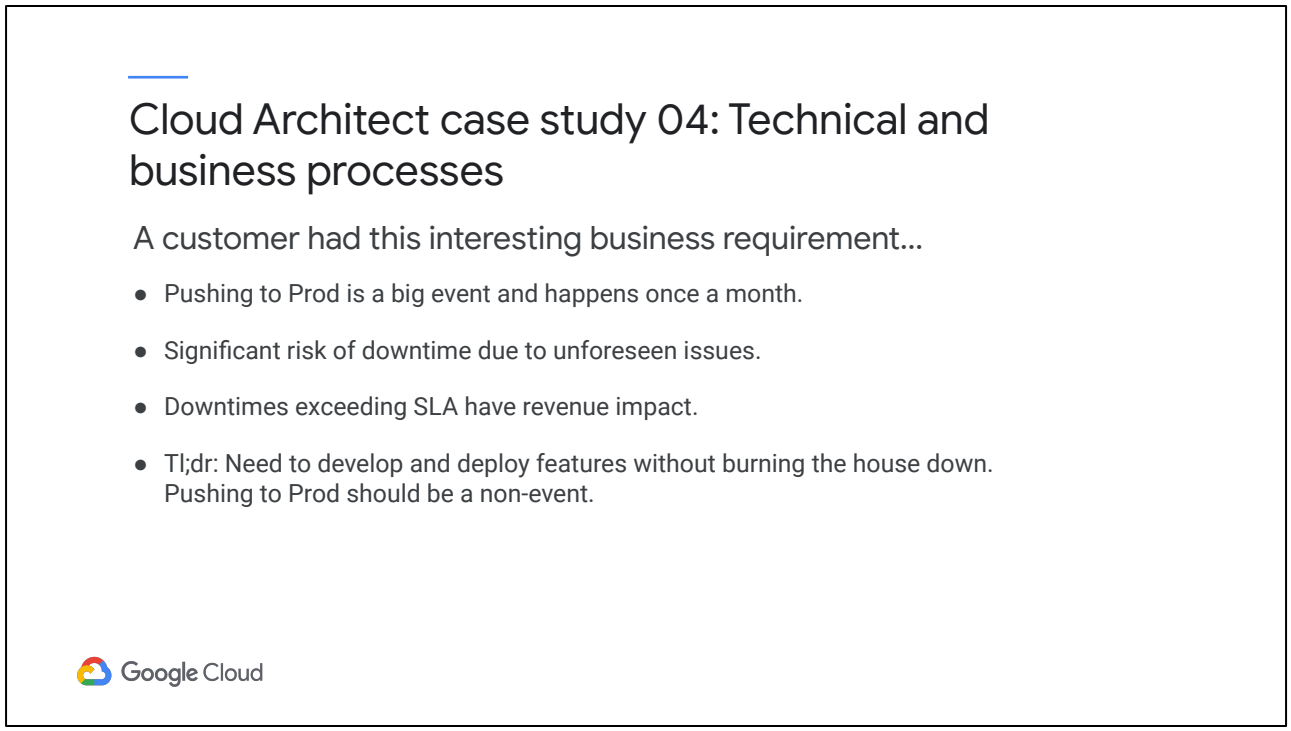
Developing Procedures to Test Resilience of Solution in Production

Managing Implementation

Ensuring Solution and Operations Reliability



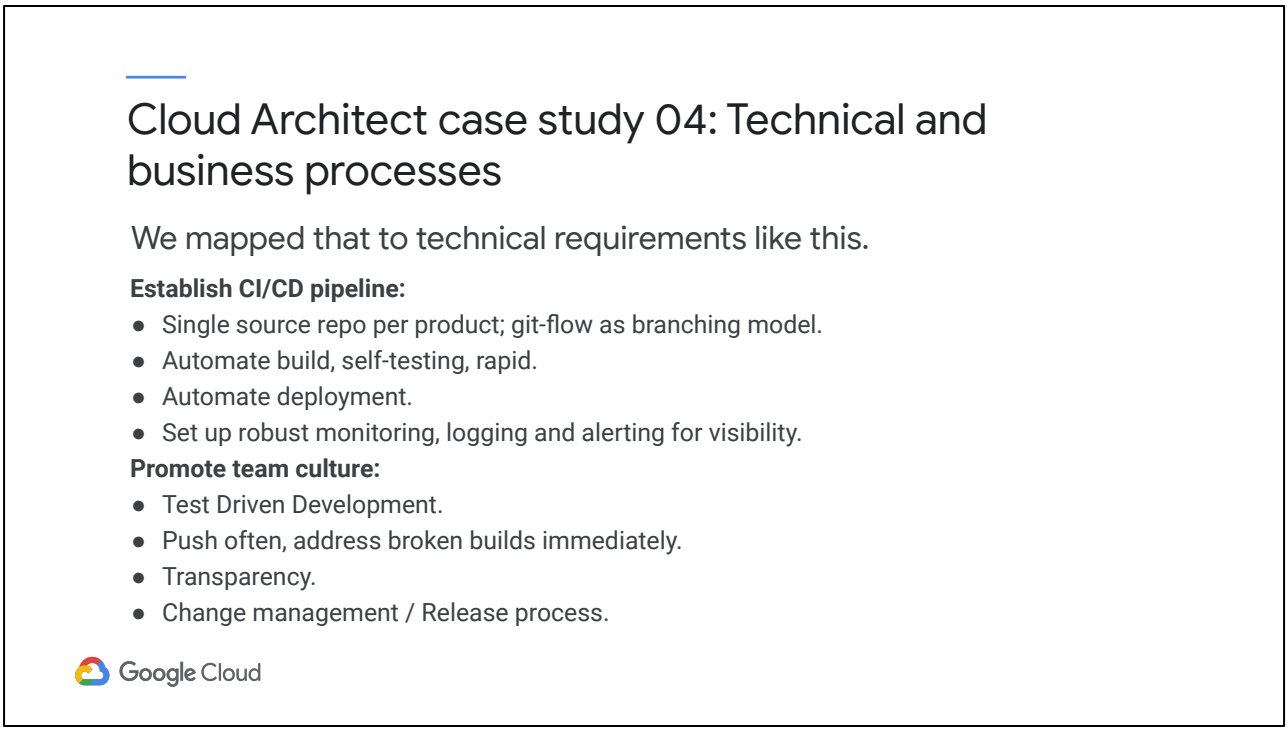
Google Cloud



You may have seen this situation before. The customer experience is that pushing to production is scary, because you never know when things might break. This customer could only push to production once a month. There was significant risk of downtime. And when there was downtime, the application breaks, and that impacts revenue. So the summary is… we need to be able to develop and deploy features, need to present to production, without it being an event.

A customer had this interesting business requirement...

- Pushing to Prod is a big event and happens once a month.
- Significant risk of downtime due to unforeseen issues.
- Downtimes exceeding SLA have revenue impact.
- Need to develop and deploy features without burning the house down. Pushing to Prod should be a non-event.



There are two passes through the problem in the architectural process. First, there is the business pass, which includes both the business challenge and the people processes - understanding what roles there are and what actions the people need to be able to take. And then there is the technical pass which is mapping all these needs and procedures to a technical solution.

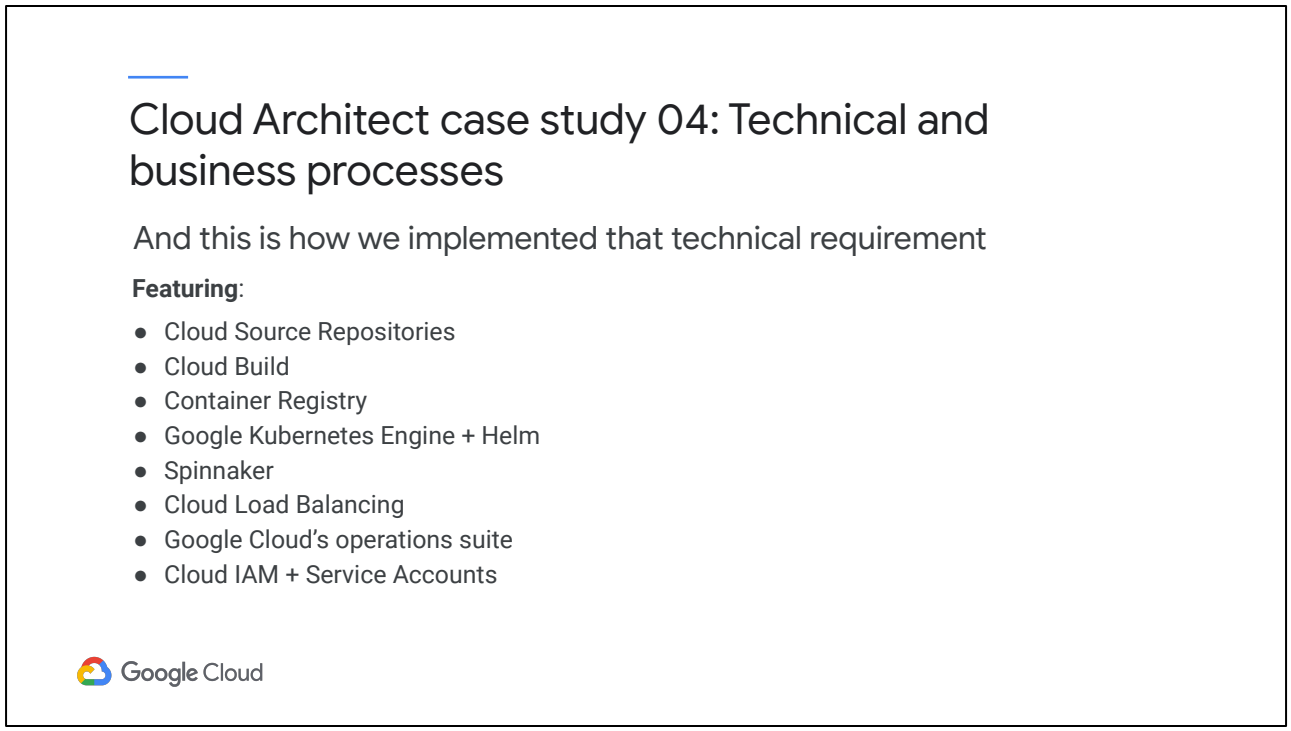
We mapped that to technical requirements like this…

Establish CI/CD pipeline:

- Single source repo per product; git-flow as branching model.
- Automate build, self-testing, rapid.
- Automate deployment.
- Set up robust monitoring, logging and alerting for visibility.

Promote team culture:

- Test Driven Development.
- Push often, address broken builds immediately.
- Transparency.
- Change management / Release process.



To push to production on demand, we needed to analyze the development process. We figured out that a single source repo made the most sense for the whole team. We decided to go with git-flow as a branching model instead of other kinds of development. We automated the build process. But also we made sure that the builds were self-testing using SALT test coverage. And we measured the build process and made sure that it was rapid. We also automated the deployment of the solution software. Couple this automation with monitoring, logging, and alerting, and you get a nice build-and-deploy system that can be handed off to a team.

But that doesn't solve the entire problem. The system has to be used. And this team was not going to be accustomed to using the development paradigm we had implemented. So we also needed to do was enhance their processes. This primarily had to do with how test and development worked together. In the new paradigm they would start writing the testing along with development. The new way was to push to production often and push early, and as soon as something break -- fix it. This meant a new team culture, one of accountability and transparency. Where all the stakeholders could participate in identifying and resolving a problem. And that meant change management and leadership buy-in was necessary for the technical solution to be successful.

And this is how we implemented that technical requirement.

- Cloud Source Repositories - for hosting their repositories
- Cloud Build - that builds the docker container images
- Container Registry - that hosts those container images

- Google Kubernetes Engine + Helm - for running and managing
- Spinnaker - for CDP - continuous deliver
- Cloud Load Balancing, Google Cloud's operations suite, Cloud IAM + Service Accounts - management constructs for proper visibility

Analyzing and defining technical processes

Exam outline	Tips
Software Development Lifecycle Plan (SDLC)	Choosing developer tools on Google Cloud https://cloud.google.com/tools/docs/
Continuous integration/continuous deployment	Kubernetes … Cloud Build
Troubleshooting/post mortem analysis culture	Blame isn't root cause. Focus on systems—what you can change in your procedures, not who is at fault.
Testing and validation	Testing and validation Error budget versus perfection
IT enterprise process (e.g., ITIL)	Information Technology Infrastructure Library defines processes, skills, and handoff of responsibilities
Business continuity and disaster recovery	What's your D.R. story? How do you know it works, if you don't test it? Tradeoffs. What are your SLAs?

