Google Cloud

Security

Stephanie Wong
Developer Advocate, Google Cloud

In this module, we cover security.

Google has been operating securely in the cloud for 20 years. There is a strong belief that security empowers innovation. The approach of the cloud architect should be that security should be put first; everything else will follow from this.

Learning objectives

- Design secure systems using best practices like separation of concerns, principle of least privilege, and regular audits.

- Leverage Google's Security Command Center to help identify vulnerabilities.

- Simplify cloud governance using organization policies and folders.

- Authenticate and authorize users with IAM roles, Identity-Aware Proxy, and Identity Platform.

- Manage the access and authorization of resources by machines and processes using service accounts.

- Secure networks with private IPs, firewalls, and Google Cloud private access.

- Mitigate DDoS attacks by leveraging Cloud DNS and Google Cloud Armor.

In this module, you will learn how to design secure systems using industry best practices. For example, when designing a secure system, you will learn how to apply the principle of least privilege and the practice of separation of concerns. Performing regular audits is also a key part of running a secure system.

Leveraging Google's Security Command Center can help to identify vulnerabilities as part of your process. Also,when securing systems, governance is a major consideration. You will learn how organization policies and folders can simplify governance.

Preventing access to unwanted visitors is always a challenge. Authentication and authorization is one approach and can be implemented in many different ways. You will learn how best to use IAM roles, Identity-Aware Proxy, and Identity Platform.

You'll also learn to manage the access and authorization of resources by machines and processes using service accounts. At a lower level of access control, you will learn how to secure network access using private IPs, firewalls, and Google Cloud private access.

Last but not least, you'll learn how to mitigate DDoS attacks by leveraging Cloud DNS and Google Cloud Armor. As you see, there is a lot to security. Let's get started!.
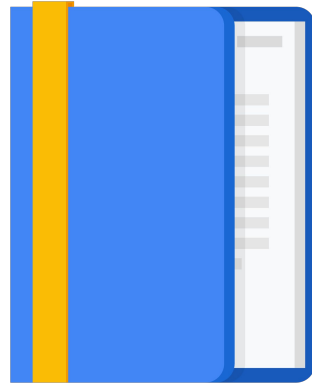
Agenda

Let's begin by talking about some security concepts and introducing some of the best practices for security design.

Google Cloud security is a shared responsibility between you and Google

Transparency

- The client is responsible for certain actions, and Google is responsible for others.
- Google Cloud provides the tools and access to monitor your service.
- Google Cloud provides the controls and features needed to leverage platform security.

Separation of duties

- What is provided by the platform?
- What are you responsible for?

When you move an application to Google Cloud, Google handles many of the lower layers of the overall security stack. Because of its scale, Google can deliver a higher level of security at these layers than most of its customers could afford to do on their own. This does not mean that Google is responsible for all the security aspects.

Google Cloud security is a shared responsibility between you and Google, so it is important that there is a clear separation of duties and there is no ambiguity between what is provided by the platform and what you are responsible for. For this there needs to be transparency. There are certain actions you as a client are responsible for and some that Google is responsible for. Google Cloud provides the controls and features required to leverage the platform together with the tools to monitor your services.

Security is implemented in layers

- Google Cloud provides tools that, when properly configured, enable a secure environment.
- You can also integrate third-party tools for enhanced security.
- There are tools for monitoring and auditing your networks and resources.

Usage
Operations
Deployment
Application
Storage
OS + IPC
Boot
Hardware
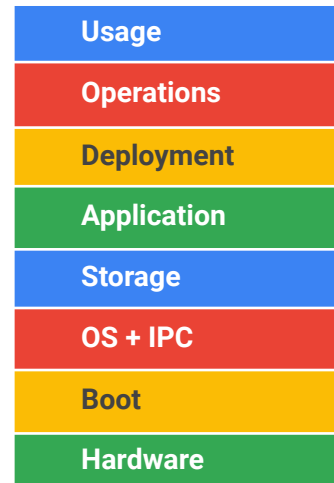
Google implements security in layers. At the base is custom-built hardware and servers that are loaded using a verified boot loading system.

All the way through the stack, security is at the forefront. When you take your part in security, for example, establishing firewall rules or configuring IAM, as long as they are configured correctly, you have a safe environment. There are tools Google Cloud provides that can be used for monitoring and auditing your networks, which we will discuss shortly, or you can also install your own tools.

Principle of least privilege

- Users should only be able to do the tasks that are required by their jobs.
- This should also apply to machine instances and run-time processes.

- Use IAM to enforce this principle.
- Identify users with their login.
- Identify machines and code using service accounts.
- Assign IAM roles to users and service accounts to restrict what they can do.

Let's talk about some best practices when implementing security.

The principle of least privilege is the practice of granting a user only the minimal set of permissions required to perform a duty. This should apply to machine instances and processes, as well as users. Google Cloud provides IAM to help apply this principle. You can use it to identify users with their login or identify machines using service accounts. Roles should be assigned to users and service accounts to restrict what they can do, always following the principle of least privilege.

Separation of duties

Separation of duties means:
- No one person can change or delete data without being detected.
- No one person can steal sensitive data.
- No one person is in charge of designing, implementing, and reporting on sensitive systems.

For example, the people who write the code shouldn't deploy the code, and those who deploy the code shouldn't be able to change it.

- Use multiple projects to separate duties.
- Different people can be given different rights in different projects.
- Use folders to help organize projects.

Separation of duties is another best practice and it has two primary objectives:
1. Prevention of conflict of interest
2. The detection of control failures, for example, security breaches, information theft

From a practical perspective, this means that no one person can change or delete data without being detected. No one person can steal sensitive data, and no single person is in charge of designing, implementing, and reporting on sensitive systems.

