Solution Guide
June 2021

Setting up a FedRAMP Aligned Three-Tier Workload on Google Cloud

DISCLAIMER: This whitepaper applies to Google Cloud products described at cloud.google.com. The content contained herein represents the status quo as of the time it was written. Google's security policies and systems may change going forward, as we continually improve protection for our customers.

Table of Contents

Table of Contents 2

Disclaimers 4

1. Overview 5

1.1 FedRAMP compliance 5

1.2 Solution guide 5

1.3 Cloud Data Protection Toolkit 6

2. FedRAMP compliance in Google Cloud 7

3. Deploying a FedRAMP aligned three-tier workload using Cloud Data Protection Toolkit 7

3.1 FedRAMP aligned three-tier reference architecture 8

3.2 In-scope Product Guidance 10

3.2.1 Cloud Storage 10

3.2.2 BigQuery 12

3.2.3 Cloud SQL 14

3.2.4 Confidential VM 15

3.2.5 Pub/Sub 18

3.2.6 Google Kubernetes Engine 20

3.2.7 Dataflow 22

3.2.8 Cloud load balancing (external) 23

3.2.9 Cloud Armor 25

3.3 Environment setup 27

3.4 Access control 28

3.4.1 Pre-deployment access control 28

3.4.2 Post-deployment access control 29

3.5 Deployment phases 30

3.5.1 Generate Terraform files 31

3.5.2 Deploy Terraform configurations 32

3.6 Pre-deployment setup 34

3.6.1 DevOps project deployments 34

3.6.2 Logging Project (Assured Workloads) deployments 35

3.6.3 Three-tier workload project deployments 44

3.7 Post-deployment verification 58

3.7.1 DevOps project 58

3.7.2 Three-tier workload project 59

3.7.3 Logging project 68

A. Appendix 70

A.1 FedRAMP and Google Cloud 70

A.2 Compliance 71

A.3 Google Cloud shared responsibility model 71

A.4 Customer responsibilities 73

Disclaimers

- This solution guide contains information on Google Cloud products described at cloud.google.com. The content contained herein is correct as of June 2021 and represents the status quo as of the time it was written. Google's security policies and systems may change going forward, as we continually improve protection for our customers.

- This solution guide and accompanying templates, including the Google Cloud Data Protection Toolkit, provide a reference architecture, leading practices, and recommendations for Google Cloud customers. This guide does not constitute any legal advice on the administrative, technical, and physical safeguards required to implement Federal Risk and Authorization Program (FedRAMP) requirements.

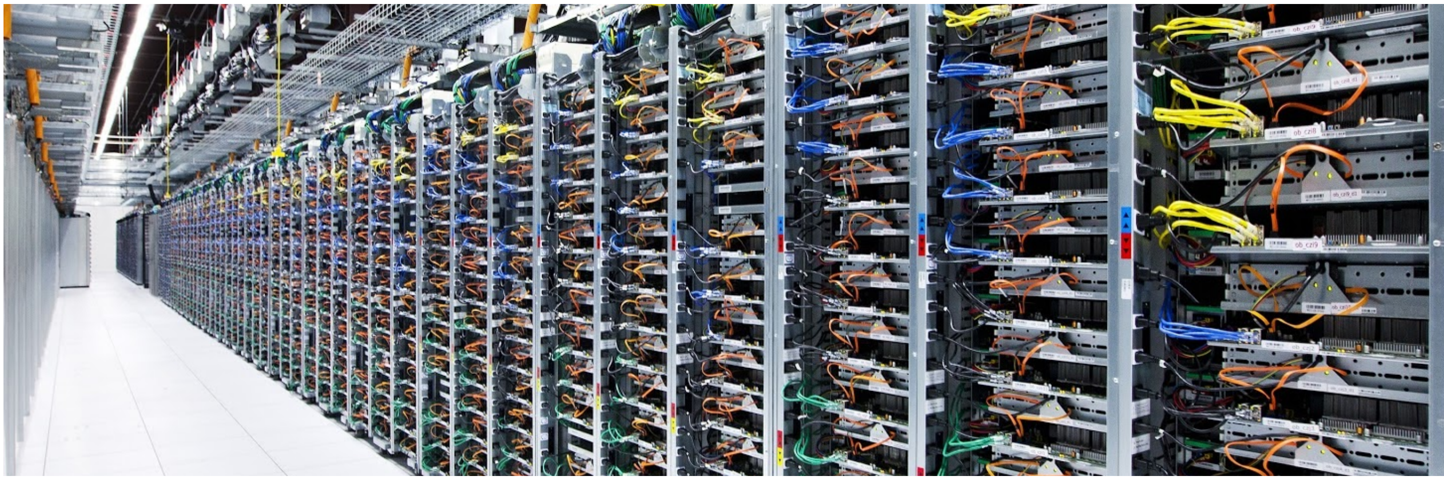
- The customer is responsible for managing data and applications, including configuration and maintenance of services hosted in Google Cloud. These responsibilities are further described in section A.4, Customer Responsibilities.

- This guide is limited to providing security guidance for protecting and monitoring data within the in-scope resources defined in Deploying a FedRAMP Aligned 3-tier Workload using Cloud Data Protection Toolkit.

- Implementation of the solution guide or reference architecture does not automatically cover any data assets that are stored or processed by other Cloud Storage services. Similar protective measures must be applied to all other data stored across the environment.

- The implementation of this solution may vary in customer environments based on the choice of products and configuration options.

- This guide is designed to help you get started. You will need to customize and verify your own settings and controls to support your specific FedRAMP compliance need



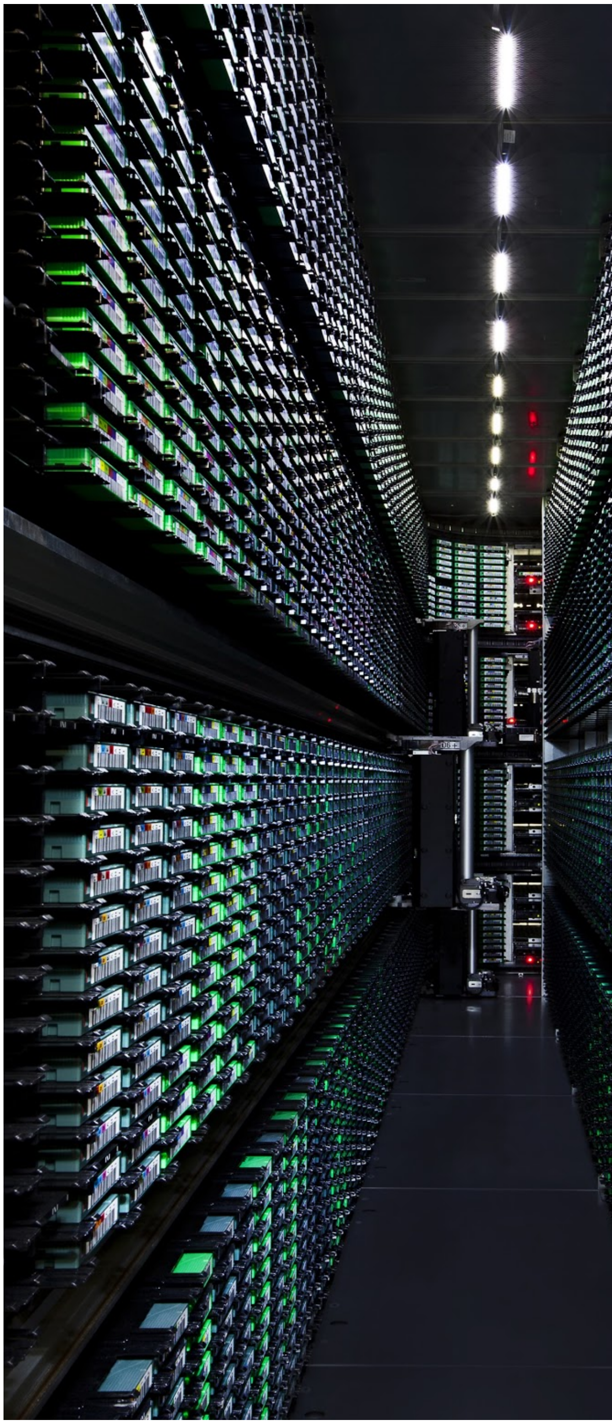
1. Overview

1.1 FedRAMP compliance

FedRAMP is a U.S. government-wide program that provides a standardized approach to security assessment, authorization, and continuous monitoring for cloud products and services offered to US federal government agencies. Most federal agency cloud deployments and service models, other than certain on-premises private clouds, must meet FedRAMP requirements at the appropriate (Low, Moderate, or High)
risk impact level.

The FedRAMP High authorization obtained by Google Cloud enables government agencies that process high impact workloads to adopt technology at a much higher velocity and at the same scale as commercial customers.

For details on Google Cloud services covered by FedRAMP, refer to the FedRAMP Marketplace.



1.2 Solution guide

This solution guide covers the process and the guidelines to deploy a FedRAMP-aligned three-tier workload on Google Cloud using Assured Workloads. Assured Workloads allows Google Cloud customers to quickly and easily create controlled environments where US data location and personnel access controls are enforced in any of our US cloud regions. Refer to the Assured Workloads guide for steps on deploying a FedRAMP aligned Assured Workloads environment.

This guide also contains recommended security configurations related to role-based access control (RBAC), data protection |and retention, audit logging, and monitoring that align with the FedRAMP requirements. It covers post-deployment verification steps in 3.7 Post-Deployment Verification using the Google Cloud console to verify resources deployed and their corresponding security parameters.

1.3 Cloud Data Protection Toolkit

The Cloud Data Protection Toolkit is an open-source utility for provisioning and managing Google Cloud projects. The toolkit combines leading practices for infrastructure-as-code, security configurations, and for provisioning Google Cloud products into a comprehensive end-to-end framework. The declarative “deployment templates” (written in HashiCorp configuration language (HCL)) makes it possible to validate the deployment workflow even before its implementation.

In summary, Cloud Data Protection Toolkit templates help with the following activities:

- Deploying identical environments (for example, development, test, and production) with minimal manual intervention.
- Minimizing build and deployment errors in comparison to manual builds.
- Configuring disaster recovery by enabling rapid deployment of failed workloads.
- Deploying infrastructure-related auditing and monitoring tools in parallel with workload deployment.
- Reducing maintenance costs by automating removal of unused resources in conjunction with capacity monitoring.

Cloud Data Protection Toolkit templates can update or restore the deployments to the required state, driving development efficiency. Also, changes to the Cloud Data Protection Toolkit template are tracked by maintaining it in a code repository. This drives accountability and maintains discipline and quality control.

Google has published Cloud Data Protection Toolkit as an open-source repository, which you can clone and use to deploy the templates. To learn more about Cloud Data Protection Toolkit, refer to the repository on GitHub.

2. FedRAMP compliance in Google Cloud

FedRAMP is a U.S. government-wide program that provides a standardized approach to security assessment, authorization, and continuous monitoring for cloud products and services offered to US federal government agencies. Most federal agency cloud deployments and service models, other than certain on-premises private clouds, must meet FedRAMP requirements at the appropriate (Low, Moderate, or High) risk impact level.