Google Cloud

CI/CD with Kubernetes:
https://cloud.google.com/kubernetes-engine/continuous-deployment/

CI/CD tools integrations:
https://cloud.google.com/container-registry/docs/continuous-delivery

https://cloud.google.com/tools/docs/

Testing

Load testing - test overload conditions

Plan and review production SLOs and SLIs

Software Development Lifecycle Plan (SDLC) to identify overload risks

Analyze and define business processes, including policy for degraded service and service outages

Develop procedures to test resilience of solution in overload conditions

Periodic testing

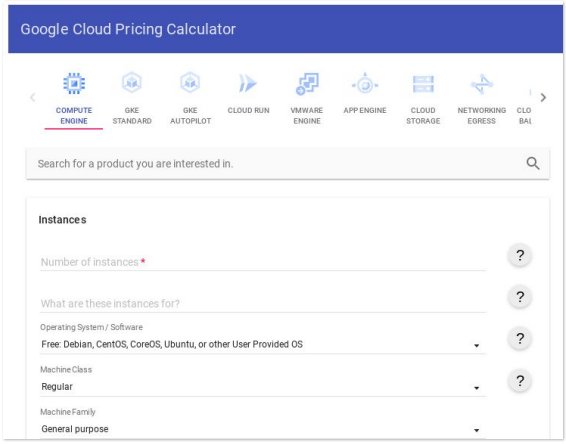
Test recovery processes

Google Cloud

TIP: It is handy to have a testing checklist in mind to help you consider all options. Consider the question(s) you are trying to answer with testing; "will the solution support the number of users", "will it handle peak traffic", "is latency acceptable", and so forth.

- If you can, test during a low use time, such as at night (called a 'dark launch')
- If not, test in pre-production using a synthetic workload that closely resembles a real workload. The results could be misleading if the workload is not designed well:
- Correct mix of read-only and state mutating operations
- Sufficiently diverse workload (with realistic cache hitrate)
- Environment should resemble production
- If expensive operations (e.g. returning large lists) exist, exercise these
- Don't rely on load testing tools that minimize the number of concurrent operations - gives optimistic view of peak throughput

Pricing calculator



https://cloud.google.com/products/calculator/

Google Cloud

TIP: The pricing calculator can be used with BigQuery to estimate the cost of a query before you submit it.

Optimizing VM cost

VM dimensioning: standard, high-CPU, high-mem, GPU, and custom

Sustained use discounts

- Machine-type discounts
- Inferred instance type discount

Committed use discounts

- Compute Engine and Google Kubernetes Engine VMs
- **Not:** App Engine flexible environment, Dataproc, Dataflow, or Cloud SQL VMs

Preemptible VMs

Discounting algorithms are subject to change. Please see current discounting or details:
Committed use discounts:
https://cloud.google.com/compute/docs/instances/signing-up-committed-use-discounts
Sustained use discounts:
https://cloud.google.com/compute/docs/sustained-use-discounts
https://cloud.google.com/compute/docs/instances/preemptible

Optimizing disk cost

Determine how much space you need.

Determine what performance characteristics your applications require.

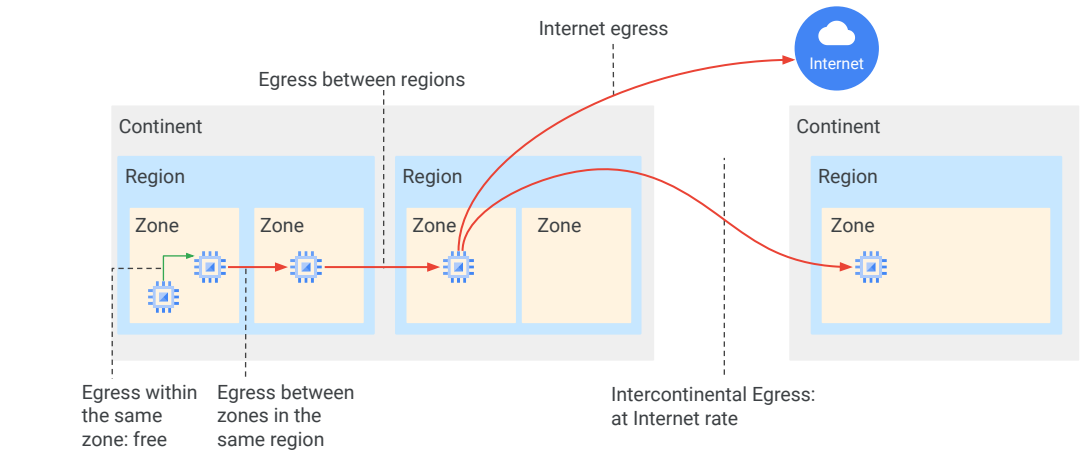
- I/O Pattern: small reads and writes or large reads and writes.
- Configure your instances to optimize storage performance.

Monthly charges	Standard PD	SSD PD
10 GB	$0.40	$1.70
1 TB	$40	$170
16 TB	$655.36	$5,570.56

Google Cloud

https://cloud.google.com/compute/docs/disks/performance

Optimizing network cost



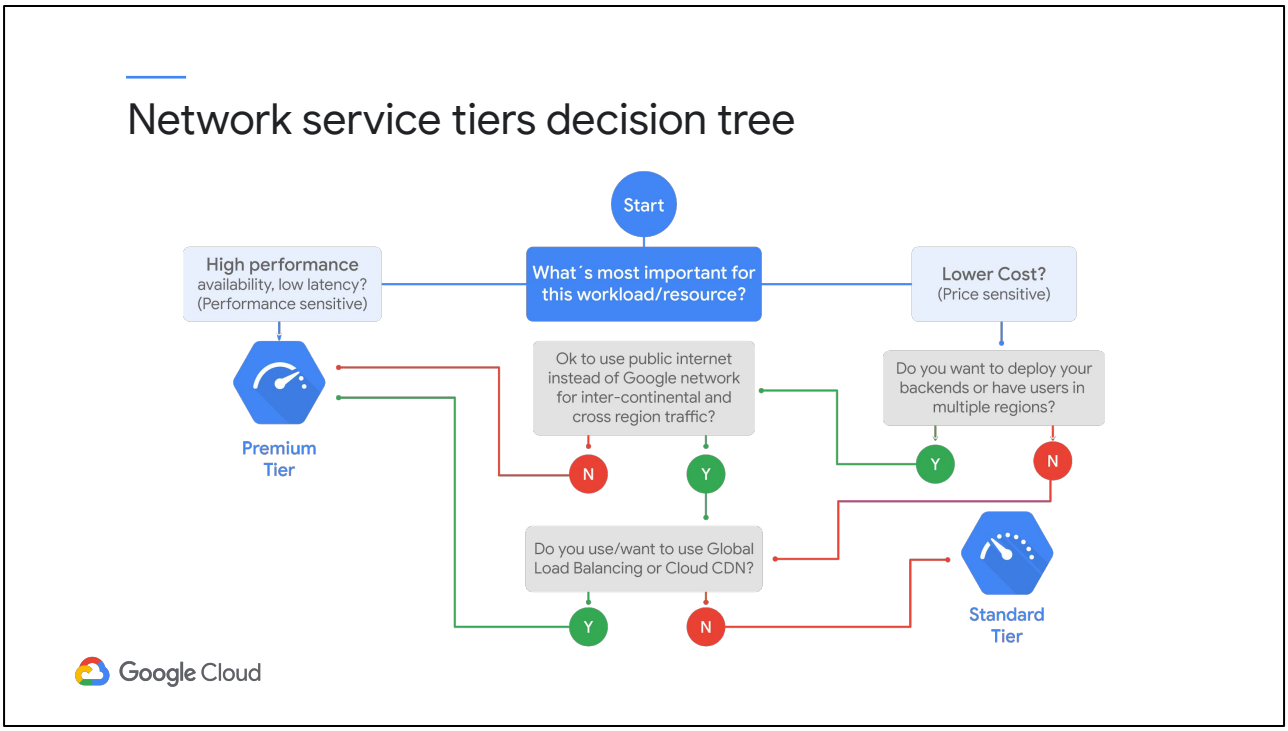
Google Cloud

Ingress is free. Networking costs are similar per Google Cloud product but are billed per product. So you need to view the pricing documentation per product for the details.
Example: Cloud Storage has standard egress costs. But there are also separate charges for data migration and for Cloud Storage operations.

Example: Egress between regions: $0.01 per GB. Egress to Internet: first terabyte to world destinations, $0.12 per GB.

https://cloud.google.com/bigtable/pricing#network

https://cloud.google.com/storage/pricing



There are performance and cost differences between the Network Service Tiers. This decision tree provides guidance on choosing the tier that best meets business needs.

Ask yourself whether high performance or lower cost is most important for your workload or resource. The Premium Tier is the clear choice for performance. If cost is your main consideration, remember that the Standard Tier has other restrictions in addition to network performance. If you want to deploy your backends or have users in multiple regions, but don't want to use the public internet over Google's network for inter-continental and cross region traffic, you want to choose the Premium Tier. Also, if you want Global Load Balancing or Cloud CDN, you need to use the Premium Tier.

Otherwise, the Standard Tier is a great choice if you don't need any of those services and are okay using the public internet instead of Google's network.

Analyzing and defining business processes

Exam outline	Tips
Stakeholder management (e.g., influencing and facilitation)	Who are the gatekeepers and what roles do they play? How can you enable them to do their jobs?
Change management	Quality is a process, not a product. Change is inevitable in cloud. Adapt process to be continuous.
Team assessment / skills readiness	Do you need a playbook? Rehearsals?
Decision-making process	How long do you have to make which decisions?
Customer success management	How do you know that what you delivered is still meeting your customer/client's needs? When is it time for version +1 or version 2.0 ?
Cost optimization / resource optimization (Capex / Opex)	What are the financial goals and how are they measured and reported?

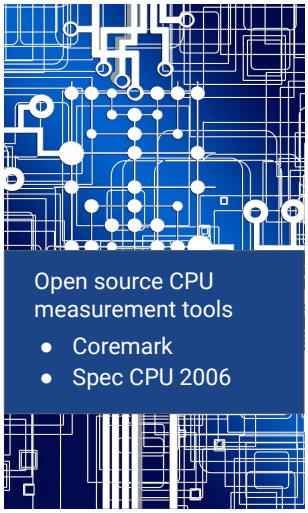
Instance overhead estimation

Of the 100% capacity of a VM, some will be consumed by overhead.

If you can't test, 30% overhead is a cautious estimate.

Load testing can better estimate the overhead.

- **OS image** choice can make a difference (up to 20%).
  - Test several images
- **Hardware architecture** (specific to zone) can make a difference for some workloads.
- **OS firewalls** turn off if Google Cloud firewalls are sufficient.
- On Windows, move **Pagefile** to Local SSD if affordable.



Google Cloud

TIP: If you can exchange a server-oriented architecture for a serverless service, you no longer have to be concerned about instance overhead, just the SLAs of the service.

This page lists the hardware architectures in the different zones. Sandy Bridge, Haswell, Broadwell, IvyBridge, Skylake, and so forth.

https://cloud.google.com/compute/docs/regions-zones/regions-zones

There are a lot of benchmark comparisons online. It is suggested that you test your application and workload in different zones to see what difference the hardware in the zone might make.