For example, a developer who writes the code should not be responsible for deploying that code, and anybody that has the permission to deploy should not be able to change the code. One approach to achieve this separation of duties in Google Cloud is to use multiple projects to separate duties. Different people can be given suitable rights to different projects, with these permissions following the principle of separation of duties. Folders are especially useful for organizing multiple projects.

Regularly audit the Google Cloud logs to discover attacks

All Google Cloud services write to audit logs:

- Admin logs
- Data access logs
- VPC Flow logs
- Firewall logs
- System logs

It is also vital to audit Google Cloud logs to discover attacks and potential security breaches. All Google Cloud services write to audit logs so there is a rich source of information available. These logs include admin, data access, VPC flow, firewall, and system logs, so an in-depth view of activity is provided for audit.

Google Cloud meets many third-party and government compliance standards worldwide

- Google Cloud has been certified as secure, but that does not mean that your application is certified.
- Don't worry about getting Google Cloud tools and services certified; only worry about what you build on top of Google Cloud.

ISO/IEC 27001

HIPAA

FedRAMP

SOC 1

Now, moving to the cloud often requires maintaining compliance with regulatory requirements or guidelines. Google Cloud meets many third-party and government compliance standards worldwide. While Google Cloud has been certified as secure, for example to ISO/IEC 27001, HIPAA, and SOC 1, that does not mean your application running on Google Cloud is certified. Your concern should always be on what you build.

Security Command Center provides access to organizational and project security configuration
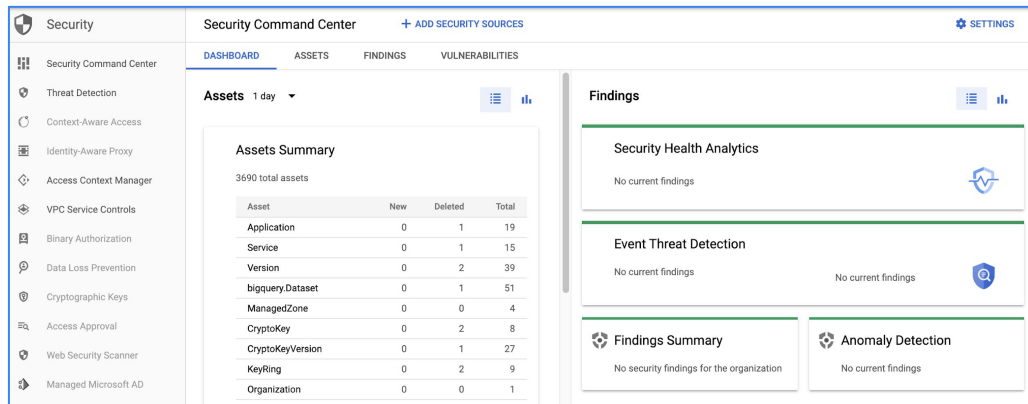
Google Cloud also offers the Security Command Center, which provides access to organizational and project security configuration. As you can see in this screenshot, the Security Command Center provides a dashboard that reports security health analysis, threat detections, anomaly detection, and a summary report. Once a threat is detected, a set of actionable recommendations is provided.
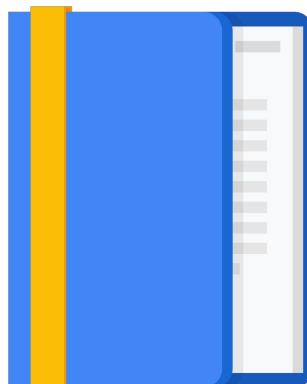
Agenda

Security Concepts

Securing People

Securing Machine Access

Network Security

Encryption

Let's move on to talk about securing people.

To grant people access to your projects, add them as members and assign them one or more roles

- Members are identified by their login.
- Add members to groups for easier management.

- Roles are simply a list of permissions.
- Use the Console to easily see what permissions are granted to roles.

BigQuery ✕ Filter table

Type	Title	Used in	Status	
☐ ⊕	BigQuery Admin	BigQuery	Enabled	⋮
☐ ⊕	BigQuery Connection Admin	BigQuery	Enabled	⋮
☐ ⊕	BigQuery Connection User	BigQuery	Enabled	⋮
☐ ⊕	BigQuery Data Editor	BigQuery	Enabled	⋮
☐ ⊕	BigQuery Data Owner	BigQuery	Enabled	⋮
☐ ⊕	BigQuery Data Viewer	BigQuery	Enabled	⋮
☐ ⊕	BigQuery Job User	BigQuery	Enabled	⋮
☐ ⊕	BigQuery Metadata Viewer	BigQuery	Enabled	⋮
☐ ⊕	BigQuery Read Session User	BigQuery	Enabled	⋮
☐ ⊕	BigQuery User	BigQuery	Enabled	⋮
☐ ⊕	Cloud Asset Owner	Cloud Asset	Enabled	⋮

BigQuery User + EDIT ROLE ⧉ CREATE FROM ROLE

ID	roles/bigquery.user
Role launch stage	General Availability

Description

Access to run queries and create datasets

15 assigned permissions

bigquery.config.get
bigquery.datasets.create
bigquery.datasets.get
bigquery.datasets.getIamPolicy
bigquery.jobs.create
bigquery.jobs.list
bigquery.models.list
bigquery.readsessions.create
bigquery.routines.list
bigquery.savedqueries.get
bigquery.savedqueries.list
bigquery.tables.list
bigquery.transfers.get
resourcemanager.projects.get
resourcemanager.projects.list

When granting people access to your projects, you should add them as members and assign them one or more roles. Roles are simply a list of permissions. To see what permissions are granted to roles, use the Cloud Console as shown on the right. Here you can see the role bigquery_user and the associated 15 permissions the role has assigned to it. You can assign these predefined roles to members or customize your own roles.