The FedRAMP High authorization obtained by Google Cloud enables government agencies that process high impact workloads to adopt technology at a much higher velocity and at the same scale as commercial customers.

For details on Google Cloud services covered by FedRAMP, refer to the FedRAMP Marketplace.

3. Deploying a FedRAMP aligned three-tier workload using Cloud Data Protection Toolkit

This section describes an example of a FedRAMP aligned three-tier architecture and how to deploy it using the Cloud Data Protection Toolkit.

To align with FedRAMP compliance requirements, a three-tier architecture should be protected through the implementation of technical controls such as:

- Access control
- Audit and accountability
- Configuration management
- Contingency planning
- Identification and authentication
- Risk assessment
- Security assessment and authorization
- System and communications protection
- System and information integrity

The sections below describe how to implement these technical security controls for a specific use-case. They include the in-scope Google Cloud products and services described in the reference architecture below. For additional guidance, also refer to the FedRAMP Shared Security Model and Google Cloud FedRAMP Implementation Guide.

3.1 FedRAMP aligned three-tier reference architecture

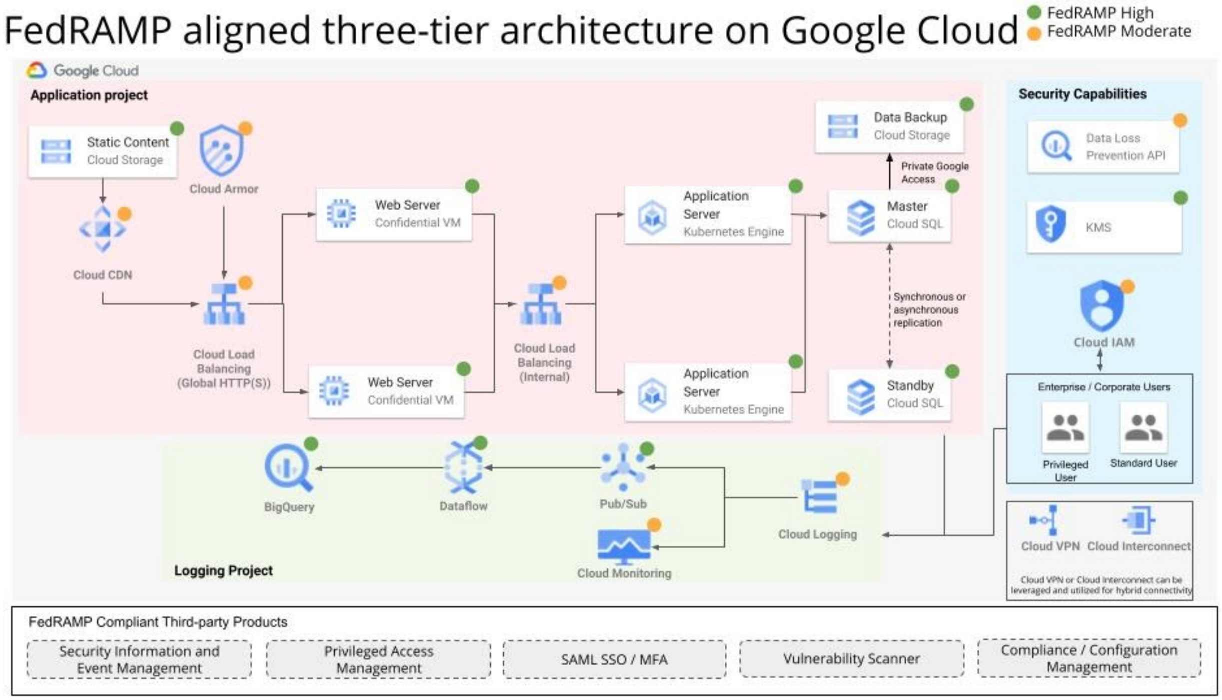


Figure 1. Architecture diagram for FedRAMP aligned three-tier workload on Google Cloud.

The details on the architecture and its components are:

Cloud Load Balancing (FedRAMP Moderate): HTTP(S) requests are first handled by Cloud Load Balancing which automatically distributes the incoming traffic amongst multiple Compute Engine instances hosting the web server and backend bucket.

Cloud Storage (FedRAMP High): Cloud Storage bucket acts as a backend service, used to store static content.

Cloud CDN (FedRAMP Moderate): Cloud CDN is used to deliver cached content from the backend bucket.

Web servers: Managed Instance Group (MIG) with Confidential Virtual Machines (VMs) (FedRAMP High) is used to host the web server because of MIG's scalability and Confidential VM's built-in optimization of both performance and security.

The application servers: Private GKE clusters (FedRAMP High) host application servers for deploying, managing, and scaling containerized applications.

Cloud SQL (FedRAMP High): A private Cloud SQL instance stores the application data and Cloud Storage is used as an automated backup location for Cloud SQL. (It is not required for the user to create the backup Cloud Storage bucket because Cloud SQL automates the backup and recovery.)

Cloud Logging (FedRAMP Moderate): All the logs from the application project are exported to BigQuery (FedRAMP High) using Pub/Sub (FedRAMP High) and Dataflow (FedRAMP High) in the Logging project for longer storage and analysis. The logs from Cloud Logging can also be monitored in Cloud Monitoring by configuring custom metrics as required.

The Privileged user and Standard user in the architecture represent the end users who securely connect to consume the workload.

The user can configure Cloud VPN or Cloud Interconnect to establish on-premises connectivity if required. The user may choose to add custom code to deploy these additional services

Note: *To meet FedRAMP Moderate requirements, the user deploying the toolkit will likely also have to configure some additional security controls such as security information and event management (SIEM), privileged access management, vulnerability scanning, and compliance and configuration management for the Google Cloud services deployed using this guide. These additional security controls are not covered in the guide. Additional services covered in the architecture can be integrated into the final solution as needed by the user.*

Though most of the components in the sample architecture below can be implemented using the Cloud Data Protection Toolkit templates, some of the capabilities, such as Identity Access Management (IAM), multi-factor authentication, Cloud Key Management Service (KMS) (with customer managed encryption keys (CMEK)), and Cloud Data Loss Prevention (DLP) API, will require additional custom configuration for integration with existing internal and external systems.

Use the three-tier architecture to deploy a web-based application on Google Cloud using Compute Engine instances and Google Kubernetes Engine (GKE), with Cloud SQL for hosting data. The entire architecture is deployed as two projects by the template. One of the projects contains all the resources deployed for use by the application and web server, while the other project contains resources to build a logging pipeline to store, monitor, and analyze logs from Cloud Logging.

3.2 In-scope Product Guidance

3.2.1 Cloud Storage

As part of the solution architecture, Cloud Storage buckets store the static content for web-hosted applications and is used to back up data from Cloud SQL.

To learn more about Cloud Storage and the parameters discussed below, refer to the Cloud Storage documentation and resource configuration respectively.

FedRAMP Guidance for Cloud Storage

FedRAMP Control Family	Default Configurations	User-Controlled Configurations (via the Data Protection Toolkit)
Access Control [AC-2(7)]	Users in the owners' group of the project are allowed to perform any action allowed by the project's organization policies.	Use Google Workspace or Cloud Identity to create users and groups. Additional role-member bindings can be added as mentioned in the template to control access to the storage bucket.
System and Information Integrity [SI-12]	Accidentally deleted or overwritten objects cannot be retrieved as versioning is disabled by default.	By default, versioning is always enabled when deployed using the toolkit. *Note: The Data Protection Toolkit has versioning as a mandatory parameter for deployment.*
Risk Assessment [RA-2]	Labels are not provided by default.	Label parameter values need to be specified per requirements in the template.

Refer to the accompanying *.hcl Template Configuration for a detailed configuration of Cloud Storage.

Note: For options for the customizable parameters in the template, refer to Cloud Storage Guidance for Terraform. The configurable values in the below template are indicative only. Please modify them to match specific requirements.

FedRAMP Control Family	FedRAMP Control Name	Code Block Description
Access Control	**Access Control** **[AC-2(7)]**	Refer to Code Block 3.2.1.a **iam_members:** Configuration for assigning roles to members and granting appropriate level permissions to the services. **role:** The role to be assigned to the user. **member:** Google Workspace or Cloud Identity users or groups to which the above role is assigned.
System and Information Integrity	**Information Handling and Retention** **[SI-12]**	Refer to Code Block 3.2.1.b **lifecycle_rule:** An array of objects where each object is a rule consisting of an action and a set of conditions. If multiple conditions are specified in a rule, an object has to match all of the conditions for the action to be taken. If multiple rules are specified with the same action, the action is taken when an object matches the condition(s) in any of the rules. Each rule should contain only one action.
Risk Assessment	**Security Categorization** **[RA-2]**	Refer to Code Block 3.2.1.c **labels:** They identify assets based on their classification. This can be used to identify the appropriate types of data stored within the service.

3.2.2 BigQuery

As part of the solution architecture, BigQuery is used to store and analyse log data that comes from Cloud Logging.

To learn more about BigQuery and the parameters discussed below, refer to the BigQuery documentation and resource configuration respectively.

FedRAMP Guidance for BigQuery

FedRAMP Control Family	Default Configurations	User-Controlled Configurations (via the Data Protection Toolkit)
Access Control [AC-2(7)]	Users in the owners' group of the project are allowed to perform any action allowed by the project's organization policies.	Use Google Workspace or Cloud Identity to create users and groups. Additional role-member bindings can be added as mentioned in the template using the access block to control access to BigQuery.
Risk Assessment [RA-2]	Labels are not provided by default.	Label parameter values can be specified to identify assets as per the required values in the template.
Audit and Accountability [AU-9 (2)], [AU-11]	BigQuery is a fully-managed, serverless data warehouse that enables scalable analysis over petabytes of data.	As part of the solution architecture, BigQuery is used to store audit log data that comes from Cloud Logging. The log data is backed up near real time in a separate Logging project. Users can choose to enable 'filter' to backup subset of audit logs.
		The data is retained in BigQuery for 90 days. Users may choose to change the retention period based on requirement.

Refer to the accompanying *.hcl Template Configurations for a detailed configuration of BigQuery.

Note: For options for the customizable parameters in the template, refer to the BigQuery guidance for Terraform. The configurable values in the template are indicative only. Please modify them to match specific requirements.