Open source CPU measurement tools
Coremark
Spec CPU 2006

https://pixabay.com/en/board-circuits-trace-control-center-1709192/

Persistent Disk estimation

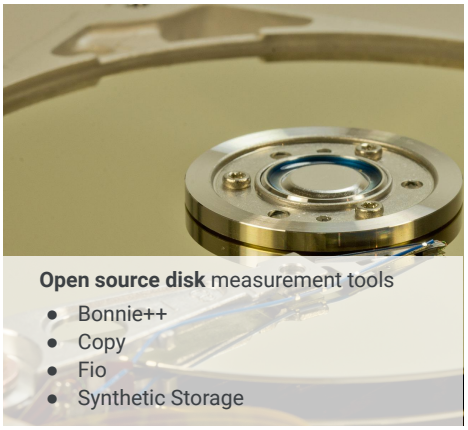
IOPS may be constrained by CPU

- n1-standard-4 = PD-SSD at capacity
- n1-standard-16 = Local SSD at capacity

If estimates are based on averages, consider potential I/O bursts; estimate might be under-provisioned.

Disk performance scales with disk size.

- If you trade up, check for performance over capacity.
- If you trade down, check for performance under capacity.



Open source disk measurement tools

- Bonnie++
- Copy
- Fio
- Synthetic Storage

Google Cloud

TIP: There are new persistent disk features and options. People often assume that a persistent disk is just a hard disk, when it has different features and capabilities. Do you know if disk encryption or RAID configurations would work with Persistent Disks? Would they be valuable?

Consider potential I/O bursts. If you have planned on IOPS based on an average, and the actual disk usage is bursty, the disk could be under provisioned for dealing with the bursts.
Persistent disk performance scales with the size of the disk. So if you trade up to a larger disk in your design, revisit the performance to avoid over capacity, and if you trade disk size down, check for under capacity.
Potential IOPS may be constrained by CPU. An n1-standard-4 can drive a PD-SSD at capacity, and an n1-standard-16 can drive a Local SSD at capacity.

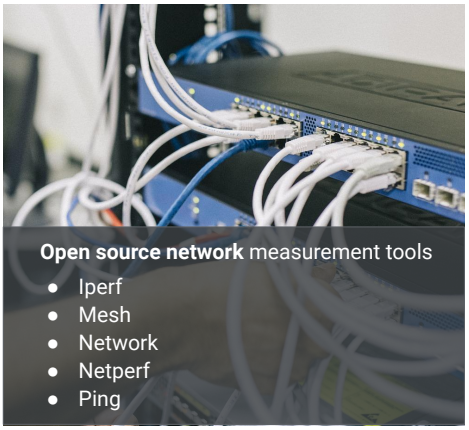
Open source disk measurement tools
Bonnie++
Copy
Fio
Synthetic Storage

Network capacity estimation

Consider potential I/O bursts.

Network capacity scales with the number of cores.

Internal IPs and External IPs are different speeds; they are not symmetrical.



Google Cloud

In general, Internal IPs are faster than External IPs.

Open source network measurement tools
Iperf
Mesh
Network
Netperf
Ping

Consider potential I/O bursts. If you have planned on IOPS based on an average, and the actual traffic is bursty, it could be under provisioned for dealing with the bursts.

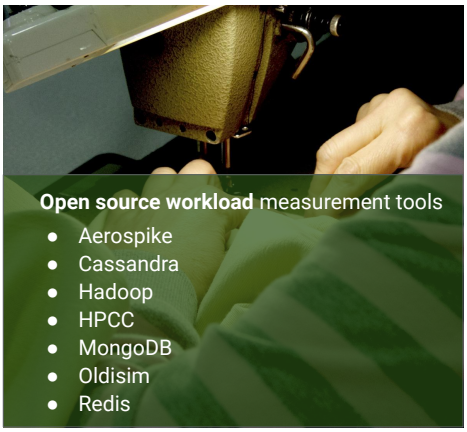
Network capacity scales with the number of cores. So if you change out the number of cores in the VMs in your design, revisit the network capacity to make sure the design does not over- or under- provision.

Internal IPs and External IPs are different speeds, they are not symmetrical. Even VM-to-VM in the same zone over External IPs can be ~1Gbps vs ~8.5Gbps for Internal IPs.

Workload estimation

Throughput depends on:

- Types of requests or operations
- Requests that change state
- Request size (payload)
- Whether system uses sharding, pipelining, or batching



Module agenda

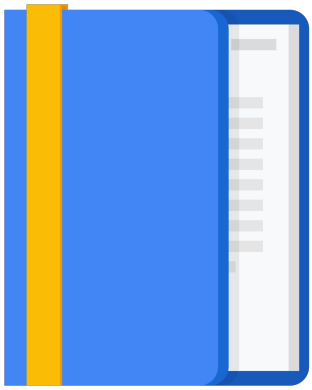
Designing for Security and Compliance

Analyzing and Optimizing Technical and Business Processes

Developing Procedures to Test Resilience of Solution in Production

Managing Implementation

Ensuring Solution and Operations Reliability



Google Cloud

Developing procedures to test resilience of solution in production (e.g., DiRT and Simian Army)

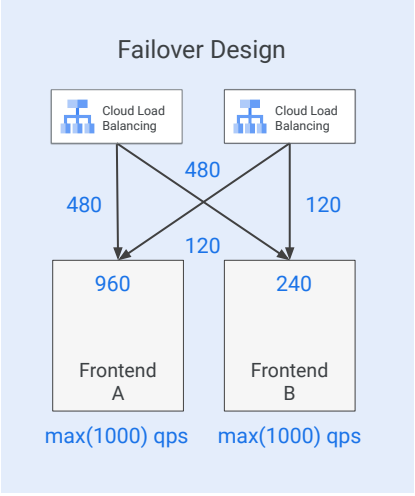
Is this resilient?

Is this a good design for handling failover?

Will it work?

Why or why not?

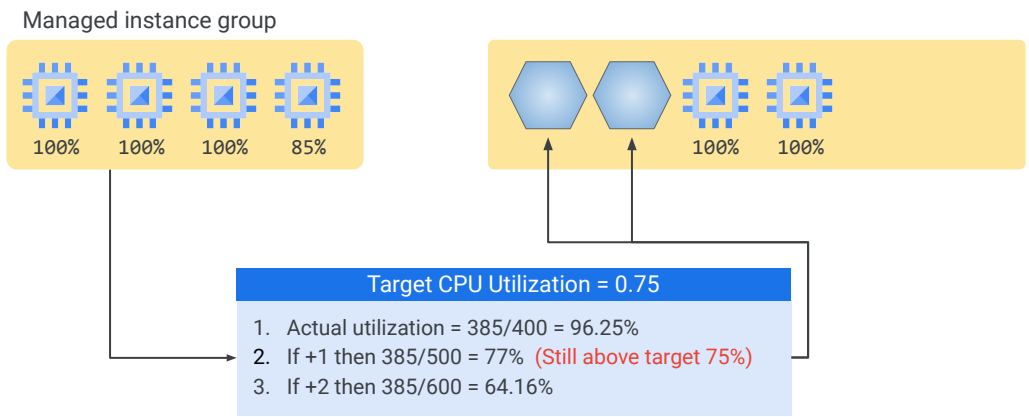
What procedures would you implement?



Google Cloud

This design illustrates that in some cases technical design is not sufficient but must be followed up with human procedures. The original design was intended to handle a maximum of 1000 queries per second. It was originally a functional design. However, over time the load has grown. While the system is still operational, if one of the Frontend servers were to fail and the combined traffic was taken up by the other Frontend, the total would be 1200 qps and would be above capacity. For this reason, resiliency requires identifying key metrics (in this case, total load) and periodically adjusting capacity to stay ahead of growth. On the other side, if load were diminishing consistently over time, there could be cost savings in downsizing the Frontend servers.

Scale-out policy decision



Google Cloud

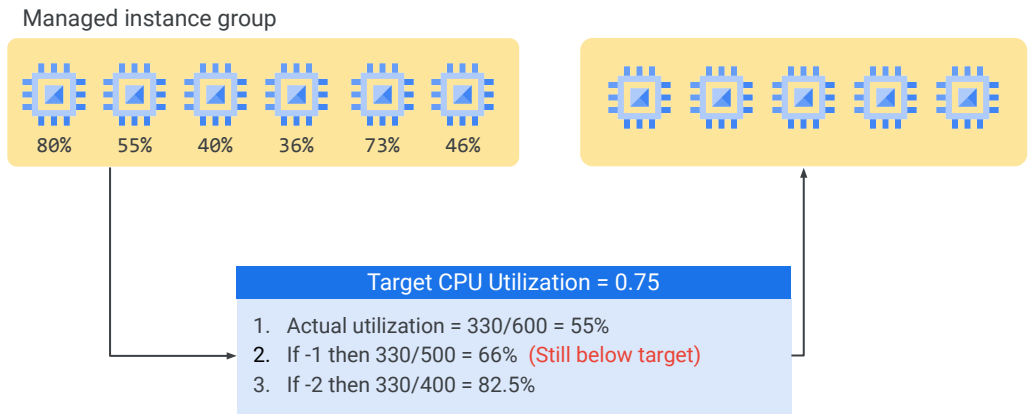
The percentage utilization that an additional VM contributes depends on the size of the group. The 4th VM added to a group offers 25% increase in capacity to the group. The 10th VM added to a group only offers 10% more capacity, even though the VMs are the same size.

In this case shown in the diagram Autoscaler is conservative and rounds up. In other words, it would prefer to start an extra VM that isn't really needed than to possibly run out of capacity.

https://cloud.google.com/compute/docs/autoscaler/understanding-autoscaler-decisions

https://cloud.google.com/compute/docs/autoscaler/multiple-policies

Scale-in policy decision



Google Cloud

In this example, removing one VM doesn't get close enough to the target of 75%. Removing a second VM would exceed the target. Autoscaler behaves conservatively. So it will shut down one VM rather than two VMs. It would prefer underutilization over running out of resource when it is needed.

TIP: When would you use Cloud Monitoring metrics for autoscaling?

Scenario #1

Question

Which of these requirements will Cloud Monitoring dashboards, metrics, and reporting satisfy?

A. Improve security by defining and adhering to a set of security and Identity and Access Management (IAM) best practices for cloud

B. Encrypt data on the wire and at rest.

C. Analyze and optimize architecture for performance in the cloud.

D. Support multiple VPN connections between the production data center and cloud environment.

Origin: Case study

Scenario #1

Answer

Which of these requirements will Cloud Monitoring dashboards, metrics, and reporting satisfy?

A. Improve security by defining and adhering to a set of security and Identity and Access Management (IAM) best practices for cloud

B. Encrypt data on the wire and at rest.

C. Analyze and optimize architecture for performance in the cloud.

D. Support multiple VPN connections between the production data center and cloud environment.

Scenario #1

Rationale

C - Cloud Monitoring metrics will help to analyze and optimize performance for the cloud, because it can be used to gather metrics -- and custom metrics if needed to get to specific behavior of the applications being migrated.

A, B, D - Cloud Monitoring does not necessarily improve security, although alerts could be set to identify problem circumstances that could indicate vulnerabilities.
Cloud Monitoring is incidental to VPN connections to on prem. It doesn't have anything to do with encryption processes.

https://cloud.google.com/stackdriver/docs/

https://cloud.google.com/monitoring/agent/

https://cloud.google.com/monitoring/custom-metrics/

Scenario #2

Question

How can a company connect cloud applications to an Oracle database in its data center to meet its business requirement of up to 10 GB of transactions with an SLA?

A. Implement a high-throughput Cloud VPN connection

B. Cloud Router with VPN

C. Dedicated Interconnect

D. Partner Interconnect

Scenario #2

Answer

How can a company connect cloud applications to an Oracle database in its data center to meet its business requirement of up to 10 GB of transactions with an SLA?