Now, any member added to your project will be identified by their login. For simplifying management of members and their permissions, I recommend that you create groups. That way you just need to add members to a group, and new members automatically acquire the permissions of the group. The same applies for removing members from a group, which also removes the permissions of that group.

Use Identity and Access Management policies to secure environments and manage resources

Grant roles to Google groups rather than individuals

- Groups can be more granular than job roles.
- Use multiple groups for better control (such as *view only*).

Roles

- Prefer pre-defined roles over custom roles.
- Grant roles at the smallest scope needed (least privilege).
- Limit use of "owner" and "editor" roles.
- Consider hierarchy inheritance when assigning roles.

Policy Inheritance

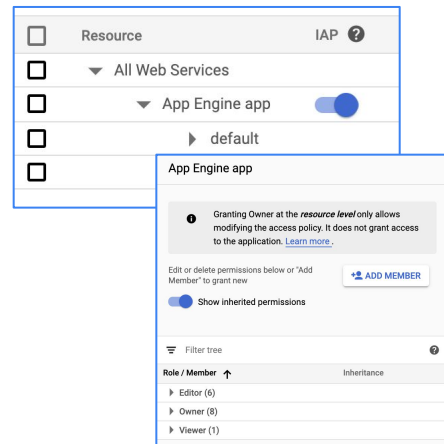Organization

Folders

Projects
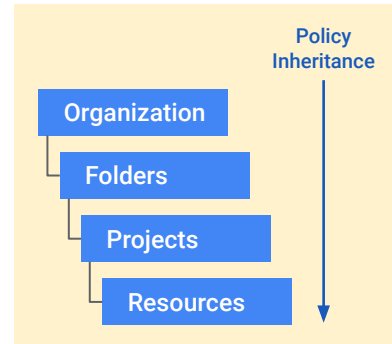
Resources

I also recommend using organizational policies and folders to simplify securing your environments and managing your resources. Organizational policies apply to all resources underneath an organization, and IAM policies are also inherited top to bottom, as shown on the right. Folders inherit policies of the organization, projects inherit policies of the folders, and so on.

I already mentioned that roles should be granted to groups, not individuals, because it simplifies management. Make sure to define groups carefully and make them more granular than job roles. It's better to use multiple groups for better control.

When it comes to roles, it is better to use predefined roles over custom roles. Google has defined the roles for a reason, and it should be an exceptional case that a role does not fit your use case. When granting roles, remember the principle of least privilege: always grant the smallest scope required. Owner and editor roles should be limited; these are not or should not be required by the majority of users.

Identity-Aware Proxy simplifies authorization to Google Cloud applications and VMs

- Works with applications deployed behind the HTTP(S) load balancer in Compute Engine, GKE, or App Engine.
- When configured, it forces users to log in.
- Admins control who can access to app.
- Allows employees to securely access web-based applications without the need for a VPN.
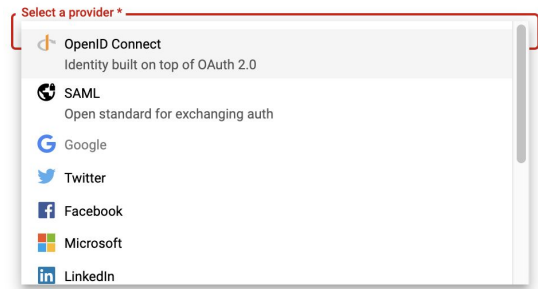
	Resource	IAP
☐	▼ All Web Services	
☐	▼ App Engine app	⬤
☐	▶ default	

App Engine app

ⓘ Granting Owner at the *resource level* only allows modifying the access policy. It does not grant access to the application. Learn more.

Edit or delete permissions below or "Add Member" to grant new

⬤ Show inherited permissions

≡ Filter tree   ❓

Role / Member ↑   Inheritance
▶ Editor (6)
▶ Owner (8)
▶ Viewer (1)

I also recommend leveraging Cloud Identity-Aware Proxy, or Cloud IAP. Cloud IAP provides managed access to applications running in App Engine standard environment, App Engine flexible environment, Compute Engine, and GKE. It allows employees to securely access web-based applications deployed on Google Cloud without requiring a VPN. Administrators control who has access, and users are required to log on to gain access to the applications. The screenshots on the right show Cloud IAP being enabled on an App Engine application and the dialog for adding new members or permissions.

Identity Platform provides authentication as a service

- Provides federated login that integrates with many common providers.
- Use it to provide sign-up and sign-in for your end users' applications.

Sign-in method

Select and configure an identity provider.

Select a provider *

◆ OpenID Connect	Identity built on top of OAuth 2.0
🌐 SAML	Open standard for exchanging auth
G Google	
🐦 Twitter	
f Facebook	
⊞ Microsoft	
in LinkedIn	

Google Cloud also offers Identity Platform as a customer identity and access management (CIAM) platform for adding identity and access management to applications. In other words, Identify Platform provides sign-up and sign-in for end user applications.

Now, you need to select a service provider to use Identity Platform. A broad range of protocol support is available including SAML, OpenID, Email and password, Phone, Social, and Apple. This graphic shows a part of the configuration with a list of potential providers.