FedRAMP Control Family	FedRAMP Control Name	Code Block Description
Access Control	**Account Management | Role-Based Schemes [AC-2 (7)]**	**Refer to Code Block 3.2.2.a** **access:** An array of objects that define dataset access for one or more entities. Each object has a role and a user entity to which the role must be assigned. These user entities can be domain, group_by_email, user_by_email or special groups.
Risk Assessment	**Security Categorization [RA-2]**	**Refer to Code Block 3.2.2.b** **labels:** Labels identify assets based on their classification. Labels identify the appropriate types of data stored within the service.
Audit and Accountability	**Protection of Audit Information | Audit Backup on Separate Physical Systems / Components [AU-9 (2)]**	**Refer to Code Block 3.2.2.c** **log_type:** Defines the logging level. DATA_READ, DATA_WRITE and ADMIN_READ capture different types of events. Admin activity logs are enabled by default and cannot be disabled. **audit_log_config:** A nested block that defines the operations you'd like to log. **filter:** The filter to apply when exporting logs. Only log entries that match the filter are exported. See Advanced Log Filters for information on how to write a filter.
	Audit Record Retention [AU-11]	**Refer to Code Block 3.2.2.d** **default_table_expiration_ms:** The default lifetime of all tables in the dataset, in milliseconds. The minimum value is 3600000 milliseconds (one hour). After this property is set, all newly-created tables in the dataset will have an expirationTime property set to the creation time plus the value in this property. Changing the value will only affect new tables, not existing ones. When the expirationTime for a given table is reached, that table will be deleted automatically. If a table's expirationTime is modified or removed before the table expires, or if you provide an explicit expirationTime when creating a table, that value takes precedence over the default expiration time indicated by this property. For the purpose of this solution, the Data Protection Toolkit template has a retention period set to 90 days.

3.2.3 Cloud SQL

Cloud SQL instances in this reference architecture are used to store application data, handling large amounts of data transactions that require consistency.

To learn more about Cloud SQL and the parameters discussed below, refer to the Cloud SQL documentation and resource configuration respectively.

FedRAMP Guidance for Cloud SQL

FedRAMP Control Family	Default Configurations	User-Controlled Configurations (via the Data Protection Toolkit)
Access Control [AC-2(7)]	Users in the owners' group of the project are allowed to perform any action allowed by the project's organization policies.	Use Google Workspace or Cloud Identity to create users and groups. Additional role-member bindings can be added as mentioned in the template to control access to the Compute Engine instance.
Configuration Management [CM-7]	Instances should have public and private network access defined. These are not enabled by default.	Instances are confined to a private network. Cloud SQL instance and its replica are created in a Virtual Private Cloud (VPC) network as configured in the template.
Contingency Planning [CP-10]	A read/read-only replica is not created by default for Cloud SQL.	A failover replica is configured for the main SQL instance which ensures the availability of data in case of a physical or a technical incident.

Refer to the accompanying *.hcl Template Configurations for a detailed configuration of Cloud SQL.

Note: For options for the customizable parameters in the template, refer to the Cloud SQL guidance for Terraform. The configurable values in the template are indicative only. Please modify them to match specific requirements.

FedRAMP Control Family	FedRAMP Control Name	Code Block Description
Access Control	**Account Management | Role-Based Schemes [AC-2 (7)]**	**Refer to Code Block 3.2.3.a** **role:** The role to be assigned to the user. **member:** Google Workspace or Cloud Identity users or groups to which the above role is assigned.
Configuration Management	**Least functionality [CM-7]**	Private Cloud SQL instances are enabled by default. The template prohibits or restricts assignment of public IPv4 addresses to Cloud SQL and permits access only from the VPC network where managed instance group and GKE cluster reside.
Contingency Planning	**Information System Recovery and Reconstitution [CP-2]**	The template enables automated backup, point in time recovery, and failover replicas by default. Users are not required to configure these parameters.

3.2.4 Confidential VM

As part of solution architecture, Confidential VMs are used to host the web server. The global load balancer distributes traffic between the two Confidential VMs.

Note: Confidential VM is available in the same zones and regions as N2D Compute Engine VMs. Refer to the available regions and zones for Confidential VM instances locations.

To learn more about Confidential VMs and the parameters discussed below, refer to the Confidential VM documentation and resource configuration respectively.

FedRAMP Guidance for Confidential VM

FedRAMP Control Family	Default Configurations	User-Controlled Configurations (via the Data Protection Toolkit)
Access Control [AC-2(7)]	Users in the owners' group of the project are allowed to perform any action allowed by the project's organization policies.	Use Google Workspace or Cloud Identity to create users and groups. Additional role-member bindings can be added as mentioned in the template to control access to the compute instance.
System and Communications Protection [SC-5, SC-6, SC-12]	An instance template serves as a basic configuration wrapper for the instances in a managed instance group.	Auto-scaling is enabled in the Data Protection Toolkit and at least two instances in different zones are maintained.
	Data is encrypted at-rest and in-transit, with Google managing the security keys.	Custom encryption keys can be used for encryption of data stored on Google Compute Engine disks. See Customer-managed encryption keys for more information. See Encryption at rest in Google Cloud for more information on how Google encrypts data at rest by default and to understand the key management options.
Risk Assessment [RA-2]	Labels are not provided by default.	Label parameter values need to be specified as per the given values in the example as guided by the template.
Configuration Management [CM-7]	The instance can be confined to use a private network or just a subnet of that network. Also, the instance may be assigned a public IP.	The VM instance is created under a subnet of a user-defined VPC network as configured in the template and public access can be restricted as required.

Refer to the accompanying *.hcl Template Configurations for a detailed configuration of Confidential VM.

Note: For options for the customizable parameters in the template, refer to Cloud Compute Instance for Terraform. The configurable values in the template are indicative only. Please modify them to match specific requirements.

FedRAMP Control Family	FedRAMP Control Name	Code Block Description
Access Control	**Account Management | Role-Based Schemes [AC-2 (7)]**	Refer to Code Block 3.2.4.a **role:** The role to be assigned to the user. **member:** Google Workspace or Cloud Identity users or groups to which the above role is assigned.
Systems and Communications Protection	**Resource Availability [SC-6] / Denial of Service Protection [SC-5]**	Refer to Code Block 3.2.4.b **autoscaling_policy:** The configuration parameters for the auto scaling algorithm. You can define one or more of the policies for an autoscaler: cpuUtilization, customMetricUtilizations, and loadBalancingUtilization. If none of these are specified, the default will be to autoscale based on cpuUtilization to 0.6 or 60%. **min_replicas:** The minimum number of replicas that the autoscaler can scale down to. This cannot be less than 0. If not provided, the autoscaler will choose a default value depending on the maximum number of instances allowed. **max_replicas:** The maximum number of instances that the autoscaler can scale up to. This is required when creating or updating an autoscaler. The maximum number of replicas should not be lower than the minimal number of replicas.
	Cryptographic key Establishment and Management [SC-12]	Refer to Code Block 3.2.4.c **disk_encryption_key:** A 256-bit custom managed encryption key to encrypt this disk. If a custom key is not specified, Google Cloud default encryption will be used for the data at rest. See Encryption at rest in Google Cloud for more information on how Google encrypts data at rest by default and to understand the key management options. **kms_key_self_link**: A key stored on Cloud KMS. If a custom KMS key is not specified, Google Cloud default encryption will be used for the data at rest. See Encryption at rest in Google Cloud for more information on how Google encrypts data at rest by default and to understand the key management options.

Risk Assessment	**Security Categorization [RA-2]**	**Refer to Code Block 3.2.4.d** **labels:** Labels are used to identify assets based on their classification. This can be used to identify the appropriate types of data stored within the service.
Configuration Management	**Least Functionality [CM-7]**	Instances inside managed instance groups are private and use Private Google Access. ('**access_config**' block in code is removed to make the instance use only private IPs.) Instances can reach the internet if required using Cloud NAT.

3.2.5 Pub/Sub

As a part of the solution architecture, Pub/Sub is used to gather logs from Cloud Logging and move them to Dataflow before storing the logs in BigQuery.

To learn more about Pub/Sub and the parameters discussed below, refer to the Pub/Sub documentation and resource configuration respectively.

FedRAMP Guidance for Pub/Sub

FedRAMP Control Family	Default Configurations	User-Controlled Configurations (via the Data Protection Toolkit)
Access Control [AC-2(7)]	Users in the owners' group of the project are allowed to perform any action allowed by the project's organization policies.	Use Google Workspace or Cloud Identity to create users and groups. Additional role-member bindings can be added as mentioned in the template using the access block to control access to the Pub/Sub topic or subscription.
Risk Assessment [RA-2]	Labels are not provided by default.	Label parameter values can be specified to identify assets as per the required values in the template.

Refer to the accompanying *.hcl Template Configurations for a detailed configuration of Pub/Sub.

Note: For options for the customizable parameters in the template, refer to Pub/Sub for Terraform. The configurable values in the template are indicative only. Please modify them to match specific requirements.

FedRAMP Control Family	FedRAMP Control Name	Code Block Description
Access Control	**Account Management | Role-Based Schemes [AC-2 (7)]**	**Refer to Code Block 3.2.5.a** **role:** The role to be assigned to the user. **member:** Google Workspace or Cloud Identity users or groups to which the above role is assigned.
Risk Assessment	**Security Categorization [RA-2]**	**Refer to Code Block 3.2.5.b** **labels:** Labels are used to identify assets based on their classification. This can be used to identify the appropriate types of data processed or stored in the service.

3.2.6 Google Kubernetes Engine

As a part of solution architecture, GKE is used to host the application server.

To learn more about GKE and the parameters discussed below, refer to the Google Kubernetes Engine documentation and resource configuration respectively.

FedRAMP Guidance for Google Kubernetes Engine

FedRAMP Control Family	Default Configurations	User-Controlled Configurations (via the Data Protection Toolkit)
Access Control [AC-2(7)]	Users in the owners' group of the project are allowed to perform any action allowed by the project's organization policies.	Use Google Workspace or Cloud Identity to create users and groups. Additional role-member bindings can be added as mentioned in the template using the access block to control access to GKE.
Configuration Management [CM-7]	Private cluster is enabled by default and it cannot be made public. It provides additional network security such as private nodes.	The cluster is created under a subnet of a user-defined VPC network as configured in the template and public access is restricted, except for allowed IPs.
Identification and Authentication [IA-4]	The template enables a few security parameters for GKE clusters.	Binary authorization and shielded nodes features are enabled by default in the template and cannot be disabled.
Risk Assessment [RA-2]	Labels are not provided by default.	Label parameter values can be specified to identify assets as per the required values in the template.

Refer to the accompanying *.hcl Template Configurations for a detailed configuration of GKE.

Note: For options for the customizable parameters in the template, refer to Google Kubernetes Engine for Terraform. The configurable values in the template are indicative only. Please modify them to match specific requirements.

FedRAMP Control Family	FedRAMP Control Name	Code Block Description
Access Control	**Account Management | Role-Based Schemes [AC-2 (7)]**	Refer to Code Block 3.2.6.a **role:** The role to be assigned to the user. **member:** Google Workspace or Cloud Identity users or groups to which the above role is assigned.
Configuration Management	**Least functionality [CM-7]**	Refer to Code Block 3.2.6.b A private cluster is enabled by default, access to the private cluster can be configured by the user using code block 3.2.6.b (1). You can restrict access to the master GKE node only from authorized IPv4 addresses. To create an allow list of IPs (other than managed instance groups IPs), add code block 3.2.6.b (2).
System and Communications Protection	**Resource Availability / Denial of Service Protection [SC-5, SC-6]**	Refer to Code Block 3.2.6.c **min_count:** Used to specify the minimum number of nodes (per zone) in a node pool. **max_count:** Used to specify the maximum number of nodes (per zone) in a node pool.
Identification and Authentication	**Identifier Management [IA-4]**	Binary authorization and shielded nodes features are enabled by default using Data Protection Toolkit and cannot be disabled.
Risk Assessment	**Security Categorization [RA-2]**	Refer to Code Block 3.2.6.d **labels:** They identify assets based on their classification. This can be used to identify the appropriate types of data stored within the service.