A. Implement a high-throughput Cloud VPN connection

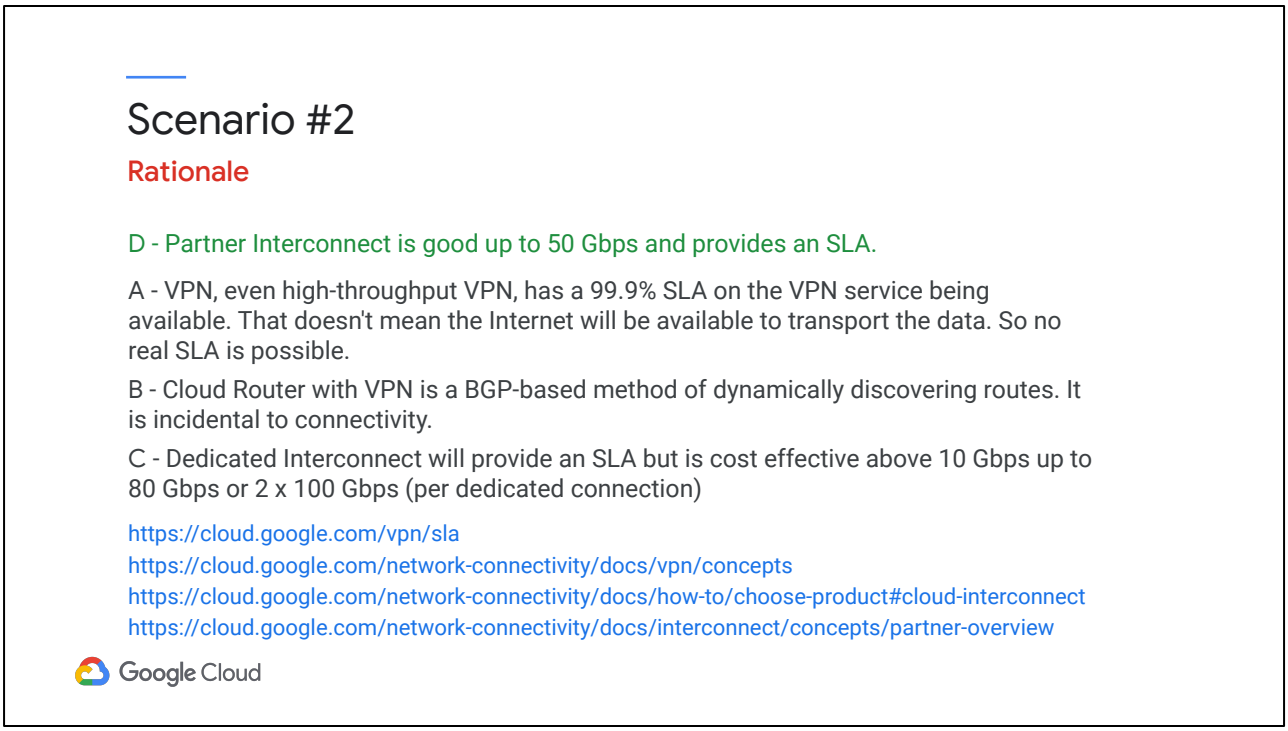
B. Cloud Router with VPN

C. Dedicated Interconnect

D. Partner Interconnect

Google Cloud

Origin: Case study



The first way to distinguish the remaining options is by speed/volume of data.

- Cloud VPN is useful for low-volume connections.
- Partner Interconnect is useful for data up to 50 Gbps.
- Direct Interconnect is useful for data from 10 Gbps to 80 Gbps or 2 x 100 Gbps.

The Cloud VPN SLA covers the availability of the VPN service, not the availability of the public internet.
Business internet SLAs from ISPs is commonly between 99% and 99.5% for a dedicated line.
Therefore, even though the Cloud VPN service is available 99.9% of the time, the communication it relies on will be down between ½% and 1% of the time. That doesn't meet "guaranteed service availability" requirements.

https://cloud.google.com/vpn/sla

https://cloud.google.com/network-connectivity/docs/vpn/concepts

The VPN SLA provides 99.9% *service availability*. That doesn't mean that the you will be able to communicate over the VPN. Because the VPN is dependent on the availability of the Internet.

https://cloud.google.com/network-connectivity/docs/how-to/choose-product#cloud-interconnect

Dedicated Interconnect:

- 10 Gbps to 80 Gbps, or 2 x 100 Gbps (200 Gbps),
- Google offers end-to-end SLA for the connectivity between your VPC and

- on-premises networks for Google-defined topologies.

Partner Interconnect:

https://cloud.google.com/network-connectivity/docs/interconnect/concepts/partner-overview

- Up to 50 Gbps
- Google offers two SLA's that apply only to the connectivity between your VPC network and the service provider's network.
- One is 99.99% and the other is 99.9%
- The SLA doesn't include the connectivity between your network and the service provider's network.

Module agenda

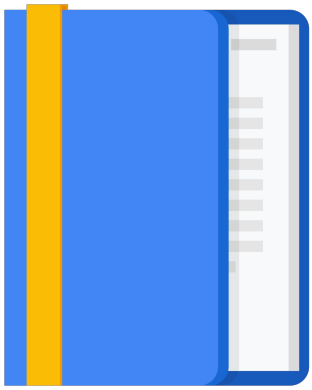
Designing for Security and Compliance

Analyzing and Optimizing Technical and Business Processes

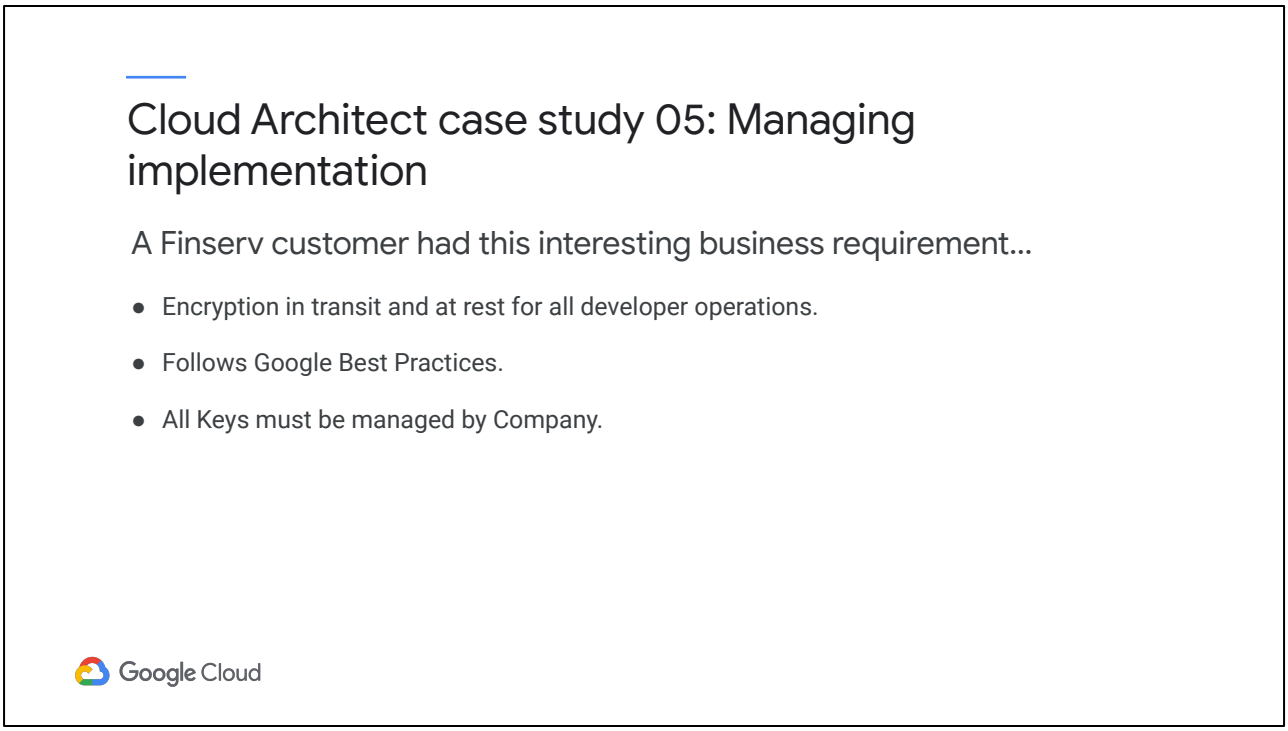
Developing Procedures to Test Resilience of Solution in Production

Managing Implementation

Ensuring Solution and Operations Reliability



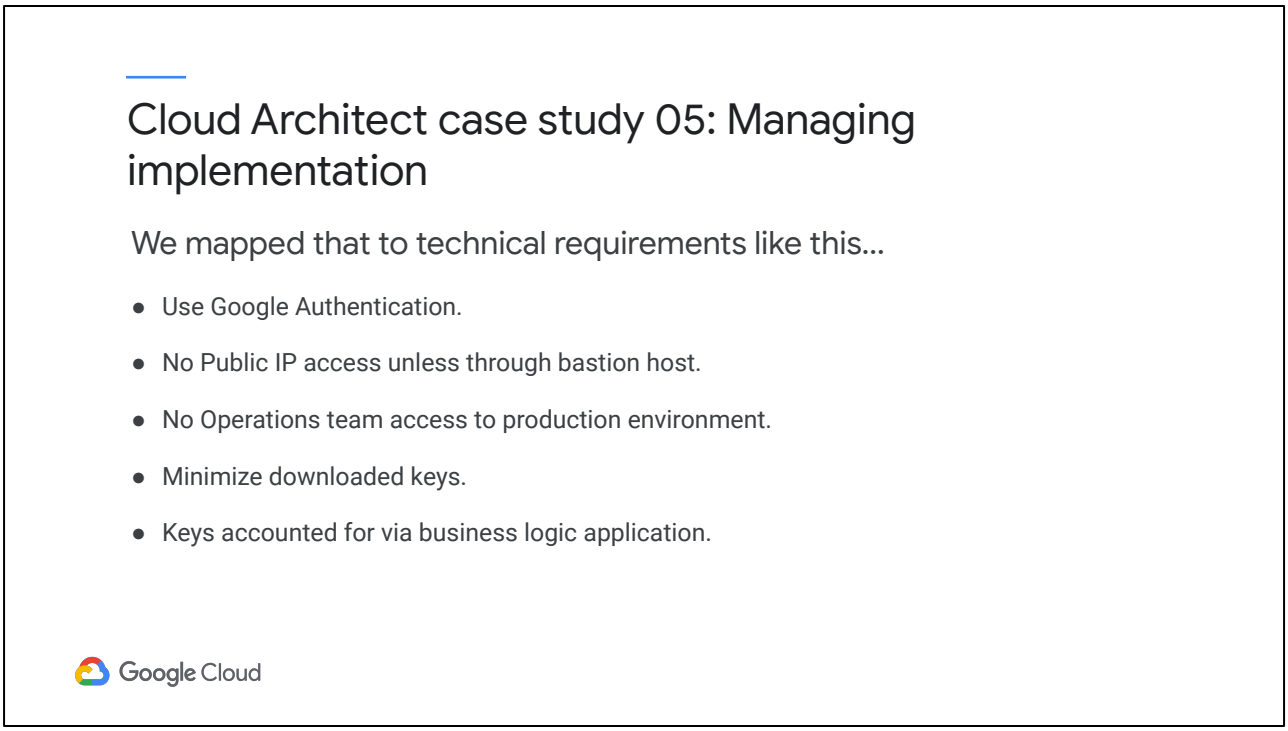
Google Cloud



A lot of the customers we see are in the Enterprise space, so their needs are very similar. This example came from a Financial Services company. We often see similar requirements among FinServ companies.

A Finserv customer had this interesting business requirement...