Agenda

Security Concepts

Securing People

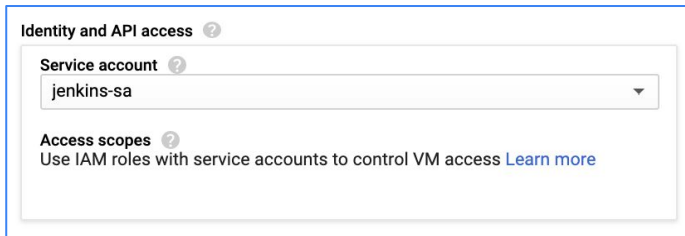Securing Machine Access

Network Security

Encryption
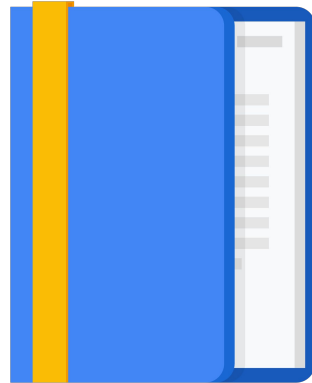
We just talked about assigning roles to members. We focused on users and Google Groups, but there's another kind of member that helps secure machine access.

Service accounts can be used for machine or application identities

- Create a service account and grant it one or more roles.
- Can assign that service account to VMs or GKE node pools.
- Those machines run with only the rights granted by the roles.

Identity and API access ⓘ

Service account ⓘ

jenkins-sa ▾

Access scopes ⓘ
Use IAM roles with service accounts to control VM access Learn more

A service account is a special kind of account used by an application, a virtual machine instance, or a GKE node pool. Applications or services use service accounts to make authorized API calls. The service account is the identity of the service and defines permissions which control the resources the service can access.

A service account is both an identity and a resource. A service account is used as an identity for your application or service to authenticate, for example, a Compute Engine VM running as a service account. To give the VM access to the necessary resources, you need to grant the relevant IAM roles to the service account. At the same time, you need to control who can create VMs with the service account so random VMs cannot assume the identity. Here, the service account is the resource to be permissioned. You assign the ServiceAccountUser role to the users you trust to use the service account.

Service accounts can be used for machine or application identities

- Generate and download a key when creating a service account.

- This key can be used for authentication.

- Key is downloaded as JSON.

- User-managed keys are extremely powerful credentials.

- Store the key safely.

Create key (optional)

Download a file that contains the private key. Store the file securely because this key can't be recovered if lost. However, if you are unsure why you need a key, skip this step for now.

+ CREATE KEY

```
{
  "type": "service_account",
  "project_id": "project-id",
  "private_key_id": "48e95bf20887235536f772dcf25d47b89f8cf49",
  "private_key": "-----BEGIN PRIVATE KEY-----\nMIIEvAIBADANB
  "client_email": "my-service-account@project-id.iam.gservic
  "client_id": "113723034034071973858",
  "auth_uri": "https://accounts.google.com/o/oauth2/auth",
  "token_uri": "https://oauth2.googleapis.com/token",
  "auth_provider_x509_cert_url": "https://www.googleapis.com
  "client_x509_cert_url": "https://www.googleapis.com/robot/
}
```

Each service account is associated with public/private RSA key-pairs that are used to authenticate to Google. These keys can be Google-managed or user-managed.

Google-managed keys, both the public and private keys, are stored by Google, and they are rotated regularly. The maximum usage period is two weeks. For user-managed keys, the developer owns both public and private keys. They can be used from outside Google Cloud. User-managed keys can be managed by the IAM API, gcloud command-line tool, or the service accounts page in the Cloud Console. It is possible to create up to 10 key-pairs per service account to support key rotation.

User-managed keys are extremely powerful credentials, and they can represent a security risk if they are not managed correctly. You can limit their use by applying the constraints/iam.disableServiceAccountKeyCreation Organization Policy Constraint to projects, folders, or even your entire organization. After applying the constraint, you can enable user-managed keys in well-controlled locations to minimize the potential risk caused by unmanaged keys. Consider using Cloud Key Management Service (Cloud KMS) to help securely manage your keys.

The slide shows the generation of a key using the Cloud Console. The private key can be seen in the screen shot. It is your responsibility for storing the private key securely.

Can use service account keys to configure the CLI

- Allows you to grant controlled Google Cloud access to developers without giving them access to the Cloud Console.

- Also useful for automation when configuring VMs to run CI/CD pipelines.

- Use: `gcloud auth activate-service-account --key-file=[PATH TO KEY FILE]`

For developers to gain controlled access to resources without acquiring access to the Cloud Console, it is possible to configure the gcloud command-line utility to use service account credentials to make requests. The command on this slide, gcloud auth activate-service-account, serves the same purpose as gcloud auth login but uses the service account instead of user credentials.

The key file contains the private key in JSON format, which I just discussed.

Agenda
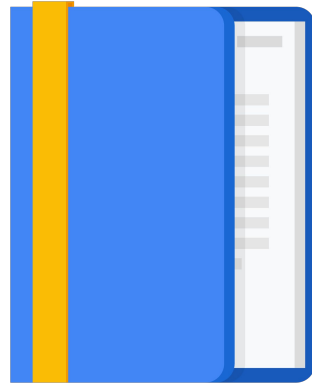
In a previous module we talked about networks but didn't get into a lot of Network Security concepts. Let's do that now.

Remove external IPs to prevent access to machines outside their network

- Use a bastion host to provide access to private machines.
- Can also SSH into internal machines using Identity-Aware Proxy from the console and CLI.
- Use Cloud NAT to provide egress to the internet from internal machines.

All internet traffic should terminate at a load balancer, third-party firewall (proxy or WAF), API Gateway, or IAP. That way, internal services cannot be launched and get public IP addresses.