3.2.7 Dataflow

As a part of the solution architecture, Dataflow is used to process logging data before moving it to BigQuery for storage and analysis.

To learn more about Dataflow and the parameters discussed below, refer to the Dataflow documentation and resource configuration respectively.

FedRAMP Guidance for Dataflow

FedRAMP Control Family	Default Configurations	User-Controlled Configurations (via the Data Protection Toolkit)
Access Control [AC-2(7)]	Users in the owners' group of the project are allowed to perform any action allowed by the project's organization policies.	Use Google Workspace or Cloud Identity to create users and groups. Additional role-member bindings can be added as mentioned in the template using the access block to control access to the Dataflow.
Risk Assessment [RA-2]	Labels are not provided by default.	Label parameter values need to be specified as described in the template.

Refer to accompanying *.hcl Template Configurations for a detailed configuration of Dataflow.

Note: For options for the customizable parameters in the template, refer to Google Dataflow for Terraform. The configurable values in the template are indicative only. Please modify them to match specific requirements.

FedRAMP Control Family	FedRAMP Control Name	Code Block Description
Access Control	**Account Management | Role-Based Schemes [AC-2 (7)]**	**Refer to Code Block 3.2.7.a** **role:** The role to be assigned to the user. **member:** Google Workspace or Cloud Identity users or groups to which the above role is assigned.
Risk Assessment	**Security Categorization [RA-2]**	**Refer to Code Block 3.2.7.b** **labels:** Labels are used to identify assets based on their classification. This can be used to identify the appropriate types of data processed or stored in the service.

3.2.8 Cloud load balancing (external)

As a part of the solution architecture, the cloud load balancing distributes traffic to web servers hosted in different zones on Confidential VMs.

To learn more about cloud load balancing and the parameters discussed below, refer to the Cloud Load Balancing documentation and resource configuration respectively.

FedRAMP Guidance for cloud load balancing

FedRAMP Control Family	Default Configurations	User-Controlled Configurations (via the Data Protection Toolkit)
Access Control [AC-2(7)]	Users in the owners' group of the project are allowed to perform any action allowed by the project's organization policies.	Use Google Workspace or Cloud Identity to create users and groups. Additional role-member bindings can be added as mentioned in the template to control access to the storage bucket.
System and Communications Protection [SC-8, SC-23]	Google Cloud uses SSL certificates to provide privacy and security from a client to a load balancer. The load balancer must have an SSL certificate and the certificate's corresponding private key.	Use a Google managed SSL certificate for the cloud load balancing. The user also has an option to add a self-managed certificate using the template.

Refer to the accompanying *.hcl Template Configurations for a detailed configuration of cloud load balancing.

Note: For options for the customizable parameters in the template, refer to Cloud Load Balancing for Terraform. The configurable values in the template are indicative only. Please modify them to match specific requirements.

FedRAMP Control Family	FedRAMP Control Name	Code Block Description
Access Control	**Account Management | Role-Based Schemes [AC-2 (7)]**	Refer to Code Block 3.2.8.a **role:** The role to be assigned to the user. **member:** Google Workspace or Cloud Identity users or groups to which the above role is assigned.
System and Communications Protection	**Session Authenticity / Transmission Confidentiality and Integrity [SC-8, SC-23]**	Refer to Code Block 3.2.8.b The code block can be utilized when Google managed certificates are used. If the user deploying the template chooses to use a self-managed certificate, the parameters from code block 3.2.8.b (1), code block 3.2.8.b (2), and code block 3.2.8.b (3) will change.

3.2.9 Cloud Armor

As a part of the solution architecture, Cloud Armor restricts traffic to the external load balancer.

To learn more about Cloud Armor and the parameters discussed below, refer to the Cloud Armor documentation and resource configuration respectively.

FedRAMP Guidance for Cloud Armor

FedRAMP Control Family	Default Configurations	User-Controlled Configurations (via the Data Protection Toolkit)
Access Control [AC-2(7)]	Users in the owners' group of the project are allowed to perform any action allowed by the project's organization policies.	Use Google Workspace or Cloud Identity to create users and groups. Additional role-member bindings can be added as mentioned in the template using the access block to control access to the Cloud Armor.
Security Assessment and Authorization [CA-3 (5)]	Deny all the traffic from the internet to the external global load balancer.	Users can add security policies to allow traffic from specific external IPs.
System and Communication Protection [SC-5]	Cloud Armor provides always-on DDoS protection against network or protocol-based volumetric DDoS attacks for applications or services behind external HTTP(S) load balancers, SSL proxy load balancers, or TCP proxy load balancers.	Cloud Armor security policy can be configured to evaluate multiple pre-configured expressions.

Refer to the accompanying *.hcl Template Configurations for a detailed configuration of Cloud Armor.

Note: For options for the customizable parameters in the template, refer to Cloud Armor for Terraform. The configurable values in the template are indicative only. Please modify them to match specific requirements.

FedRAMP Control Family	FedRAMP Control Name	Code Block Description
Access Control	**Account Management | Role-Based Schemes [AC-2 (7)]**	Refer to Code Block 3.2.9.a **role:** The role to be assigned to the user. **member:** Google Workspace or Cloud Identity users or groups to which the above role is assigned.
Security Assessment and Authorization	**System Interconnections | Restrictions on External System Connection [CA-3 (5)]**	Refer to Code Block 3.2.9.b **action:** Specifies the action to be taken if ‘match’ matches the request. It could either be ‘allow’ or ‘deny’. **src_ip_ranges:** Set of IP addresses or ranges (IPv4 or IPv6) in classless inter-domain routing (CIDR) notation to match against inbound traffic. There is a limit of 10 IP ranges per rule. A value of '*' matches all IPs (can be used to override the default behavior).
System and Communication Protection	**Denial of Service Protection [SC-5]**	Refer to Code Block 3.2.9.b **expression:** Textual representation of an expression in Common Expression Language (CEL) syntax. The application context of the containing message determines which well-known feature set of CEL is supported.

3.3 Environment setup

Run the Cloud Data Protection Toolkit locally on a computer or by using Google Cloud Shell.

Before you run the toolkit locally, install the following tools:

- Go (1.14+): an open source programming language to build software.
- Terraform (0.14.4+): a cloud provisioning tool.
- Cloud SDK: a set of tools for managing resources and applications hosted on Google Cloud.
- Git: a distributed version control system.
- Google Workspace or Cloud Identity: Privileges to modify users and groups in Google Workspaces or Cloud Identity.
- Google Cloud Organization A Google Cloud organization with a Billing Account.
- A domain purchased from a Domain registrar (for example, Google Domains).

If Google Cloud Shell is used to run the toolkit, Go (1.16), Git, and Cloud SDK are preinstalled. However, newer version of Terraform must be installed in Google Cloud Shell before deploying the template using the below steps:

1. Go to Terraform Downloads and copy the link of 'Linux 64-bit' binary by right-clicking on it.

2. Download the Linux 64-bit binary into Google Cloud Shell:

```
$ wget <link
copied from
Terraform
Downloads>
```

3. Download the Linux 64-bit binary into Google Cloud Shell:

```
$ wget <link
copied from
Terraform
Downloads>
```

4. Move the unzipped terraform binary to /usr/local/bin:

```
$ sudo cp
terraform
/usr/local/bin
```

3.4 Access control

Sections 3.4.1 and 3.4.2 explain the access grants before, during, and after the deployment of resources. Deploying the Data Protection Toolkit requires an ‘owner’ (privileged) role at the organisation, folder, or project level to deploy resources. This approach ensures that only required accesses are granted based on requirement in each stage of the project creation lifecycle.

3.4.1 Pre-deployment access control

Before deploying a template, create three groups:

Owner:
project-owners@{DOMAIN}. This group is granted the owner's role for the project, which allows members to do anything permitted by organization policies within the project. Additions to the owner’s group should be short term and controlled tightly. Members of this group get owners access to the devops project to make changes to the CI/CD project or to make changes to the Terraform state. Make sure to include yourself as an owner of this group. Otherwise, you might lose access to the devops project after the ownership is transferred to this group.

Admin: org-admins@{DOMAIN}. Members of this group get administrative access to the organization or folder. This group can be used in break-glass situations to give humans access to the organization or folder to make changes. Include yourself as a member of this group to deploy the toolkit templates.

Cloud-users:
project-cloud-users @{DOMAIN} Members of this group will get access to the resources deployed by the toolkit after deployment.

Note: The names of groups are for reference only. Users can name the groups differently.

The user groups running the template must have the following IAM roles:

- roles/resourcemanager.organizationAdmin on the organization for organization deployment.
- roles/resourcemanager.folderAdmin on the folder for folder deployment. (This role is required if workloads are deployed under a folder instead of organization.)
- roles/resourcemanager.projectCreator on the organization or folder.
- roles/billing.admin on the billing account.
- roles/owner on assured-workload projects for FedRAMP aligned workload deployment. (This role is specifically given to the owners group.)

The Data Protection Toolkit needs ‘owner’ permissions for the projects in the template to provision their resources and “project creator” permission to create the DevOps project under the organization or folder. So initially, to grant provisioning access, the user identity deploying the template is temporarily added to the “owner” group and “Admin” group for provisioning projects and resources. User identity can be removed from groups after the deployment is done.

3.4.2 Post-deployment access control

Post deployment of a template, the Data Protection Toolkit does not remove the deploying user (or user identity) from the groups. Ensure that only specific pre-approved owners continue to have control after deployment.

Note: The Data Protection Toolkit grants roles and permissions to users, groups, and entities (for example, service accounts). To further customize access after deployment is complete, create user groups to control access to the projects and their underlying resources using the Google Workspace or Cloud Identity Admin Console, Cloud Identity, or Google Cloud Directory Sync (GCDS). These user groups can be granted custom roles and permissions using IAM and conditional access policies. For further information, refer to the IAM documentation.

3.5 Deployment phases

This section describes the deployment of FedRAMP-aligned three-tier workload configuration templates.

Note: The Data Protection Toolkit template deploys resources on the "Assured Workloads", however the toolkit does not create these two Assured Workloads. Before you deploy the toolkit, create two FedRAMP Moderate Assured Workloads (one for the three-tier workload and one for the logging project) using the console or gcloud. Refer to this Create a new workload environment. Create Assured Workloads in regions where N2D machine type is supported. Refer to Regions and zones to see which regions support the N2D machine type.

Before you run the tfengine to generate terraform files, complete these steps:

Clone the Data Protection Toolkit Git repository to a folder (locally or on cloud shell).

Install tfengine.

Clone the FedRAMP aligned three-tier workload HCL files into a folder before running tfengine.