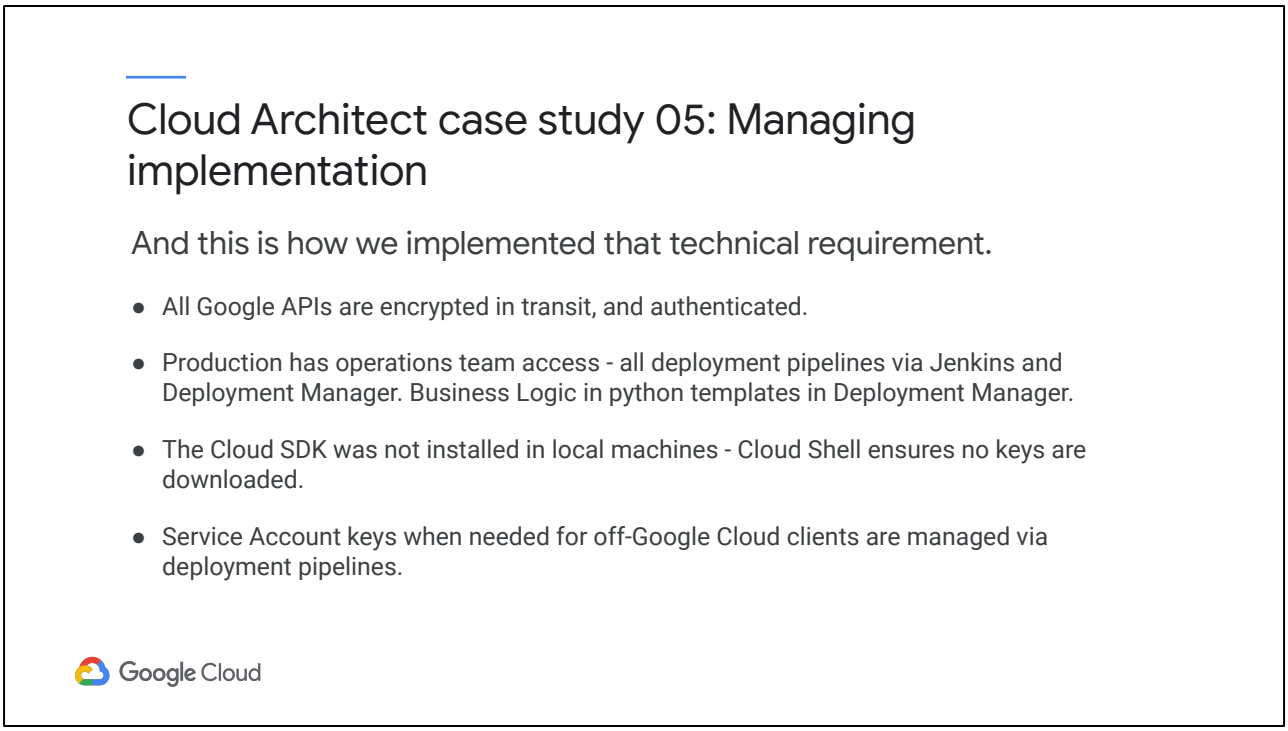
- Encryption in transit and at rest for all developer operations
- Follows Google Best Practices
- All Keys must be managed by Company - they wanted to own the keys



The real trick here is that the structure and solution had to be put into production at one time. It couldn't be built in parts into production. It had to be all working when it went into production. That caused us to think about what parts were inherent, and what parts we could automate. So we ended up using a Jenkins pipeline and Deployment Manager templates for parts of this automation.

We mapped that to technical requirements like this…

- Use Google Authentication.
- No Public IP access unless through bastion host.
- No Operations team access to production environment. That means "no ops" - everything is automated.
- Minimize downloaded keys.
- Keys accounted for via business logic application.



All of the Google APIs are encrypted in transit, and authenticated. So that requirement was inherited and automatic. The production team needed operations access but without handing them keys. So what we did is implemented all operations in deployment pipelines using Jenkins and Deployment Manager. The business logic was implemented using python in the Deployment Manager templates.

And this is how we implemented that technical requirement.

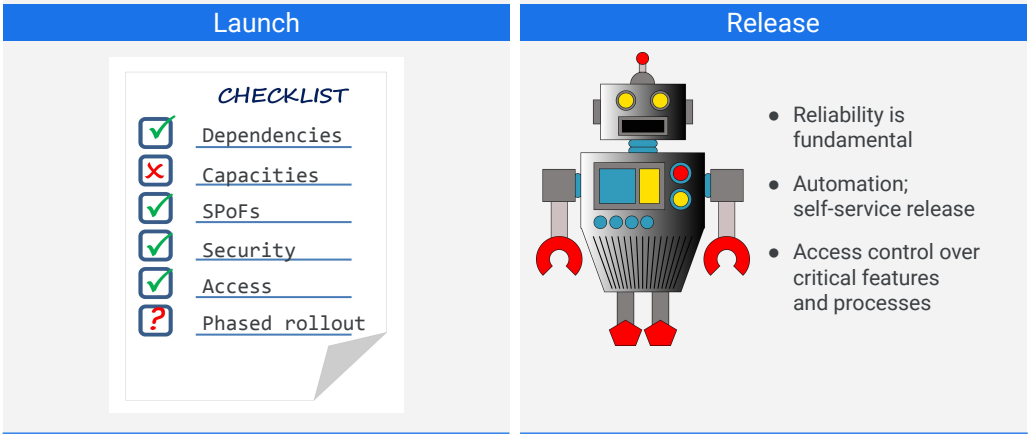
- All Google APIs are encrypted in transit, and authenticated.
- Production has operations team access - all deployment pipelines via Jenkins and Deployment Manager. Business Logic in python templates in Deployment Manager.
- Cloud SDK was not installed in local machines - Cloud Shell ensures no keys are downloaded.
- Service Account keys when needed for off-Google Cloud clients are managed via deployment pipelines.

There are two kinds of operations actions; on-Google Cloud actions and off-Google Cloud actions. For on-Google Cloud actions, we didn't install Cloud SDK on local machines. Instead, we set them up to use Cloud Shell. That ensured that no keys were downloaded. For off-Google Cloud actions,the Service Account keys were managed via the deployment pipelines. Any time there was a need for off-Google Cloud access, the clients are managed via the deployment pipelines. So that means there is full audit, control, and records of those keys, who had access to them, and when and where they were used.

Advising development/operation teams to ensure successful deployment of the solution

Exam outline	Tips
Application development	Kubernetes, App Engine, Containers
API best practices	Be familiar with the Cloud SDK and how it works for automating infrastructure. Be familiar with ML APIs such as the Cloud Natural Language API.
Testing frameworks (load/unit/integration)	Be familiar with the different kinds of testing: black box/clear box, unit, integration. Be able to differentiate them.
Data and system migration tooling	Especially Be familiar with the features for migrating or synchronizing data between on-premises or non-Google Cloud cloud and Google Cloud.

Preparing for deployment and release management



Create a launch checklist. 08 ArchDP Deployment, monitoring and alerting, and incident response v1.0.3, Slide 5.

Dependencies, Capacities, Single Points of Failure, Security and access, Rollout plan (phased)

Release Management

- Create a launch checklist. 08 ArchDP Deployment, monitoring and alerting, and incident response v1.0.3, Slide 6.
- Automate everything you can. Reliability is key. Self-service release process. Implement access control over critical release features/processes.

Capacity planning for launch

Test first

Work through issues before serving user traffic

- Identify bottlenecks

Slow, staged, iterative

- Dark launches
- Use invitations to stage launch

Google Cloud

Use invitations to slow or stage a launch.

Interacting with Google Cloud using the Cloud SDK (gcloud, gsutil and bq)

Exam outline	Tips
Local installation	Install tools on a computer or on a VM. Example: gsutil for backup to Cloud Storage from data center.
Cloud Shell	DevOps interface. Cloud-based shared VM with tools pre-installed. Convenient, but no SLA. So use a VM for reliability because Compute Engine has an SLA.

Scenario #1

Question

Implement back-out/rollback for website with 100s of VMs. Site has frequent critical updates.

A. Create a Nearline copy of static data in Cloud Storage.

B. Create a snapshot of each VM prior to update, in case of failure.

C. Use managed instance groups with the “rolling-action start-update” command when starting a rolling update.

D. Only deploy changes using Deployment Manager templates.

Origin: CAPE

Scenario #1

Answer

Implement back-out/rollback for website with 100s of VMs. Site has frequent critical updates.

A. Create a Nearline copy of static data in Cloud Storage.

B. Create a snapshot of each VM prior to update, in case of failure.

C. Use managed instance groups with the “rolling-action start-update” command when starting a rolling update.

D. Only deploy changes using Deployment Manager templates.

Google Cloud

Origin: CAPE

Scenario #1

Rationale

C - Allows compute engine to handle updates. Easy management of VMs.
Website with 100's of VMs. Load balanced. Likely already using a managed instance group.

D - Large overhead and chance for version conflicts between DM templates if an old template is changed that running infrastructure relies on.

B - A valid approach. But with 100s of VMs, it will be slow and storing all those images will be expensive. The recommendation is to create snapshots outside of normal production periods to avoid the slow down. More data = larger snapshots, slower to create.

A - Unreliable recovery method. Can't roll back once the copy is overwritten.

https://cloud.google.com/compute/docs/instance-groups/updating-managed-instance-groups
https://cloud.google.com/compute/docs/disks/restore-and-delete-snapshots

Scenario #2

Question

A car reservation system has long-running transactions. Which one of the following deployment methods should be avoided?

A. Execute canary releases.

B. Perform A/B testing prior to release.

C. Introduce a blue-green deployment model.

D. Introduce a pipeline deployment model.

Google Cloud

Origin: Experience

Scenario #2

Answer

A car reservation system has long-running transactions. Which one of the following deployment methods should be avoided?

A. Execute canary releases.

B. Perform A/B testing prior to release.

C. Introduce a blue-green deployment model.

D. Introduce a pipeline deployment model.

Google Cloud

Origin: CAPE

Scenario #2

Rationale

C - Switching the load balancer from pointing at the green "good" environment to the blue "new" environment is a fast way to rollback if there is a problem during release. However, long-running transactions will be disrupted by that switch.

A - Testing the application with a few users before releasing to everyone will detect problems early and confine their impact.

B - Performing testing of features "A" with the feature, "B" without the feature, will detect problems before release.

D - Pipeline deployment - introducing orderly procedures into the QA process can improve the effectiveness of QA.

https://cloud.google.com/load-balancing/docs/
https://cloud.google.com/load-balancing/docs/backend-service

Google Cloud

The second link discusses long-running connections and how to support them.

Module agenda

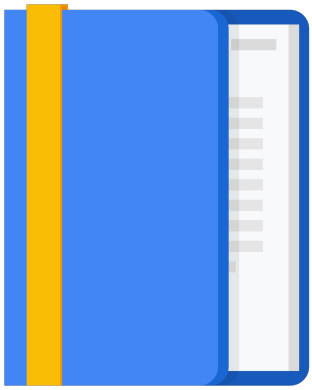
Designing for Security and Compliance

Analyzing and Optimizing Technical and Business Processes

Developing Procedures to Test Resilience of Solution in Production

Managing Implementation

Ensuring Solution and Operations Reliability



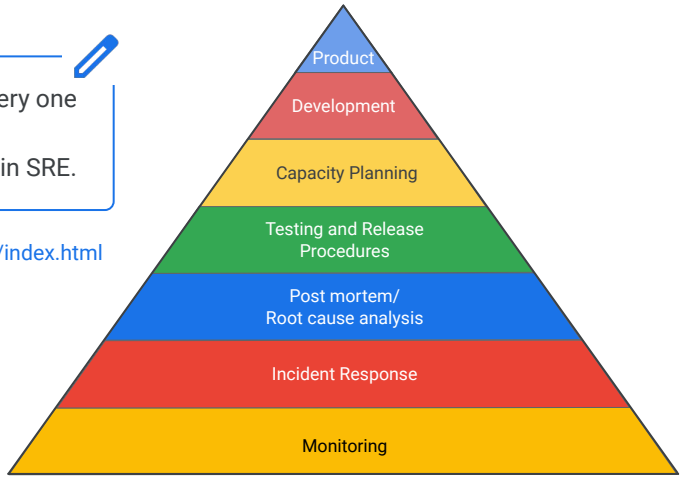
Google Cloud

Site Reliability Engineering

TIP

You need to know about every one of these layers and their order—how work is divided in SRE.

https://landing.google.com/sre/book/index.html



Google Cloud

Site Reliability Engineering or SRE

https://landing.google.com/sre/

The SRE Book -- Read Online for FREE:
https://landing.google.com/sre/book/index.html

Do you want to automate infrastructure or workflow?