Client

Google Cloud

Firewall Rules

Bastion Host
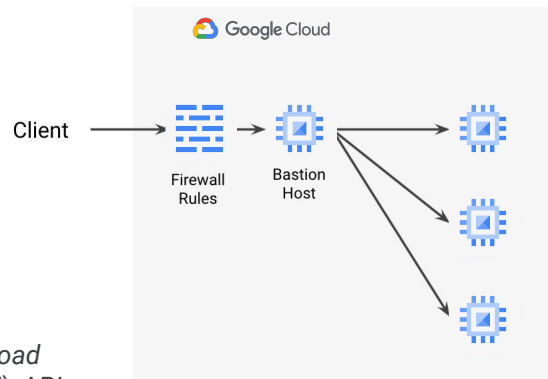
First, I recommend removing external IPs to prevent access to machines from outside their network whenever possible.

Several options are available for securely communicating with VMs that do not have public IP addresses. These services do not have a public IP address because they are deployed to be consumed by other instances in the project or maybe through Dedicated Interconnect options. However, for those instances that do not have an external IP address, it can be a requirement to gain external access, for instance for updates or patches to be applied. The options for accessing the VMs include a bastion host for external access to private machines, Identity-Aware Proxy to enable SSH access, or Cloud NAT to provide egress to the internet for internal machines. VM instances that only have internal IP addresses can also use Private Google Access.

The diagram on the right shows an external client accessing Compute Engine resources via a bastion host. The host is behind a firewall where access can be filtered. Whichever method you choose, all internet traffic should terminate at a load balancer, third-party firewall, or API gateway, or through IAP. That way internal services cannot be launched and get public IP addresses.

# Private access allows access to Google Cloud services using an internal address

- Enabled when creating subnets.
- Allows access to Google Cloud services from VMs that only have internal IPs.
  - For example, a machine with only an internal IP would be able to reach a Cloud Storage bucket.

130.211.160.207

Google Cloud

Cloud Storage

Public IP

Private IP

Bucket

SUBNETWORK A

Compute Engine

SUBNETWORK B

Compute Engine

192.168.1.5

```
gcloud compute networks subnets
update subnet-b \
    --enable-private-ip-google-access
```
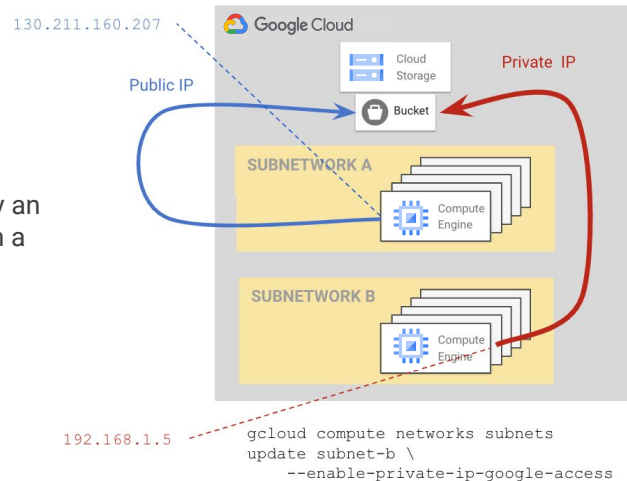
Now, VM instances that only have internal IP addresses can use Private Google Access to access Google services that have external IP addresses. The diagram on the right shows a Compute Engine instance accessing a Cloud Storage bucket using its internal IP address. Private Google Access must be enabled when creating the subnet. You can achieve this either with the gcloud command shown here or through the Cloud Console.

Configure firewall rules to allow access to VMs

- By default, ingress on all ports is denied.
- Add firewall rules to control which clients have access to which VMs on which ports.
- Application level security is the responsibility of the customer.
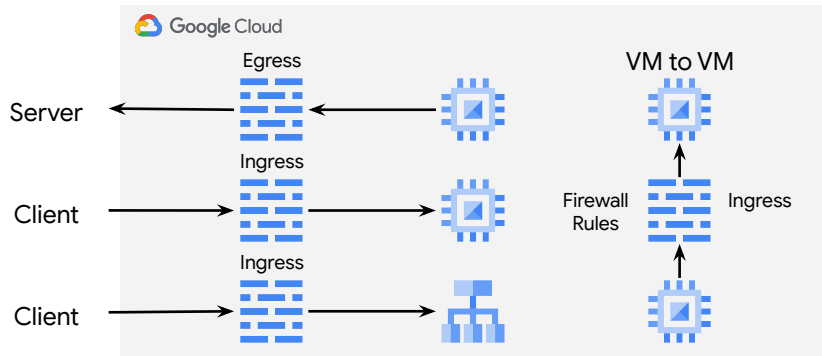
Regardless of whether your VM instances have public IP addresses, you should always configure firewall rules to control access.

By default, ingress on all ports is denied and all egress is allowed. It's your responsibility to define separate rules to allow or deny access to specific instances for specific IP ranges, protocols, and ports.

This graphic shows some scenarios where firewall rules can be configured. Egress from Compute Engine to external servers is the first scenario. For ingress, firewall rules should be configured if direct access to an instance is being provided or if via a load balancer. The right-hand graphic shows the scenario of VM instance-to-instance communication. Firewall rules should be considered here to control access also. Remember, you're still responsible for application-level security.