After the HCL files are cloned, configure the variable values in commonVariables.hcl and variable.hcl files based on the requirements. Refer the README.md in the GitHub repository for details.

```
#clone dpt git repository
$ git clone https://github.com/GoogleCloudPlatform/healthcare-data-protection-suite
$ cd healthcare-data-protection-suite


#install tfengine

$ go install ./cmd/tfengine


#clone modularised .hcl files from github to a folder in local machine or cloud shell
$ git clone https://github.com/GoogleCloudPlatform/gcp-fedramp-quickstart.git

# configure the variable values in commonVariables.hcl and variable.hcl files based
# on requirements. Refer the README.md in the GitHub repository for details.
```

3.5.1 Generate Terraform files

This step generates six folders or subfolders with Terraform configuration files in the –output_path location.

Generated folders:

- devops
- logging/network
- logging/workload
- threetierworkload/network
- threetierworkload/loadbalancer-mig
- threetierworkload/gke-sql

To generate terraform configuration files using tfengine, run the following command:

```
# config path is the path to downloaded .hcl files

$ tfengine --config_path=/{path-to-variablefile}/commonVariables.hcl
--output_path=/{output-path}

# {path-to-variablefile}: path to commonVariable.hcl file.
# {output-path}: Folder path, where terraform configuration files are
# generated by tfengine.
```

3.5.2 Deploy Terraform configurations

After generating terraform configurations using tfengine, run the generated main.tf files in the following order.

1. Open the DevOps folder and run the Terraform configuration. This will deploy a project and a Terraform state storage bucket in the project with the name of choosing.

```
$ cd /{output-path}/devops
$ terraform init
$ terraform apply
```

2. After the project and state bucket are deployed, go to devops.hcl file, uncomment or set the **enable_gcs_backend** to **true** in the devops template recipe:

```
template "devops" {
  recipe_path = "recipes/devops.hcl"
  output_path = "./devops"
  data = {
    # TODO(user): Uncomment and re-run the engine after the generated devops
    # module has been deployed.
    # Run `terraform init` in the devops module to backup its state to GCS.
    #enable_gcs_backend = true

    admins_group = {
      id = “{{.admin_group}}”
      exists = true
    }
```

3. Open the DevOps folder and run the Terraform configuration. This will deploy a project and a Terraform state storage bucket in the project with the name of choosing.

```
$ tfengine --config_path=/{path}/commonVariables.hcl --output_path=/{path}
$ cd /{output-path}/devops
$ terraform init -force-copy
```

4. After the states are transferred to the state bucket, deploy network resources in the Logging project (Assured Workload).

```
$ cd /{output-path}/logging/network
$ terraform init
$ terraform apply
```

5. After the logging network is deployed, run the following commands to deploy the remaining resources in the Logging project (Assured Workload) such as Dataflow, Pub/Sub, BigQuery, and so on:

```
$ cd /{output-path}/logging/workload
$ terraform init
$ terraform apply
```

6. Deploy the network in the threetierworkload/network folder to create the network, private service access and enable the APIs in the three-tier workload project:

```
$ cd /{output-path}/threetierworkload/network
$ terraform init
$ terraform apply
```

7. Deploy the resources in the threetierworkload/loadbalancer-mig folder to create resources such as MIG, Cloud Load Balancing, Cloud Armor, Google managed SSL, and managed DNS:.

```
$ cd /{output-path}/threetierworkload/loadbalancer-mig
$ terraform init
$ terraform apply
```

8. Deploy the additional resources in the threetierworkload/gke-sql folder to create resources such GKE and SQL:

```
$ cd /{output-path}/threetierworkload/gke-sql
$ terraform init
$ terraform apply
```

3.6 Pre-deployment setup

3.6.1 DevOps project deployments

This DevOps template includes code to create a DevOps project.

It creates a new project if a project with the same ID does not exist in the organization.

It creates a terraform state storage bucket. The bucket name has to be unique.

It adds admin and owners groups for access control on the project.

If CI/Cd pipeline have to be created using the CI/CD recipe of Cloud Data Protection Toolkit refer the github link and example. This document only explains the creation of DevOps, three tier workload and logging projects.

```
template "devops" {
 recipe_path = "git://github.com/GoogleCloudPlatform/healthcare-data-protection-suite//templates/tfengine/recipes/devops.hcl"
 output_path = "./devops"
 data = {
   # TODO(user): Uncomment and re-run the engine after the generated devops module has been deployed.
   # Run `terraform init` in the devops module to back up its state to GCS.
   enable_gcs_backend = true

   admins_group = {
     id = "{{.admin_group}}"
     exists = true
   }

   project = {
     project_id = "{{.devops_project_id}}"
     owners_group = {
       id = "{{.owners_group}}"
       exists = true
     }
   }
 }
}
```

3.6.2 Logging Project (Assured Workloads) deployments

The “Logging” template includes code to create resources in the Logging project (Assured Workload), assuming that the project already exists. Refer to Deployment Phases to create Assured Workloads.	template "logging" {
	 recipe_path =
	"git://github.com/GoogleCloudPlatform/healthcare-data-protection-suite//templates/tfengine/recipes/project.hcl"
	 output_path = "./logging/workload"
	 data = {
	    }
	}

This code enables a list of APIs that are required by the logging Assured Workload project.	apis = ["compute.googleapis.com", "iam.googleapis.com", "bigquery.googleapis.com", "bigqueryconnection.googleapis.com", "bigquerydatatransfer.googleapis.com", "bigqueryreservation.googleapis.com", "bigquerystorage.googleapis.com", "secretmanager.googleapis.com", "pubsub.googleapis.com", "dataflow.googleapis.com"]

In this code, one Pub/Sub topic with the corresponding subscription is created.

Add data classification labels to this logging project. These labels define the data that is pushed from the log sink in the three-tier workload project to the pubsub topic.

Acknowledgement deadline by the subscriber (Dataflow job) is applied to the subscription.

```
pubsub_topics = [{
         name = "{{.logs_streaming_pubsub_topic_name}}"
#Code Block 3.2.5.b
         labels = {
             data_type =
"{{.logs_streaming_pubsub_topic_datatype_label}}"
             data_criticality =
"{{.logs_streaming_pubsub_topic_data_criticality_label}}"
         }
         pull_subscriptions = [
             {
                 name =
"{{.logs_streaming_pubsub_subscription_name}}"
                 ack_deadline_seconds =
{{.logs_streaming_pubsub_subscription_acknowledgmenet_
seconds}}
             }
         ]
       }]
```

In this code, a Cloud Storage bucket for storing the temp data of a Dataflow job is created.

The template code assigns the object viewer role to the specified member to the specified group.

Versioning is enabled by default.

```
storage_buckets = [{
           name = "{{.dataflow_temp_storage_bucket_name}}"
           resource_name = "dataflow_temp_storage_bucket"
           labels = {
               data_type =
"{{.dataflow_temp_storage_bucket_datatype_label}}"
               data_criticality =
"{{.dataflow_temp_storage_bucket_data_criticality_label}}"
           }
           iam_members = [
             {
               role   = "roles/storage.objectViewer"
               member = "group:{{.cloud_users_group}}"
             }
          ]
       }]
```

In this code, the BigQuery data set is deployed under the logging project (Assured Workload).

The BigQuery data set stores transformed logs from the three-tier workload project.

The template also accommodates provision of role-based access to individual resources through IAM.

Note: The configurations are customizable and can be changed as required to meet specific use cases.

```
bigquery_datasets = [{
# Override Terraform resource name as it cannot start
with a number.
           resource_name               =
"log_analysis_dataset"
           dataset_id                  =
"{{.logs_storage_bigquery_dataset_name}}"
#Code Block 3.2.2.d
# Retains log records for 90 days. Can be customized to
retain for longer period
           default_table_expiration_ms = 7.776e+9
           #depends_on = ["$${module.project-services}"]
#Code Block 3.2.2.b
           labels = {
               data_type =
"{{.logs_streaming_pubsub_topic_datatype_label}}"
               data_criticality =
"{{.logs_streaming_pubsub_topic_data_criticality_label}}"
           }
#Code Block 3.2.2.a
           access = [
           {
               role          = "roles/bigquery.dataOwner"
               special_group = "projectOwners"
           },
           {
               role           = "roles/bigquery.dataViewer"
               group_by_email = "{{.cloud_users_group}}"
           }
           ]
       }]
```

In this code, a Bigquery table is created based on the schema provided. If the schema is left commented, an empty table is created. This will cause errors in the Dataflow job. Set the table schema based on the “logs sink filter” (three-tier workload project) and “pubsub messages format”.

Data classification labels of the Pub/Sub topic are used as it is in this resource, because this is part of the same data lifecycle. User can change the template as required.

```
resource "google_bigquery_table" "logs_table" {
           dataset_id =
"$${module.log_analysis_dataset.bigquery_dataset.dataset_id
}"
           table_id =
"{{.logs_storage_bigquery_table_name}}"
           project = module.project.project_id
           labels = {
               data_type =
"{{.logs_streaming_pubsub_topic_datatype_label}}"
               data_criticality =
"{{.logs_streaming_pubsub_topic_data_criticality_label}}"
           }
# Uncomment and provide schema file path. If left
unchanged, this will create an empty table.
# Schema also can be given inline.
           #schema = file("/{replace with file
path}/schema.json") }
```

In this code, a Dataflow job is created to process logs from Pub/Sub.

The “Pub/Sub subscription to BigQuery” template provided by Google is used. Can be updated to use a custom template.

The data classification labels of the Pub/Sub topic are used as it is in this resource, because this is part of the same data lifecycle. User can change the template as required.

The max_workers field is used to specify scalability of worker nodes.

```
resource "google_dataflow_job" "psto_bq_job" {
           name  = "{{.data_flow_job_name}}"
           max_workers = {{.data_flow_job_max_workers}}
           on_delete = "cancel"
           project = module.project.project_id
           network               = "{{.dataflow_network_name}}"
           subnetwork            =
"regions/{{.logging_project_region}}/subnetworks/{{.dataflow_
subnet_name}}"
           ip_configuration      = "WORKER_IP_PRIVATE"
           region                = "{{.logging_project_region}}"
           depends_on =
[google_project_iam_binding.data_flow_service_account_
access_worker ]
           template_gcs_path =
"gs://dataflow-templates-us-central1/latest/PubSub_Subscription_
to_BigQuery"
           temp_gcs_location =
"$${module.dataflow_temp_storage_bucket.bucket.url}"
           service_account_email =
"$${google_service_account.data_flow_job_service_account.email}"
#Code Block 3.2.7.b
           labels = {
               data_type =
"{{.logs_streaming_pubsub_topic_datatype_label}}"
               data_criticality =
"{{.logs_streaming_pubsub_topic_data_criticality_label}}"
           }
           parameters = {
               inputSubscription =
module.logging_pubsub_topic.subscription_paths[0]
               outputTableSpec =
"$${google_bigquery_table.logs_table.project}:$${google_bigquery
_table.logs_table.dataset_id}.$${google_bigquery_table.logs_
table.table_id}"

#*****"$${google_bigquery_table.logs_table.project}:$${google_
bigquery_table.logs_table.dataset_id}.$${google_bigquery_table.
logs_table.table_id}"
           }
       }
```

In this code, the network, subnet, and firewall required for Dataflow job is created.