Infrastructure Automation

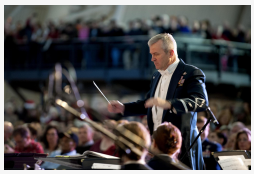
Deployment Manager

- Terraform
- Ansible
- Chef
- Puppet

Workflow Orchestration

Cloud Composer

- Apache Airflow



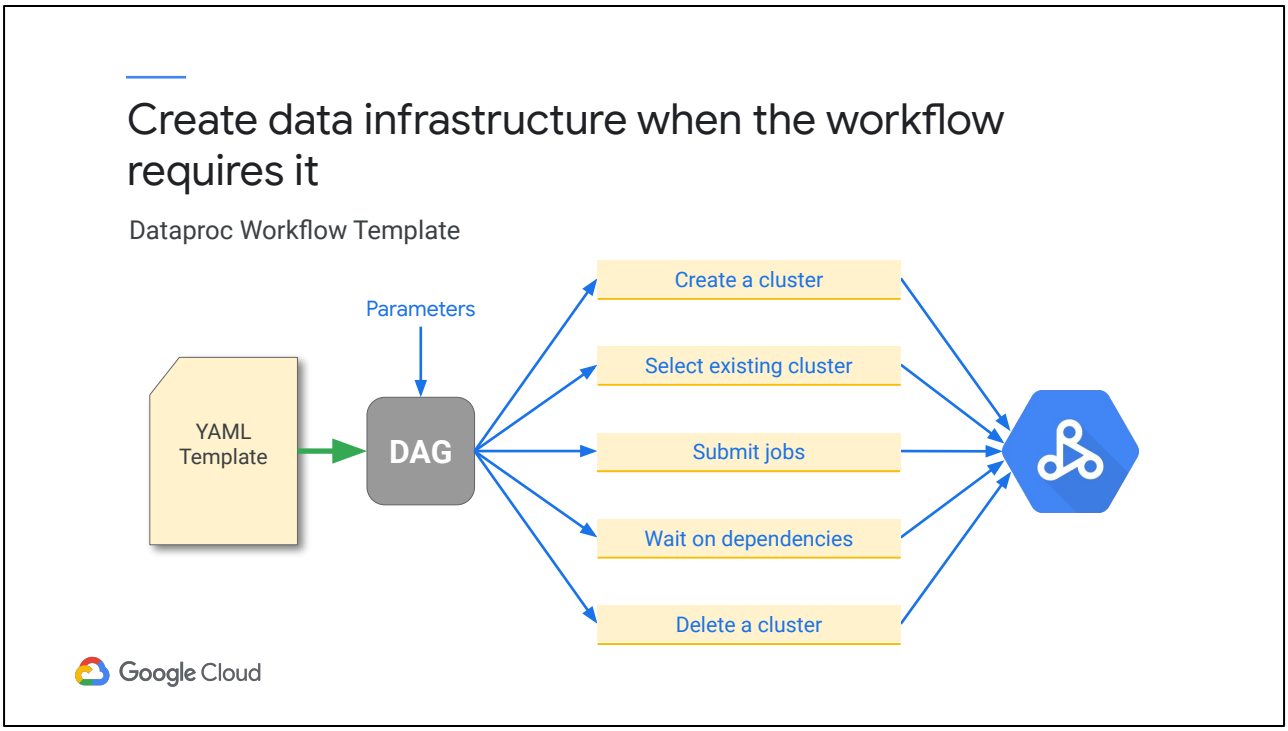
Google Cloud

Infrastructure automation tools like Deployment Manager make creating cloud infrastructure manageable and repeatable and provide a method of documenting the infrastructure which is a common business requirement.

Workflow orchestration is the idea that most business applications rely on passing data from one area of work concentration to another. For example, in the supply chain raw materials are ordered. In the factory, the materials are manufactured into products. At the loading dock the products are shipped. And in the store the products are sold. Each of these parts of the business have systems associated with them: resource management, manufacturing control, shipping and receiving, sales. And those systems need to share information with one another to operate the business. The overall solution is more about workflow between these systems than about the infrastructure of any single system.

If you start to consider workflow in your overall design, you will start to take control over the overall solution and the reliability and methods of operations of the total system.

Q: Why should you have a conductor for your orchestra?
A: To ensure solution and operations reliability.



A Dataproc Workflow Template is a simple example if the crossover between workflow and infrastructure. In this case, the workflow (scheduled or on demand) causes the service to create a Dataproc cluster to process data, and then deletes the cluster when the work is done. **Infrastructure becomes dynamic in a workflow context.**

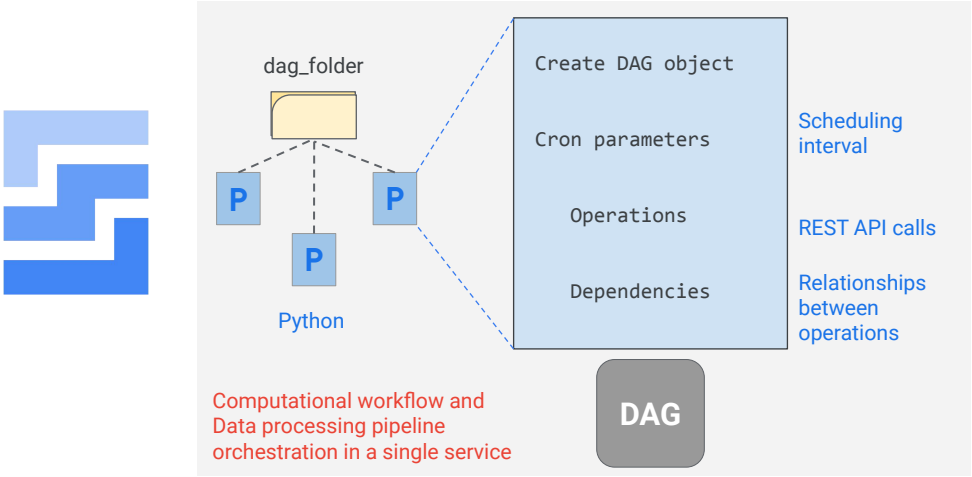
Details:

The Dataproc Workflow Template is a YAML file that is processed through a Directed Acyclic Graph (DAG). It can create a new cluster, select from an existing cluster, submit jobs, hold jobs for submission until dependencies can complete, and it can delete a cluster when the job is done.

It is currently available through the gcloud command and the REST API, but not through Console.

The Workflow Template becomes active when it is instantiated into the DAG. The Template can be submitted multiple times with different parameter values. You can also write a template inline in the the gcloud command, and you can list workflows and workflow metadata to help diagnose issues.

Cloud Composer: Extensible workflow orchestration



Google Cloud

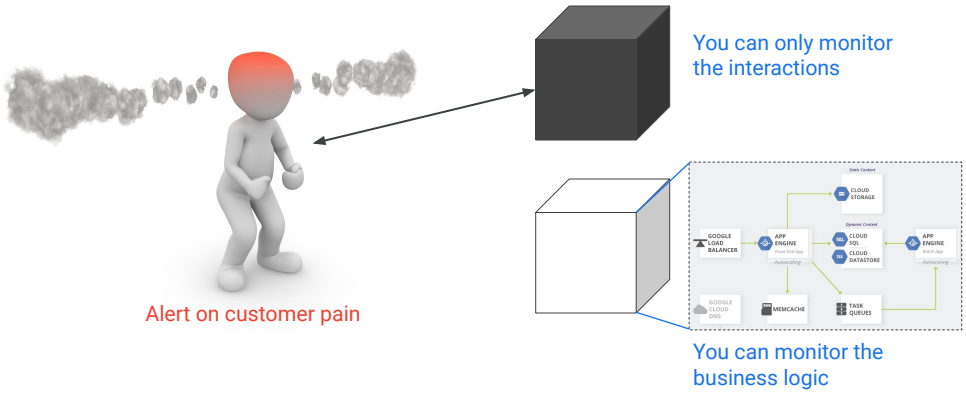
Cloud Composer, based on Apache Airflow, is a service that provides extensible dependency management for complex workflows. Because the Directed Acyclic Graphs (DAGs) are written in Python, Cloud Composer can be extended to coordinate and orchestrate anything with a Python-compatible API -- *which are most services today*.

In Airflow/Composer, there can be multiple DAGs, and they are defined in Python. Each DAG lives in the dag_folder.

A few notes about Cloud Composer:

- Workflow orchestration
  - Concerned with the instructions necessary to complete each step.
  - Computational workflow
  - Data processing pipeline
  - Dependency management
  - Extensible operators
    - operations --> REST APIs

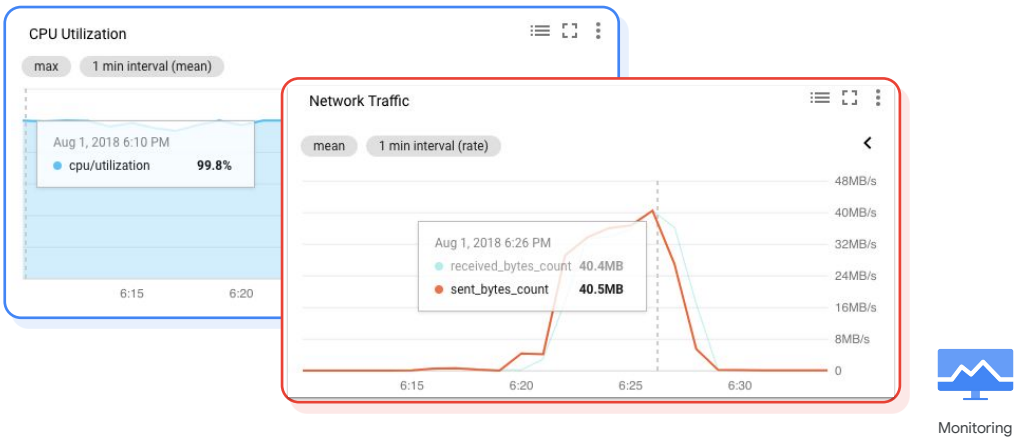
Monitoring/Logging/Alerting solution



Google Cloud

TIP: Make sure you know the difference between black box monitoring and clear box monitoring.

Dashboards can visualize utilization and network traffic



Google Cloud

Cloud Monitoring allows you to create custom dashboards that contain charts of the metrics that you want to monitor. For example, you can create charts that display your instances' CPU utilization, the packets or bytes sent and received by those instances, and the packets or bytes dropped by the firewall of those instances.

In other words, charts provide visibility into the utilization and network traffic of your VM instances, as shown on this slide. These charts can be customized with filters to remove noise, groups to reduce the number of time series, and aggregates to group multiple time series together.

For a full list of supported metrics, please refer to the documentation: https://cloud.google.com/monitoring/api/metrics_gcp

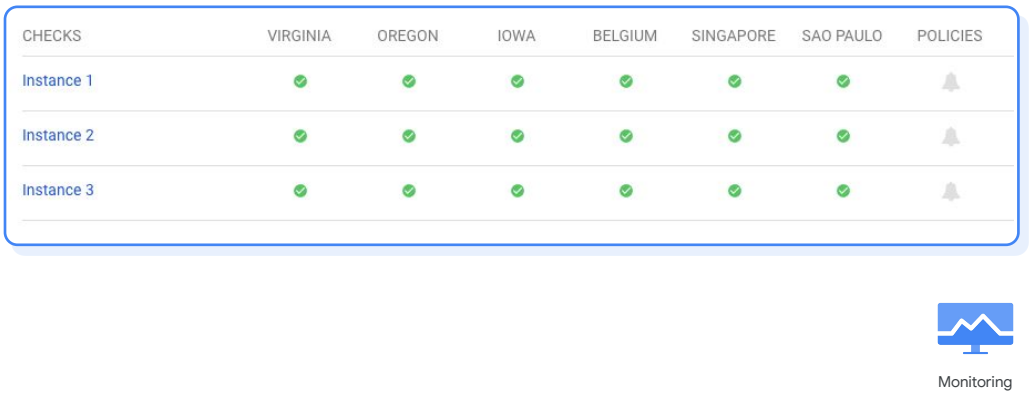
Alerting policies can notify you of certain conditions

Google Cloud

Now, although charts are extremely useful, they can only provide insight while someone is looking at them. But what if your server goes down in the middle of the night or over the weekend? Do you expect someone to always look at dashboards to determine whether your servers are available or have enough capacity or bandwidth? If not, you want to create alerting policies that notify you when specific conditions are met.