Control access to APIs using Cloud Endpoints
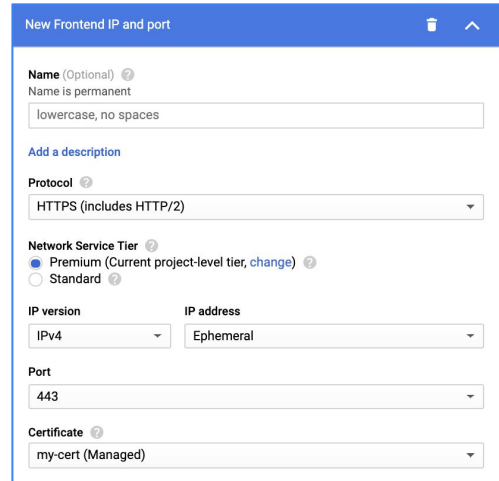
- Protect and monitor your public APIs.
- Control who has access to your API.
- Validate every call with JSON Web Tokens and Google API keys.
- Integrates with Identity Platform.

If you need to manage APIs, you can use Cloud Endpoints. Endpoints is an API management gateway that helps you develop, deploy, and manage APIs on any Google Cloud backend. It provides functionality to protect and monitor your public APIs, control who has access—using, for example, Auth0—and validate every call with a JSON Web Token signed with a service account private key. Cloud Endpoints also integrates with Identity Platform for authentication.

Restrict access to your services to TLS only

- All Google Cloud service endpoints use HTTPS.
- It's up to you to configure your service endpoints.
- In the load balancer setup, only create a secure frontend.


All Google Cloud service endpoints use HTTPS. I recommend that you use TLS for your service endpoints, and it is your responsibility to configure your service endpoints for TLS. When configuring load balancers, only ever create secure frontends. This dialog shows the configuration of a frontend and the protocol selected is HTTPS, with the certificate also being selected.

Leverage Google Cloud network services for DDoS protection
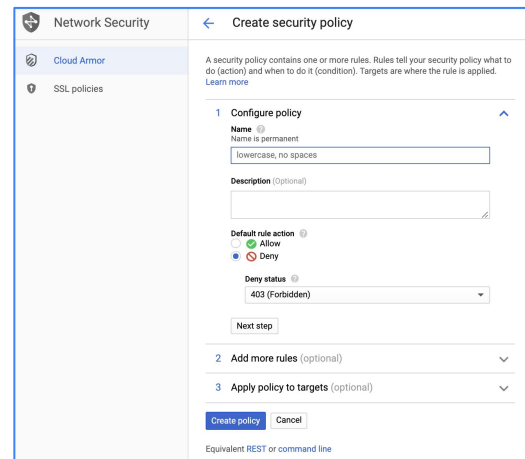
- Global load balancers detect attacks and drop them.
- Enabling the CDN will protect backend resources.
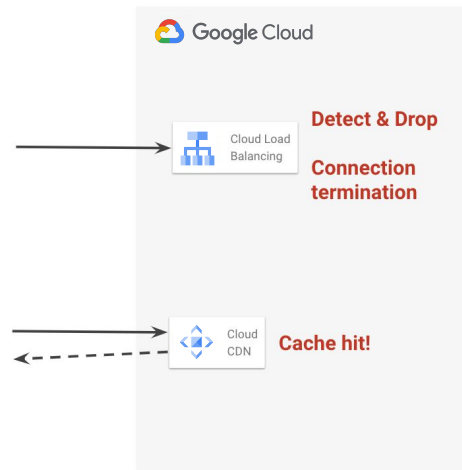
Google provides infrastructure DDoS support through global load balancers at level 3 and level 4 traffic. If you have enabled CDN, this will also protect backend resources because a DDos results in a cache hit instead of hitting your resources, as shown on the right.

Use Google Cloud Armor to create network security policies

- Can allow or deny access to your Google Cloud resources using IP addresses or ranges.

- Create allow lists to allow known addresses.

- Create deny lists to block known attackers.

We already mentioned Google Cloud Armor in the Networking module.

For additional features over the built in DDoS protection, you can use Google Cloud Armour to create network security policies. For example, you can create allow lists that allow known/required addresses through, and deny lists to block known attackers.

This dialog shows a typical security policy configuration where you begin by selecting it as an allow list or deny list with allow or deny for the rule. If it's a deny, the appropriate action in this example should be a 403 error.

Google Cloud Armor supports layer 7 web application firewall (WAF) rules

- Predefined rules for preventing common attacks like SQL injection and cross-site scripting
- Flexible rules language allows you to allow or deny traffic using request headers, geographic location, ip addresses, cookies, etc.
- Examples:

```
inIpRange(origin.ip, '9.9.9.0/24')
request.headers['cookie'].contains('80=BLAH')
origin.region_code == 'AU'
inIpRange(origin.ip, '1.2.3.4/32') &&
request.headers['user-agent'].contains('WordPress')
evaluatePreconfiguredExpr('xss-canary')
```

In addition to Layer 3 and Layer 4 security, Google Cloud Armor supports Layer 7 application rules. For example, predefined rules are provided for cross-site scripting (XSS) and SQL injection attacks. Google Cloud Armor provides a rules language for filtering request traffic. As an example, consider the first expression on this slide: inIpRange(origin.ip, '9.9.9.0/24). In this case, the expression returns true if the origin IP in a request is within the 9.9.9.0/24 range.

The second line, request.headers['cookie'].contains('80=BLAH'), returns true if the cookie 80 with value BLAH exists in the request header, and the third line is true if the origin region code is AU. The expressions can be combined logically with logical AND and OR.

The expressions are all assigned to an allow or deny rule that is then applied to incoming traffic.
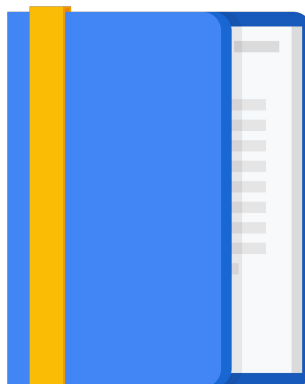
Agenda

Security Concepts

Securing People

Securing Machine Access

Network Security

Encryption

Last but certainly not least, let's go over encryption.