Private Google access is enabled for Dataflow workers to securely communicate with public APIs (BigQuery, Pub/Sub and Cloud Storage)

This code section resides in the network.hcl file (also used for three-tier workload network creation).

```
template "logging" {
 recipe_path =
"git://github.com/GoogleCloudPlatform/healthcare-data-protection-
suite//templates/tfengine/recipes/project.hcl"
 output_path = "./logging/network"
 data = {
   project = {
     project_id = "{{.logging_project_id}}"
     exists     = true
     apis = [
           "compute.googleapis.com",
           "iam.googleapis.com",
           "servicenetworking.googleapis.com",
            "logging.googleapis.com",
           "stackdriver.googleapis.com",
           "bigquery.googleapis.com",
           "bigqueryconnection.googleapis.com",
           "bigquerydatatransfer.googleapis.com",
           "bigqueryreservation.googleapis.com",
           "bigquerystorage.googleapis.com",
           "pubsub.googleapis.com",
           "dataflow.googleapis.com",
     ]
   }
   terraform_addons = {
       raw_config = <<EOF

       provider "google" {
           project     = "{{.logging_project_id}}"
           region      = "{{.logging_project_region}}"
       }

       provider "google-beta" {
           project     = "{{.logging_project_id}}"
           region      = "{{.logging_project_region}}"
       }

       # This firewall rule helps Dataflow worker VMs
communicate with each other
       resource "google_compute_firewall"
"dataflow_workers_internal_communication_firewall" {
           name    =
"dataflow-workers-internal-communication-firewall"
           network = "{{.dataflow_network_name}}"
           project = module.project.project_id
           depends_on = [ module.logging_network,
module.cloud_sql_private_service_access_logging_network]
           allow {
               protocol = "icmp"
           }

           allow {
               protocol = "tcp"
               ports    = ["12345-12346"]
           }
           source_ranges = [
               "{{.dataflow_subnet_ip_range}}"
           ]
           target_tags = ["dataflow"]
       }


   EOF
   }
   resources = {
       # Network used by Dataflow workers
       compute_networks =  [{

           name = "{{.dataflow_network_name}}"
           resource_name = "logging_network"
           # Enabling private Service access
           cloud_sql_private_service_access = {}
           subnets = [
               {
               name="{{.dataflow_subnet_name}}"
               compute_region="{{.logging_project_region}}"
               ip_range="{{.dataflow_subnet_ip_range}}"
               },
           ]
       }]
    }
 }
}
```

In this code, IAM permissions and bindings required for the Dataflow service account, and the Dataflow service agent are set at BigQuery and Pub/Sub topic.

Access to the cloud users group is given at the project level.

```
resource "google_project_iam_binding" "data_flow_access" {
           project = module.project.project_id
           role    = "roles/dataflow.developer"

           members = [
               "group:{{.cloud_users_group}}",
           ]
       }
       #Access given to Dataflow service account to write data
to BigQuery

       resource "google_bigquery_dataset_access"
"data_flow_service_account_access_bigquery" {
           dataset_id    =
"$${module.log_analysis_dataset.bigquery_dataset.dataset_id}"
           role          = "roles/bigquery.dataEditor"
           user_by_email =
google_service_account.data_flow_job_service_account.email
       }
       # Access given to Dataflow service account to write to
temp storage bucket
       resource
"google_storage_bucket_iam_binding"
"data_flow_service_account_access_bucket"
{
           bucket =
"$${module.dataflow_temp_storage_bucket.bucket.name}"
           role = "roles/storage.objectCreator"
           members = [

"serviceAccount:$${google_service_account.data_flow_job_service_
account.email}",
           ]
       }


       # This access is necessary for a Compute Engine service
account to execute work units for an Apache Beam pipeline
       resource
"google_project_iam_binding"
"data_flow_service_account_access_worker" {
           project = module.project.project_id
           role    = "roles/dataflow.worker"
```

```
        members = [
            "serviceAccount:$${google_service_account.data_flow_job_service_account.email}",
        ]
    }
    # PubSub subscriber access to dataflow service service account used by worker VMs to pull and acknowledge the messages
    # PubSub subscriber access to Dataflow service agent to pull and acknowledge the messages
    resource "google_pubsub_topic_iam_binding" "data_flow_service_account_access_subscriber" {
        project = module.project.project_id
        topic = module.logging_pubsub_topic.topic
        role = "roles/pubsub.subscriber"
        members = [
            "serviceAccount:$${google_service_account.data_flow_job_service_account.email}",
            "serviceAccount:service-${data.google_project.project_number.number}@dataflow-service-producer-prod.iam.gserviceaccount.com",
        ]
    }
```

3.6.3 Three-tier workload project deployments

The “three-tier-workload” template includes code to create resources in the three-tier workload project (Assured Workloads), assuming the project already exists. See section Deployment Phases to create Assured Workloads.

```
template "three-tier-workload" {
 recipe_path =
"git://github.com/GoogleCloudPlatform/healthcare-data-
protection-suite//templates/tfengine/recipes/project.hcl"
 output_path = "./threetierworkload/network"
 data = {
   project = {
     project_id = "{{.ttw_project_id}}"
     exists     = true
 }
}
```

This code enables a list of APIs required by the three-tier workload project (Assured Workloads) before deploying resources.

```
apis = [
           "compute.googleapis.com",
           "iam.googleapis.com",
           "sql-component.googleapis.com",
           "sqladmin.googleapis.com",
           "cloudkms.googleapis.com",
           "bigquery.googleapis.com",
           "bigqueryconnection.googleapis.com",
           "bigquerydatatransfer.googleapis.com",
"bigqueryreservation.googleapis.com",
           "bigquerystorage.googleapis.com",
           "dns.googleapis.com",
           "secretmanager.googleapis.com",
           "dlp.googleapis.com",
"servicenetworking.googleapis.com",
           "container.googleapis.com",
           "pubsub.googleapis.com",
           "dataflow.googleapis.com",
            "logging.googleapis.com",
           "monitoring.googleapis.com"
     ]
```

In this code, a VPC with two subnets, one for the web server managed instance group, and the GKE cluster is created.

Private Google access and private service access (for Cloud SQL) are created.

```
compute_networks =  [{

           name = "{{.vpc_network_name}}"
           resource_name = "ttw_network"
           # Enabling private Service access
           cloud_sql_private_service_access = {}
           subnets = [
               {
               name="{{.web_subnet_name}}"
               compute_region="{{.ttw_region}}"
               ip_range="{{.web_subnet_ip_range}}"

               },

               {
               name = "{{.gke_subnet_name}}"
               compute_region = "{{.ttw_region}}"
               ip_range = "{{.gke_subnet_primary_ip_range}}"
               secondary_ranges = [
                   {
                   name = "gke-subnet-secondary-pod-range"
                   ip_range =
"{{.gke_subnet_secondary_pod_ip_range}}"
                   },
                   {
                   name =
"gke-subnet-secondary-service-range"
                   ip_range =
"{{.gke_subnet_secondary_service_ip_range}}"
                   }

                 ]
               }
           ]
       }]
```

In this code, Cloud audit logs are enabled on all services in the three-tier workload project.

User can remove any of the Cloud audit logs (DATA_READ, DATA_WRITE, ADMIN_READ) if they are not required.

```
#Code Block 3.2.2.c
       resource "google_project_iam_audit_config" "project"
{
           project = "{{.ttw_project_id}}"
           service = "allServices"
           audit_log_config {
               log_type = "DATA_READ"
           }
           audit_log_config {
               log_type = "DATA_WRITE"
           }
           audit_log_config {
               log_type = "ADMIN_READ"
           }
       }
```

In this code, a private Cloud SQL instance is created, and access is authorized from the three-tier workload network. Failover replica is created and automated backups are enabled by default.

User can customize the creation of Cloud SQL DBuser and DBpassword using the “secrets manager” template. By default, custom DBuser and DBpassword are disabled as the Secrets Manager is not FedRAMP compliant.

```
cloud_sql_instances = [{
           name                 =
"{{.private_cloud_sql_name}}"
           resource_name      = "ttw_sql_instance"
           type               = "mysql"
           network_project_id = "{{.ttw_project_id}}"
           network            = "{{.vpc_network_name}}"
           tier               =
"{{.private_cloud_sql_machine_type}}"
           labels = {
               component = "database"
               data_type =
"{{.mig_instance_datatype_label}}"
               data_criticality =
"{{.mig_instance_data_criticality_label}}"
           }
           # At the time of this writing, Secret Manager is
not FedRAMP compliant. The following code creates a default
user and password.
           #user_name        = "user1"
           #user_password     =
"$${data.google_secret_manager_secret_version.db_password.
secret_data}"
```

In this code, a private GKE cluster is created. User can add a master authorized IP range to allow access to the private cluster.

Node pool block can be removed and a customized pool as per user requirement can be added.

A custom service account is created by the safer GKE module used in the Data Protection Toolkit.

User can create an internal load balancer listening from the web server subnet by deploying a service with the load balancer annotation type "[internal](https://cloud.google.com/kubernetes-engine/docs/how-to/internal-load-balancing)". GKE deployments are not created in this template.

```
gke_clusters = [{
           name                   =
"{{.gke_private_cluster_name}}"
           resource_name          = "ttw_gke-cluster"
           network_project_id     = "{{.ttw_project_id}}"
           network                = "{{.vpc_network_name}}"
 subnet                 = "{{.gke_subnet_name}}"
           ip_range_pods_name     =
"gke-subnet-secondary-pod-range"
           ip_range_services_name =
"gke-subnet-secondary-service-range"
           master_ipv4_cidr_block =
"{{.gke_private_master_ip_range}}"
#Code Block 3.2.6.c
           node_pools = [
               {
               name              = "{{.gke_node_pool_name}}"
               machine_type      =
"{{.gke_node_pool_machine_type}}"
               min_count         =
{{.gke_node_pool_min_instance_count}}
               max_count         =
{{.gke_node_pool_max_instance_count}}
               disk_size_gb      =
{{.gke_node_pool_instance_disk_size}}
               #Uncomment the following parameters to
customize node pool
               #disk_type         =
               #accelerator_count =
               #accelerator_type  =
               image_type        =
"{{.gke_node_pool_image_type}}"
               #uncomment to enable auto_repair. Default
is false
               #auto_repair       = true
               auto_upgrade    = true
              }
           ]
#Code Block 3.2.6.b (1)
     master_authorized_networks = [
           {
               cidr_block = "{{.web_subnet_ip_range}}"
               display_name = "web-subnet"

           }
#Code Block 3.2.6.b (2)
           # Uncomment to whitelist additional IPs.
          #{
               #cidr_block = ""
               #display_name = ""
          #}
           ]
#Code Block 3.2.6.d
           labels = {
               component = "application-server"
               data_type =
"{{.mig_instance_datatype_label}}"
               data_criticality =
"{{.mig_instance_data_criticality_label}}"
           }
       }]
```

<table>
<tr>
<td>

In this code, a Cloud NAT and a Cloud Router are created, which are used by the web server managed instance group and GKE cluster nodes to reach the internet.

Cloud NAT translates both primary and secondary ranges. Can be customized to allow either primary or secondary ranges only.