For example, as shown on this slide, you can create an alerting policy when the network egress of your VM instance goes above a certain threshold for a specific timeframe. When this condition is met, you or someone else can be automatically notified through email, SMS, or other channels in order to troubleshoot this issue.

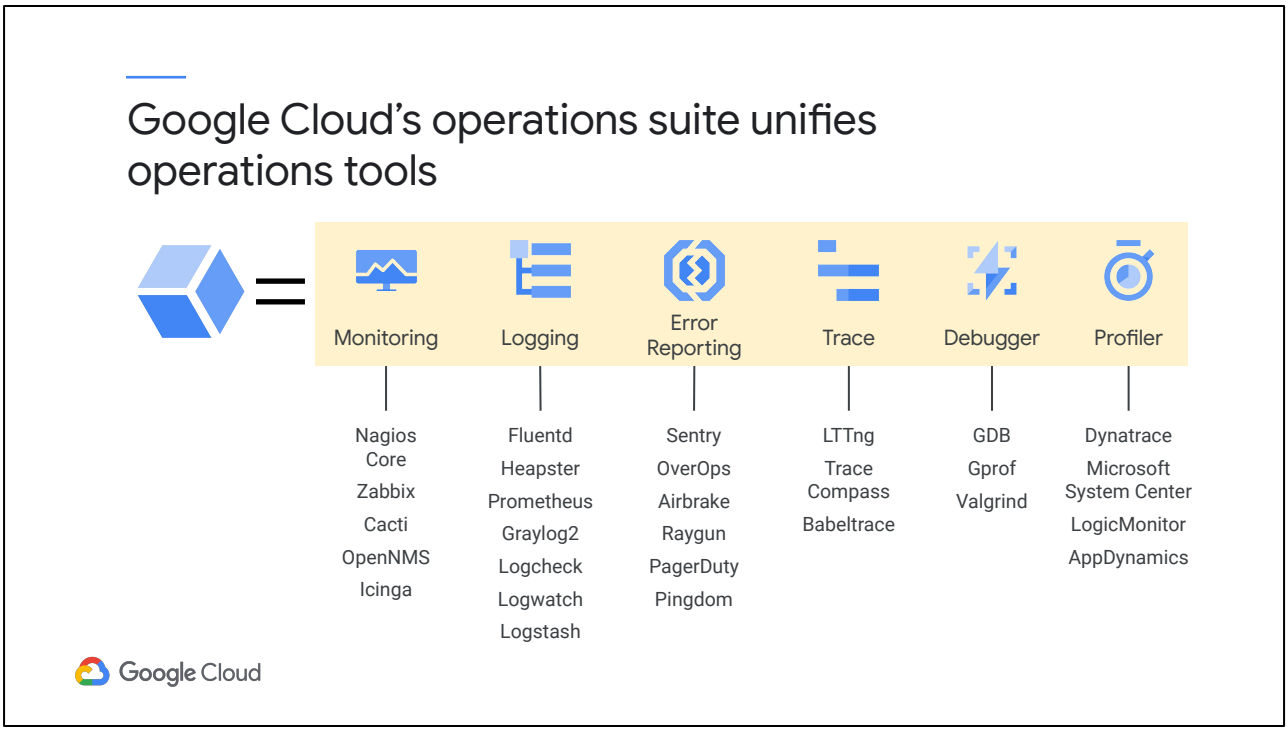
Uptime checks test the availability of your public services



Google Cloud

There are also uptime checks that you can configure to test the availability of your public services from locations around the world, as you can see on this slide. The type of uptime check can be set to HTTP, HTTPS, or TCP. The resource to be checked can be an App Engine application, a Compute Engine instance, a URL of a host, or an AWS instance or load balancer.

For each uptime check, you can create an alerting policy and view the latency of each global location.



TIP: Know how to use Trace and Debug

Examples of other tools that Google Cloud's operations suite replaces. Note that it is not just a collection of alternate tools that is the issue, but how you use them together. The individual tools are not integrated or designed to work together, so a lot of manual procedures and translation/massaging of data are required to use them together. With Google Cloud's operations suite, the integration is by design so that work disappears. It is is also multi-cloud, able to manage projects across Google Cloud and AWS.

Playbooks and "Easy" buttons

Google Cloud

TIP: Another old saying: "People don't plan to fail, they fail to plan." Another way of saying this is "The only time we have to prepare for emergencies is BEFORE they happen. Once the emergency is occurring it is too late to prepare." You can design a great technical solution, but if it doesn't include human processes then it might not be adaptive and resilient.

Easy buttons are tools and processes that automate common actions.
A playbook is a list of what to do when.

General rule: For every alert you should have a play in the playbook.

08 ArchDP Deployment, monitoring and alerting, and incident response v1.0.3, slides 25 and 26.

Use microservices with RESTful APIs
Use Google Cloud APIs
Drain a service for maintenance

For every alert the monitoring system can generate, you should have a Playbook entry.
Standardize the format of the playbook.
Write for newly onboarded employees -- don't assume any company-specific knowledge.

Develop a structured incident response

Monitoring dashboard

Alerting regimen

Plans and tools for responding to issues

Google Cloud

TIP: What are the differences between a Dashboard, an Alert, and Incident Response?

Being prepared: Fast response. Know what to do when.
Consistency: Reduce duplication of effort. Know who is doing what.
Rehearsal: practice incident response. Turn incidents into training -- learn from them -- what worked and what didn't.

SRE tip: Rehearse. The time to make sure everything is working and people know their procedures is BEFORE an emergency occurs, not during it.

https://docs.google.com/presentation/d/10og__CX4qygNG0mpH7GOphpkPeXOQ3gRJ5dlzGKcbQs/edit#slide=id.g250387a28f_0_4050

- Plans and tools for responding to issues
- A set of EASY buttons

Supporting operational troubleshooting

Log

Trace

Debug

Google Cloud

TIP: Find a lab that uses logging and trace and debug to identify and solve an application problem. This will give you a sense of the value and how these components work together.

https://cloud.google.com/logging/docs/

https://cloud.google.com/trace/docs/ ← App Engine Performance / Latency information

https://cloud.google.com/debugger/docs/

Strategies for dealing with failure

Obviation	Design a system in which a particular type of failure is impossible.
Prevention	Take steps to ensure that a possible failure does not occur.
Detection and Mitigation	Detect a failure before or as it is happening and take steps to reduce or eliminate the effects of it.
Graceful Degradation	Instead of failing completely, handle stress and return to full service when the issue passes.
Repair	Fix the problem. Hopefully, it won't come back, at least not in the same way.
Recover	Allow the problem to occur and get service back as quickly as possible. Measure indicators of recovery and work to improve them.

SRE saying: "Hope is not a strategy."

Evaluating quality control measures

Measures
- Metrics, Reports, SLIs
- Objectives, goals, watermarks, limits

Human processes
- Evaluation
- Decisions
- Actions

Service Level Indicators **(SLIs)**

Google Cloud

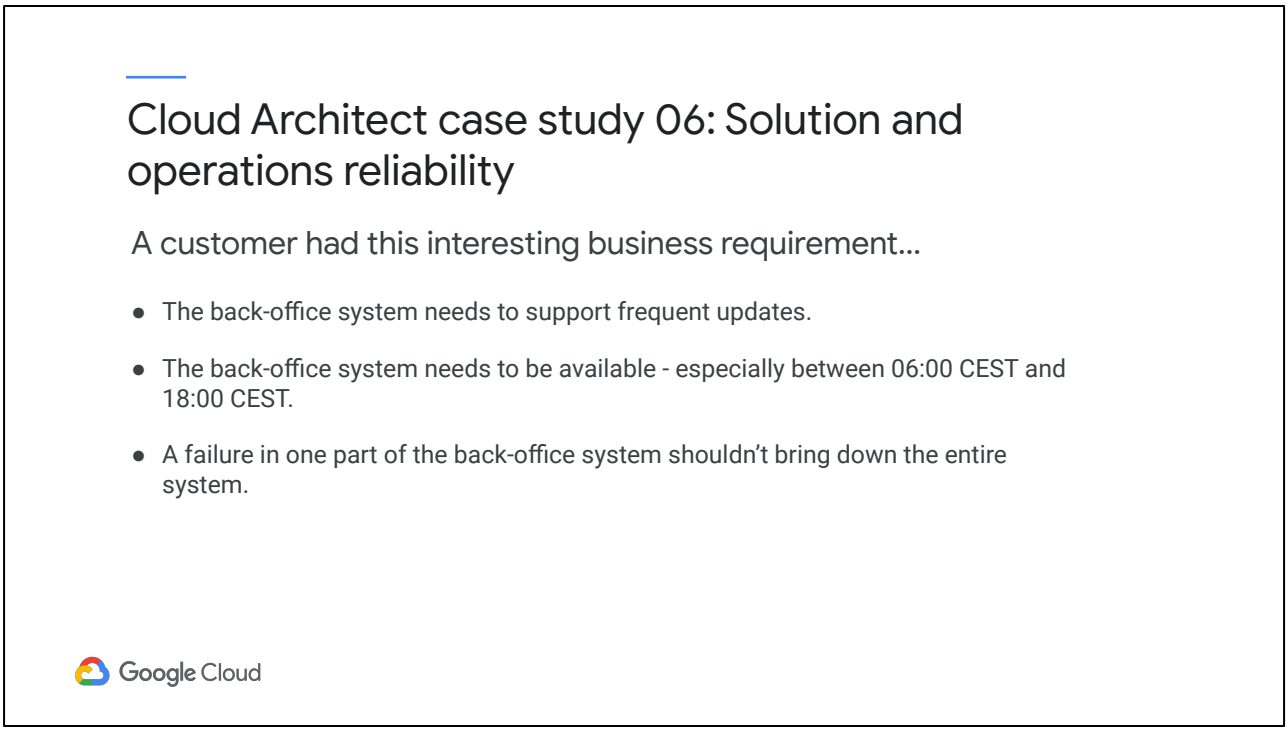
TIP: Qualities are often where our goals start. But figuring out how to measure them quantitatively enables data-driven operations. It can be difficult to figure out exactly what to measure, because sometimes what is easily measured is not a good indicator of customer interests.

Example: CPU utilization may or may not indicate user satisfaction. Round-trip delay or frequency of request errors might be a better measure of the user's experience.

What metrics are you using? Can you define metrics that relate directly to user experience and service objectives?

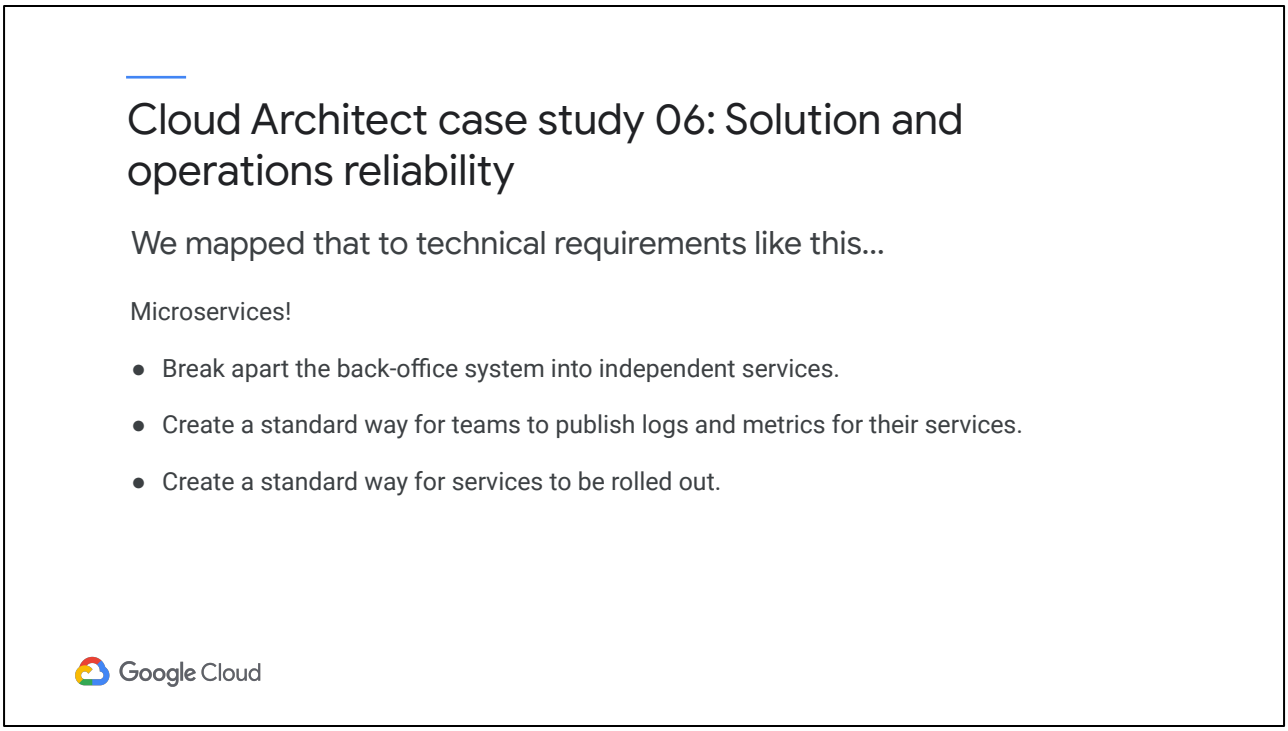
What are the watermarks or alert levels at which human processes are engaged? How are you setting those values? When do they need to be revisited and updated? How do you know they are related to important events?