Google Cloud provides server-side encryption of data at rest by default

- Data Encryption Key (DEK) uses AES-256 symmetric key.
- Keys are encrypted by Key Encryption Keys (KEK).
- Google controls root keys in Cloud KMS.
- Keys are automatically periodically rotated.
- On-the-fly decryption by authorized user access with no visible performance impact.
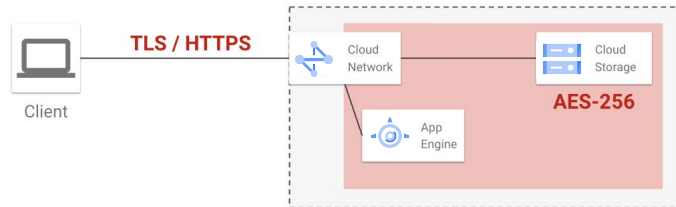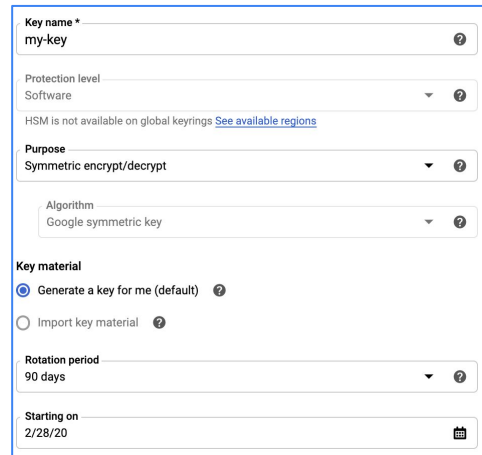

Google Cloud encrypts customer data stored at rest by default, with no additional action required from users. A Data Encryption Key (or DEK) using AES-256 symmetric key is used, and the key itself is encrypted by Google using a Key Encryption Key (KEK). This is so that the DEK can be stored local to the encrypted data for fast decryption with no visible performance impact to the user. To protect the KEKs, they are stored in Cloud KMS. The keys are rotated periodically and automatically for added security.

This diagram shows a simple App Engine application that uses Cloud Storage. The data is encrypted using AES-256 using a DEK and decrypted transparently to the application when the data is read.

For compliance reasons, you may need to manage your own keys

- Customer-managed encryption keys are created in the cloud using Cloud Key Management Service (KMS).
- You create the keys and specify the rotation frequency.
- You can then select your keys when creating storage resources like bucket and disks.

Key name *	
my-key	❓

Protection level	
Software	▼ ❓

HSM is not available on global keyrings See available regions

Purpose	
Symmetric encrypt/decrypt	▼ ❓

Algorithm	
Google symmetric key	▼ ❓

Key material

🔘 Generate a key for me (default) ❓

⚪ Import key material ❓

Rotation period	
90 days	▼ ❓

Starting on	
2/28/20	📅

Now, for compliance reasons, you may need to manage your own encryption keys rather than the automatically generated keys as just discussed.

In this scenario, you can use Cloud Key Management Service (or Cloud KMS) to generate what are known as Customer Managed Encryption Keys (CMEK). These keys are stored in Cloud KMS for direct use by Cloud services. You can manually create the key using a dialog similar to the one shown here and specify the rotation frequency, which defaults to 90 days. The keys you create can then be used when creating storage resources such as disks or buckets.

Customer-supplied encryption keys are created in your environment and provided to Google Cloud

- Use your own keys with Google Cloud services.
- CSEK are supplied by the calling application per-API call.
- Only cached in RAM by Google.
- They decrypt a single payload (or column) or block of returned data.
- Supported by Compute Engine (persistent disks) and Cloud Storage.

When you're required to generate your own encryption key or manage it on-premises, Google Cloud supports Customer Supplied Encryption Keys (CSEK). Those keys are kept on-premises and not in Google Cloud. The keys are provided as part of API service calls, and Google only keeps the key in memory and uses it to decrypt a single payload or block of returned data. Currently, Customer Supplied Encryption Keys can be used with Cloud Storage and Compute Engine.

The Data Loss Prevention API can be used to protect sensitive data by finding it and redacting it

- Scans data in Cloud Storage, BigQuery, or Firestore.
- Can also scan images.
- Detects many different types of sensitive data, including:
  - Emails
  - Credit cards
  - Tax IDs
- You can add your own information types.
- Can delete, mask, tokenize, or just identify the location of the sensitive data.