</td>
<td>

```
compute_routers = [{
           name    = "ttw-fedramp-router"
           resource_name = "ttw_fedramp_router"
           network = "{{.vpc_network_name}}"
           nats = [{
               name = "ttw-webserver-router-nat"
               source_subnetwork_ip_ranges_to_nat = "LIST_OF_SUBNETWORKS"
               subnetworks = [
                   {
                   name = "{{.web_subnet_name}}"
                   source_ip_ranges_to_nat  = ["ALL_IP_RANGES"]
                   },
                   {
                   name = "{{.gke_subnet_name}}"
                   source_ip_ranges_to_nat  = ["ALL_IP_RANGES"]
                   },
               ]
           }]
       }]
```

</td>
</tr>
<tr>
<td>

In this code, a service account required for the web server managed instance group is created.

</td>
<td>

```
service_accounts = [
           {
           account_id   = "web-server-service-account"
           resource_name = "web_server_service_account"
           description  = "web Service Account"
           display_name = "web Service Account"
           }
       ]
```

</td>
</tr>
</table>

In this code a Cloud Armor security policy is created.

Cloud Armor security policy can be configured to evaluate multiple pre-configured expressions.

```
resource "google_compute_security_policy" "policy" {
           name =
"{{.cloud_armor_security_policy_name}}cloud-armor-security-policy"
           # Default rule to deny traffic from internet rule {
               action   = "deny(403)"
               priority = "2147483647"
               match {
                   versioned_expr = "SRC_IPS_V1"
                   config {
                       src_ip_ranges = ["*"]
                   }
               }
               description = "default rule"
           }
           #user can configure rules to evaluate more
preconfigured expressions
           rule {
               action   = "deny(403)"
               priority = "1000"
               match {
                   expr {
                       expression=
"evaluatePreconfiguredExpr('xss-canary')"
                   }
               }
               description = "Deny access to XSS attempts"
           }
           # Custom rule to allow specific IPs (allow listing)
           rule {
               action   = "allow"
               priority = "500"
               match {
                   versioned_expr = "SRC_IPS_V1"
                   config {
                       src_ip_ranges = [
"{{.cloud_armor_security_policy_allow_range}}"
                       ]
                   }
               }
               description = "allow only from Specific range"
           }
       }
```

In this code, three resources are created: the instance template, the managed instance group with autocaler, and the health check.

User can customize the instance template to use custom images that support Confidential Compute).

Startup script can be changed inline as per requirement.

The machine type must be from the family N2D to support Confidential Compute.

User can choose to use CMEK instead of Google managed keys.

The basic health check block is used in the template. User can customize the health check with complex request response patterns.

```
resource "google_compute_instance_template"
"ttw_webserver_instance_template_region_1" {
           name = "{{.ttw_instance_template_name}}"
           description = "template for DPT FedRAMP"
           region = "{{.ttw_region}}"
           tags =
["ttw-webserver","ttw-health-check"]
           metadata_startup_script = <<SCRIPT
               sudo apt-get -y update
               apt-get install -y apache2 php
               sudo apt-get -y install mysql-client
               SCRIPT
#Code Block 3.2.4.d
           labels = {
             component = "webserver"
             data_type = "{{.mig_instance_datatype_label}}"
             data_criticality =
"{{.mig_instance_data_criticality_label}}"
           }
          # Machine type should support Confidential Compute.
Use N2D type instances.
           machine_type  = "{{.mig_instance_type}}"
           can_ip_forward = false
           scheduling {
             automatic_restart   = true
             on_host_maintenance = "TERMINATE"
           }
           disk {
             # This is a basic image, which supports
Confidential Compute.
             # For custom created images, provide an image path
as required. Custom image should support Confidential Compute
           source_image =
"projects/confidential-vm-images/global/images/ubuntu-1804-
bionic-v20201014"
             auto_delete = true
             boot        = true
             # disk size can be uncommented to customize.
             #disk_size_gb =
# Code Block 3.2.4.c
             # Provide the CMEK disk encryption key self link
in the block below
```

```
             #disk_encryption_key {
             #     kms_key_self_link =
             #}


           }
           network_interface {
             network = "{{.vpc_network_name}}"
             subnetwork = "{{.web_subnet_name}}"
           }
           service_account {
             email  =
"$${google_service_account.web_server_service_account.email}"
             scopes = ["cloud-platform"]
           }
           confidential_instance_config {

enable_confidential_compute = true
           }

       }
       #*****************Managed Instance Group With
Autoscaling**************************


       resource "google_compute_health_check"
"ttw-webserver-health-check" {

           name = "{{.ttw_compute_http_health_check_name}}"
           timeout_sec =
{{.ttw_compute_http_health_check_timeout_sec}}
           check_interval_sec =
{{.ttw_compute_http_health_check_interval_sec}}
           healthy_threshold =
{{.ttw_compute_http_health_check_healthy_threshold}}
           unhealthy_threshold =
{{.ttw_compute_http_health_check_unhealthy_threshold}}
           http_health_check {
                 #port_name =
                 #port_specification = "USE_NAMED_PORT"
                 port = 80
                 request_path =
"{{.ttw_compute_http_health_check_request_path}}"
         proxy_header =
"{{.ttw_compute_http_health_check_proxy_header}}"
         response =
"{{.ttw_compute_http_health_check_response}}"
      }
    }
    resource "google_compute_region_instance_group_manager"
"ttw-webserver-mig1" {
      name = "{{.mig_name}}"
      base_instance_name =  "webserver-mig1-instance"
      # Distribution policy defines in which zones the
instances have to be distributed.
      # User has to check regions and zones that support
N2D machine type (For Confidential Compute)
https://cloud.google.com/compute/docs/regions-zones#available

      distribution_policy_zones  =
{{.mig_distribution_policy_zones}}
      depends_on =
"$${[google_compute_instance_template.ttw_webserver_instance_
template_region_1]}"
      version {
        instance_template =
"$${google_compute_instance_template.ttw_webserver_instance_
template_region_1.id}"
 }
      region = "{{.ttw_region}}"
      project = "{{.ttw_project_id}}"
      auto_healing_policies {
        health_check =
"$${google_compute_health_check.ttw-webserver-health-check.id}"
        initial_delay_sec = 300
        # This is the port number that your backend
instances (Compute Engine instances) are listening on
        # If not specified, instances will listen on
the same port as the load balancer listening port.
        #named_port:
         #name: "customHTTP"
         #port: 8080
      }
    }
```

```
#Code Block 3.2.4.b
       resource "google_compute_region_autoscaler"
"ttw-webserver-autoscaler" {
           name = "ttw-webserver-autoscaler-1"
           region = "{{.ttw_region}}"
           target =
"$${google_compute_region_instance_group_manager.ttw-webserver-
mig1.id}"
           autoscaling_policy {
               max_replicas = {{.autoscaling_max_replicas}}
               min_replicas = {{.autoscaling_min_replicas}}
               cooldown_period = {{.autoscaling_cooldown_period}}
               cpu_utilization {
                 target = {{.autoscaling_cpu_utilization}}

                 # uncomment to use autoscaling metric instead
of CPU utilization
                 #metric{

                 #}
               }
           }
       }
```

In this code, three resources are created: the HTTPS global load balancer, the Google managed SSL certificate, and the Cloud DNS zone.

User can customize to use a self-managed certificate. Refer to the product guidance section.

A basic URL map is used in this template. User can customize it to use more complex URL maps.

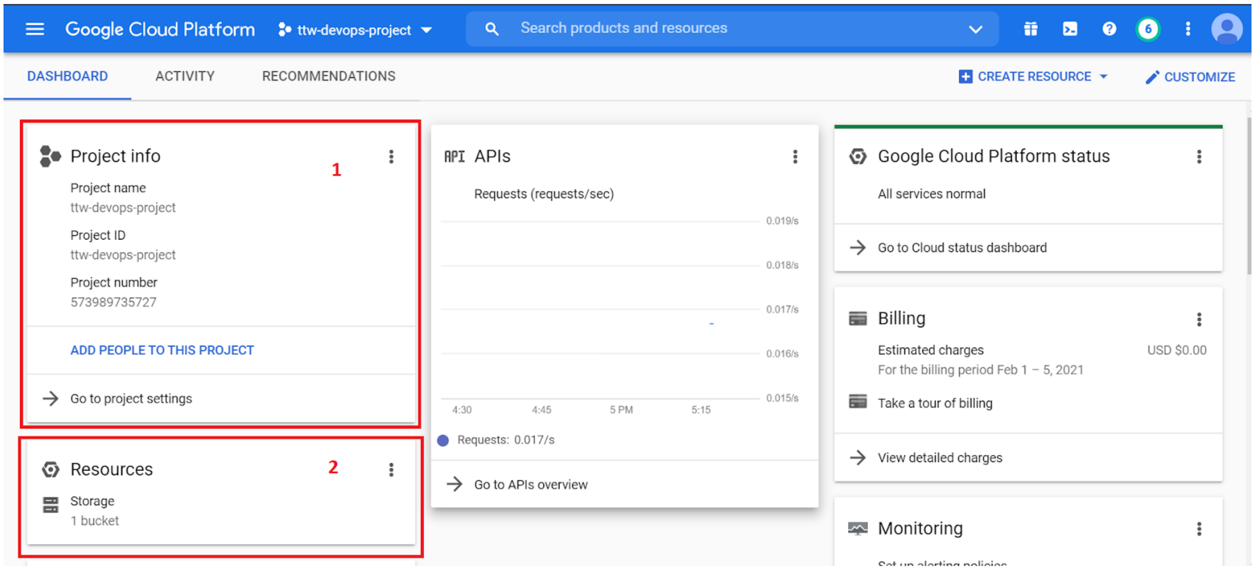
User have to change the NS records of the domain registrar to match the custom NS records created by the Cloud DNS zone. Add the user who is deploying this workload as the owner of the domain or verify the domain at http://www.google.com/webmasters/verification/.

```
resource "google_compute_managed_ssl_certificate"
"ttw-ssl-certificate" {
               name = "ttw-cert"
               managed {
                   domains =
["{{.load_balancer_ssl_certificate_domain_name}}"]
               }
       }

       #resource "google_compute_ssl_certificate" "default" {
       #    name_prefix = "my-certificate-"
       #   description = "a description"
       #   private_key = file("path/to/private.key")
       #    certificate = file("path/to/certificate.crt")
       #
       #    lifecycle {
       #        create_before_destroy = true
       #    }
       #}
       resource "google_compute_global_forwarding_rule"
"ttw-https-global-forwarding-rule"{
               name = "ttw-https-forwarding-rule"
               target =
"$${google_compute_target_https_proxy.ttw-https-target-proxy.id}"
               port_range = "443"
               # Ephemeral IP address will be auto created.
Uncomment and provide self link of custom external IP
               # ip_address =
               load_balancing_scheme =
"EXTERNAL"

       }
       resource "google_compute_target_https_proxy"
"ttw-https-target-proxy" {
               name = "ttw-target-proxy"
               url_map =
"$${google_compute_url_map.ttw-url-map.id}"
#Code Block 3.2.8.b (2)

#If self-managed certificate is used, change the reference in
the below SSL parameter
               ssl_certificates =
"$${[google_compute_managed_ssl_certificate.ttw-ssl-certificate.
id]}"
       }

       resource "google_compute_url_map" "ttw-url-map" {
               name = "ttw-url-target-proxy"
               default_service =
"$${google_compute_backend_service.ttw-backend-service-1.id}"
               host_rule {
                 hosts = ["{{.load_balancer_url_map_host}}"]
                 path_matcher = "backendpath"
               }
               path_matcher{
                 name = "backendpath"
                 default_service =
"$${google_compute_backend_service.ttw-backend-service-1.id}"
                 path_rule {
                       paths   =
["{{.load_balancer_url_map_compute_backend_path}}"]
                       service =
"$${google_compute_backend_service.ttw-backend-service-1.id}"
                   }
                   path_rule {
                       paths   =
["{{.load_balancer_url_map_bucket_backend_path}}"]
                       service =
"$${google_compute_backend_bucket.ttw-static-website.id}"
                   }
               }
       }

       resource "google_compute_backend_service"
"ttw-backend-service-1" {
               name = "ttw-regional-backend-service"
               backend {
                 group          =
"$${google_compute_region_instance_group_manager.ttw-webserver-
mig1.instance_group}"
                 balancing_mode = "UTILIZATION"
                 capacity_scaler = 1.0
               }
               protocol = "{{.backend_mig_protocol}}"
               #seconds to wait for the backend before
considering it a failed request. Default is 30 seconds.
               timeout_sec = {{.backend_mig_timeout}}
               security_policy =
google_compute_security_policy.policy.self_link
               #enable_cdn  = false
               health_checks
="$${[google_compute_health_check.ttw-webserver-health-check.id]}
"
       }

       resource "google_compute_backend_bucket"
"ttw-static-website" {
           name        = "static-bucket-backend"
           description = "Contains static files"
           bucket_name =
"$${module.ttw_static_files_bucket.bucket.name}"
           enable_cdn  = true
           depends_on = "$${[module.ttw_static_files_bucket]}"
       }
#Code Block 3.2.8.b (3)
#Customize this code block when using a self-managed certificate
       resource "google_dns_record_set" "set" {
               name         =
"$${google_dns_managed_zone.ttw-zone.dns_name}"
               type         = "A"
               ttl          = 3600
               managed_zone =
"$${google_dns_managed_zone.ttw-zone.name}"
               rrdatas      =
"$${[google_compute_global_forwarding_rule.ttw-https-global-
forwarding-rule.ip_address]}"
       }
       resource "google_dns_managed_zone" "ttw-zone" {
               name     = "ttw-zone"
               dns_name =
"{{.load_balancer_ssl_certificate_domain_name}}"
       }
```