What are the people supposed to do? What decisions or actions are they supposed to make/take? Are these documented?



A customer had this interesting business requirement...

- The back-office system needs to support frequent updates
- The back-office system needs to be available - especially between 06:00 CEST and 18:00 CEST
- A failure in one part of the back-office system shouldn't bring down the entire system
- Customer wants to re-architect system. Does not want to bring down the entire system when doing an update.

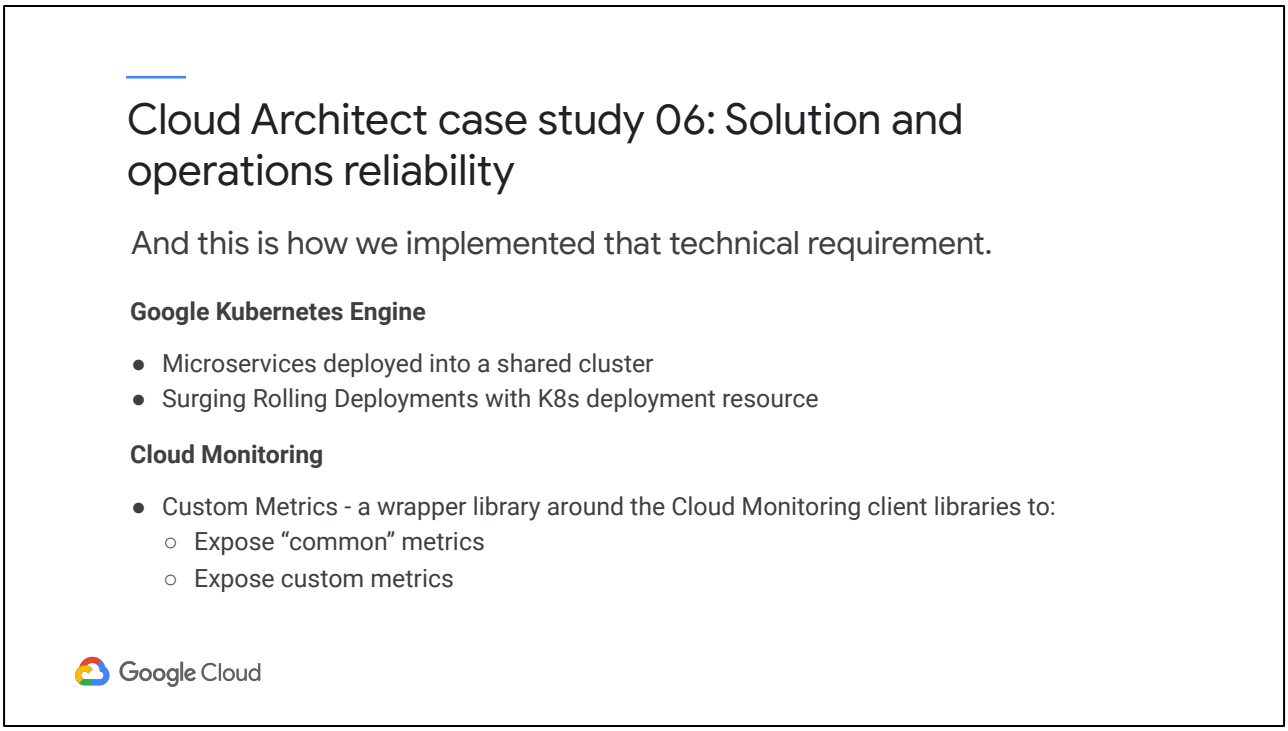


We mapped that to technical requirements like this…

Microservices!

- Break apart the back-office system into independent services
- Create a standard way for teams to publish logs and metrics for their services
- Create a standard way for services to be rolled out

This was a natural fit for microservices. They knew that when they told development groups that they would be developing their own microservices, that they needed standards for reliability and scalability, and they want common ways to monitor the applications.



And this is how we implemented that technical requirement.

Google Kubernetes Engine

- Microservices deployed into a shared cluster
- Surging Rolling Deployments with K8s deployment resource

Cloud Monitoring

- Custom Metrics - a wrapper library around the Cloud Monitoring client libraries to:
- Expose “common” metrics
- Expose custom metrics

Solution was to use Cloud Monitoring. Exposing the metrics could be done through dashboards. Exposed metrics through prometheus standard, scraped from APIs, and sent to Cloud Monitoring where it could be exposed through dashboards.

They used custom metrics in Cloud Monitoring, so they were able to monitor and scale their microservices based on those metrics.

Scenario #1

Question

A microservice has intermittent problems that bursts logs. How can you trap it for live debugging?

A. Log into machine with microservice and wait for the log messages.

B. Look for error in Error Reporting dashboard.

C. Configure microservice to send traces to Cloud Trace.

D. Set a log metric in Cloud Monitoring, alert on it past a threshold.

Origin: CAPE

Scenario #1

Answer

A microservice has intermittent problems that bursts logs. How can you trap it for live debugging?

A. Log into machine with microservice and wait for the log messages.

B. Look for error in Error Reporting dashboard.

C. Configure microservice to send traces to Cloud Trace.

D. Set a log metric in Cloud Monitoring, alert on it past a threshold.

Scenario #1

Rationale

D - A Cloud Monitoring metric can identify a burst of log lines. You can set an alert. Then connect to the machine while the problem is happening.

A - Chances of catching it on one machine is low.

B - Error reporting won't necessarily catch the log lines unless they are stack traces in the proper format. Additionally just because there is a pattern doesn't mean you will know exactly when and where to log in to debug.

C - Trace may tell you where time is being spent but wont let you hone in on the exact host that the problem is occurring on because you generally only send samples of traces. There is also no alerting on traces to notify exactly when the problem is happening.

https://cloud.google.com/logging/docs/
https://cloud.google.com/monitoring/alerts/

Scenario #2

Question

Last week a region had a 1% failure rate in web tier VMs? How should you respond?

A. Monitor the application for a 5% failure rate.

B. Duplicate the application on prem to compensate for failures in the cloud.

C. Perform a root cause analysis, reviewing cloud provider and deployment details to prevent similar future failures.

D. Halt all development until the application issue can be found and fixed.

Origin: Experience

Scenario #2

Answer

Last week a region had a 1% failure rate in web tier VMs? How should you respond?

A. Monitor the application for a 5% failure rate.

B. Duplicate the application on prem to compensate for failures in the cloud.

C. Perform a root cause analysis, reviewing cloud provider and deployment details to prevent similar future failures.

D. Halt all development until the application issue can be found and fixed.

Scenario #2

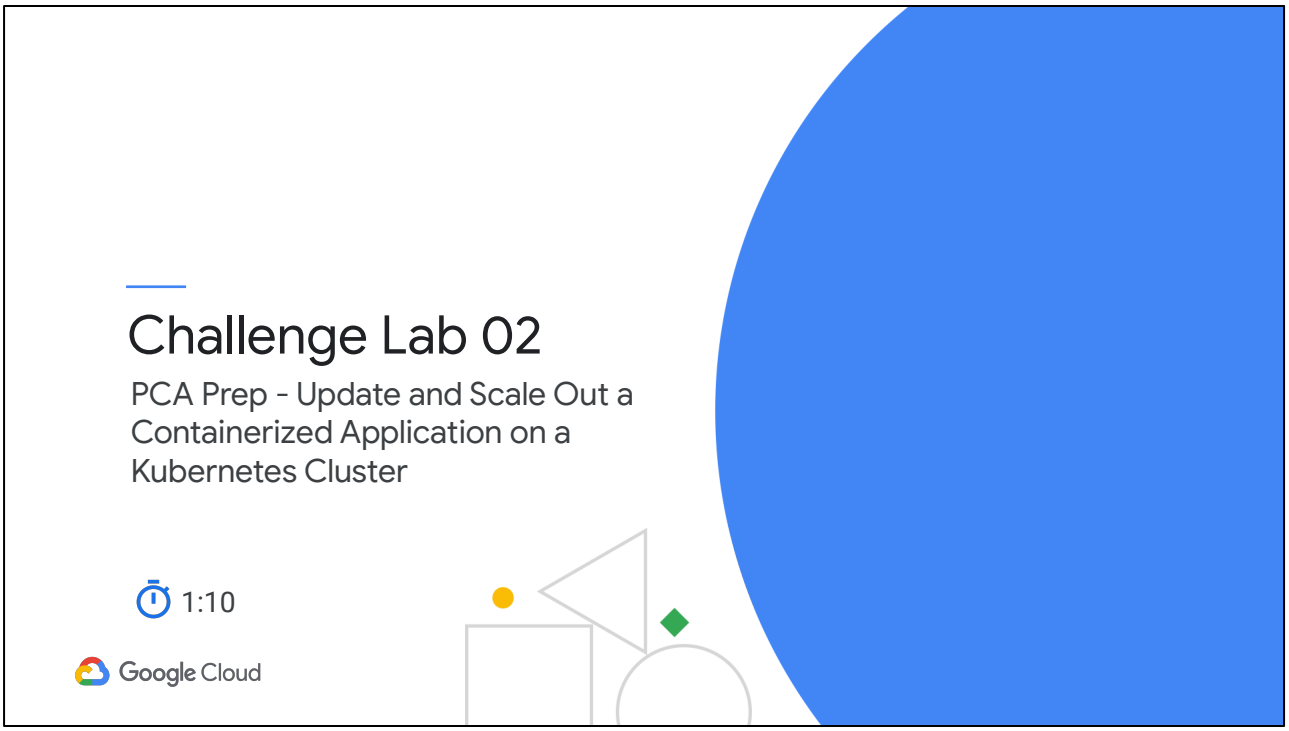
Rationale

C - Perform root cause analysis, because you don't know from the information given whether the issue had to do with the Cloud Provider or was in the application or something to do with the interface between the application and cloud resources. The goal of identifying root cause is to prevent future failures, that might include changing procedures.

A - Raising the threshold doesn't help identify the underlying issue.

B - The assumption is that the cloud is unreliable and on prem is more reliable, so it needs to act as a backup. That's a lot of work that might not be needed and still doesn't find the cause.

D - The assumption is that the application is the problem. But a 1% error could be within SLA for some services. It might not be the application at all. It could be an one-time issue. The information doesn't tell us if this is a recurring problem.



A Challenge Lab has minimal instructions. It explains the circumstance and the expected results -- you have to figure out how to implement them.

This is a timed lab.
The lab will expire after 1 hour and 10 minutes.
The lab *can* be completed in 50 minutes.

A Challenge Lab has minimal instructions. It explains the circumstance and the expected results -- you have to figure out how to implement them.

This is a timed lab.
The lab will expire after 1 hour and 30 minutes.
The lab *can* be completed in 1 hour and 15 minutes.


Google Cloud