CREDIT_CARD_NUMBER ⊗   EMAIL_ADDRESS ⊗

GCP_CREDENTIALS ⊗   IMEI_HARDWARE_ID ⊗

IP_PASSPORT ⊗   MAC_ADDRESS ⊗   PASSPORT ⊗

MAC_ADDRESS_LOCAL ⊗   PHONE_NUMBER ⊗

US_BANK_ROUTING_MICR ⊗

US_EMPLOYER_IDENTIFICATION_NUMBER ⊗

US_INDIVIDUAL_TAXPAYER_IDENTIFICATION_NUMBER ⊗

US_SOCIAL_SECURITY_NUMBER ⊗

You should also consider the Data Loss Prevention API to protect sensitive data by finding it and redacting it. Cloud DLP provides fast, scalable classification and redaction for sensitive data elements like credit card numbers, names, social security numbers, US and selected international identifier numbers, phone numbers, and Google Cloud credentials. Cloud DLP classifies this data using more than 90 predefined detectors to identify patterns, formats, and checksums, and even understands contextual clues. Some of these are shown on the right. You can optionally redact data as well, using techniques like masking, secure hashing, tokenization, bucketing, and format-preserving encryption.
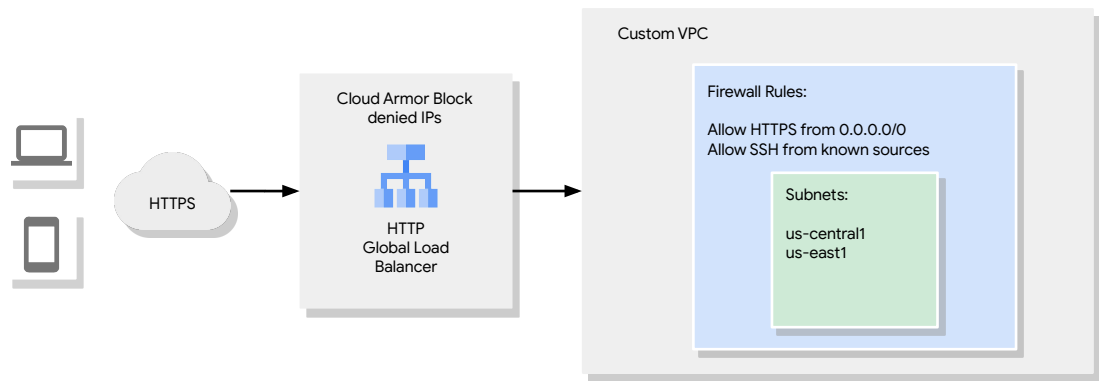
Activity 12: Modeling Secure Google Cloud Services

Refer to your Design and Process Workbook.

● Draw a diagram that depicts your case study security requirements.

In this design activity, you draw a diagram that depicts your case study's security requirements. Let me show you an example of what to draw.

This diagram illustrates a custom VPC network with two subnets in the US. Maybe us-central1 is our primary region and us-east1 is our backup region. The firewall rules allow HTTPS ingress from the internal and SSH from known sources. Otherwise, all other incoming traffic is disabled by the implied deny all ingress firewall rule that every VPC network has.

Because we're allowing HTTPs from anywhere, it's useful to configure Google Cloud Armor on a global HTTP Load Balancer to block any denied IP addresses at the edge of Google Cloud's network. This is a simple design but a great starting point, because it allows us to grow our backends without changing our security design.
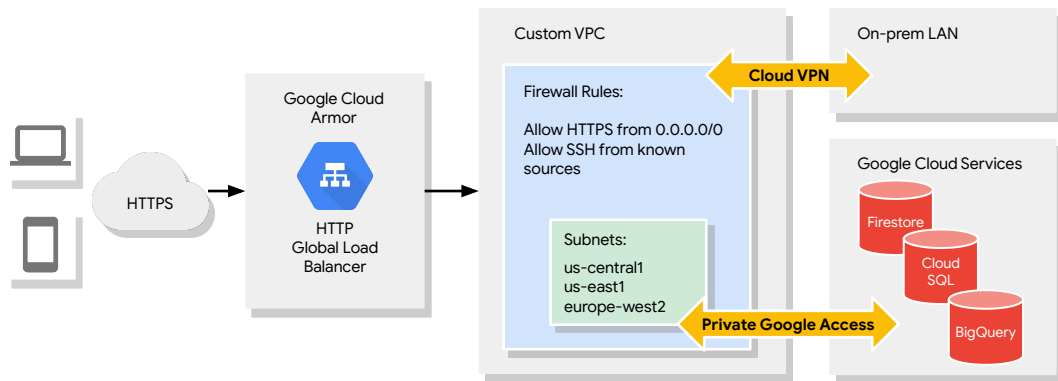
Refer to activity 12 in your workbook to create a similar diagram for your case study.

Review Activity 12: Modeling Secure Google Cloud Services

- Draw a diagram that depicts your case study security requirements.

In this activity, you were asked to draw a diagram depicting the security requirements for your case study.

Custom VPC

Cloud VPN

On-prem LAN

Google Cloud
Armor

Firewall Rules:

Allow HTTPS from 0.0.0.0/0
Allow SSH from known
sources

Google Cloud Services

Firestore

HTTPS

HTTP
Global Load
Balancer

Subnets:

us-central1
us-east1
europe-west2

Cloud
SQL

Private Google Access

BigQuery

Here's the diagram that I drew for our online travel portal, ClickTravel.

This is a similar design to what I showed you earlier. First, I configured Google Cloud Armor on a global HTTP Load Balancer to block any denied IP addresses. My custom VPC network has subnets in us-central1 for my American customers, and a backup subnet in us-east1 and a subnet in europe-west2 for my European customers.

My firewall rules only allow SSH from known sources, and although I allow HTTPS from anywhere, I can always deny IP addresses with Google Cloud Armor at the edge of Google Cloud's network. I also configured Cloud VPN tunnels to securely communicate with my on-premises network for my reporting service.

Now, while my load balancer needs a public IP address, I can secure my backend services by creating them without external IP addresses. In order for those instances to communicate with the Google Cloud database services, I enable Private Google Access. This enables the inventory, orders, and analytics services' traffic to remain private, while reducing my networking costs.

Review

Security

In this module, we covered how to secure our Google Cloud resources. This includes securing both the network and our stored data. We also covered how to secure people using IAM, Cloud Identity, and Identity Aware Proxy, and how we can secure our applications and machines using service accounts.

Remember, security should be put first; everything else will follow from this.