3.7 Post-deployment verification

3.7.1 DevOps project

Cloud Console Dashboard: Shows the list of deployed projects. Under the DevOps project, the project information and the resources that are deployed using the template are listed (Figure 2).



Project info

Project name
ttw-devops-project

Project ID
ttw-devops-project

Project number
573989735727

Resources

Storage
1 bucket

APIs

Requests (requests/sec)

Requests: 0.017/s

Google Cloud Platform status

All services normal

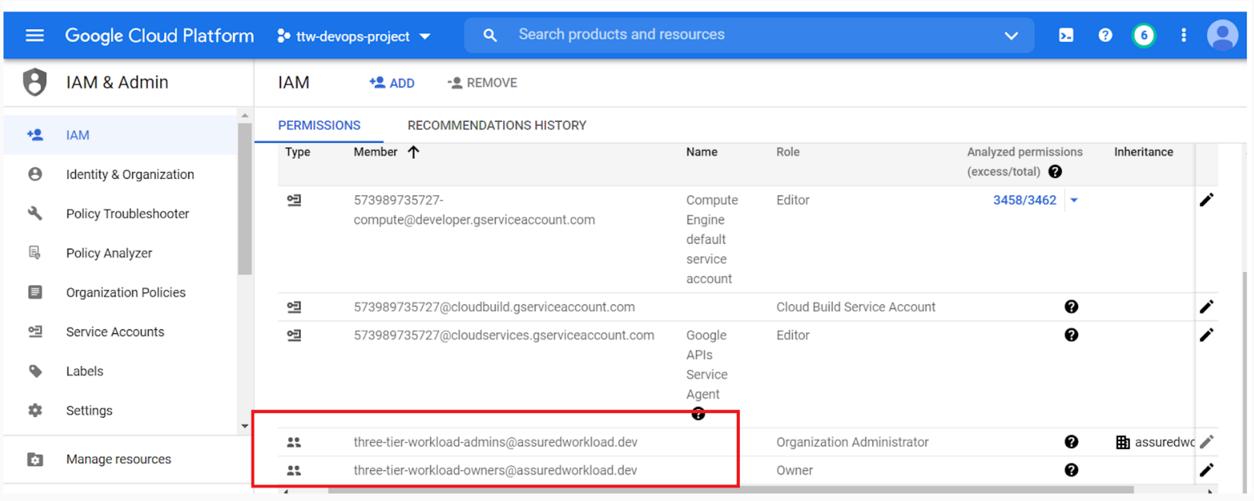
Billing

Estimated charges USD $0.00
For the billing period Feb 1 – 5, 2021

Monitoring

Figure 2. 1: DevOps project information 2: Project resources.

IAM Console: Shows the IAM permissions and the admins and owners groups (Figure 3).

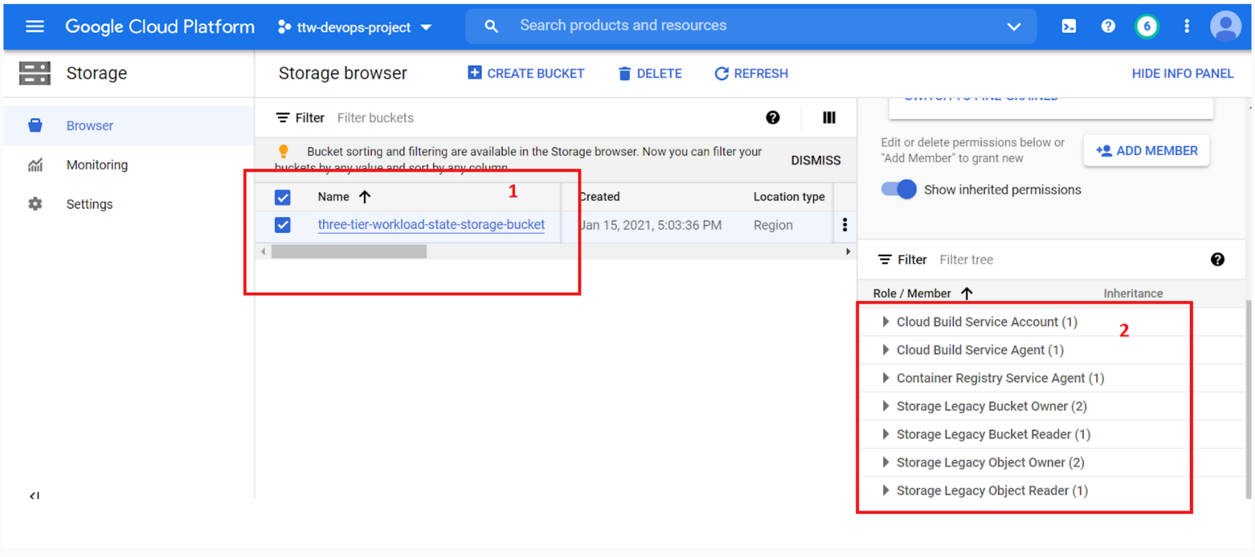


IAM

Type	Member	Name	Role	Analyzed permissions (excess/total)	Inheritance
	573989735727-compute@developer.gserviceaccount.com	Compute Engine default service account	Editor	3458/3462	
	573989735727@cloudbuild.gserviceaccount.com		Cloud Build Service Account		
	573989735727@cloudservices.gserviceaccount.com	Google APIs Service Agent	Editor		
	three-tier-workload-admins@assuredworkload.dev		Organization Administrator		assuredwc
	three-tier-workload-owners@assuredworkload.dev		Owner		

Figure 3. Owners and Admin group permissions.

Cloud Storage Console: Shows the Terraform state storage bucket that is deployed through the template (Figure 4). User can view the storage bucket and its corresponding permissions by selecting the respective buckets in the list.



Bucket sorting and filtering are available in the Storage browser. Now you can filter your buckets by any value and sort by any column.

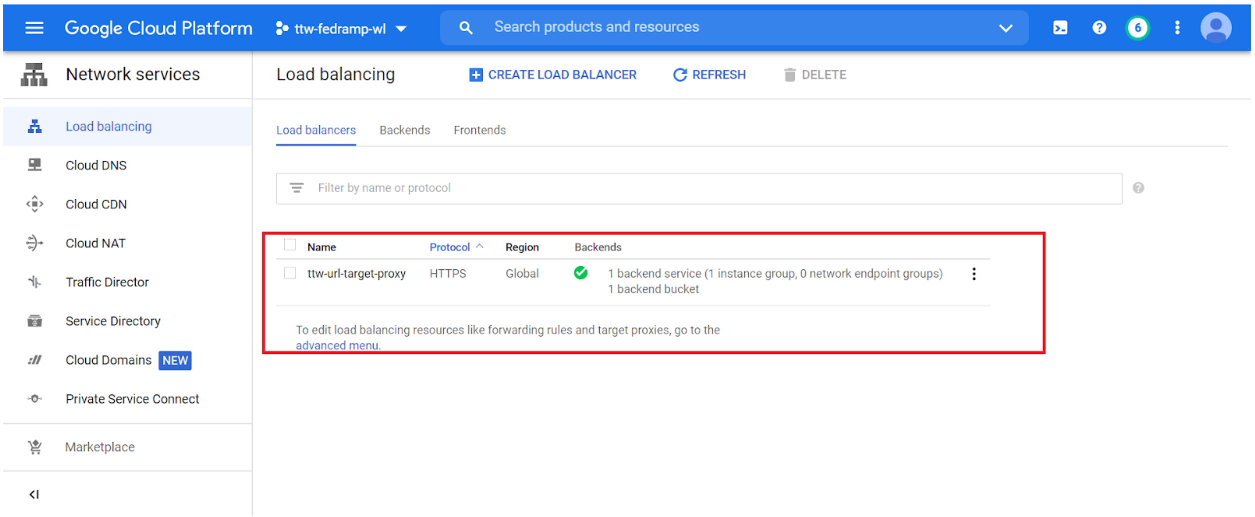
☑	Name	Created	Location type
☑	three-tier-workload-state-storage-bucket	Jan 15, 2021, 5:03:36 PM	Region

Role / Member	Inheritance
Cloud Build Service Account (1)	
Cloud Build Service Agent (1)	
Container Registry Service Agent (1)	
Storage Legacy Bucket Owner (2)	
Storage Legacy Bucket Reader (1)	
Storage Legacy Object Owner (2)	
Storage Legacy Object Reader (1)	

Figure 4. 1: State storage bucket 2: Bucket permissions.

3.7.2 Three-tier workload project

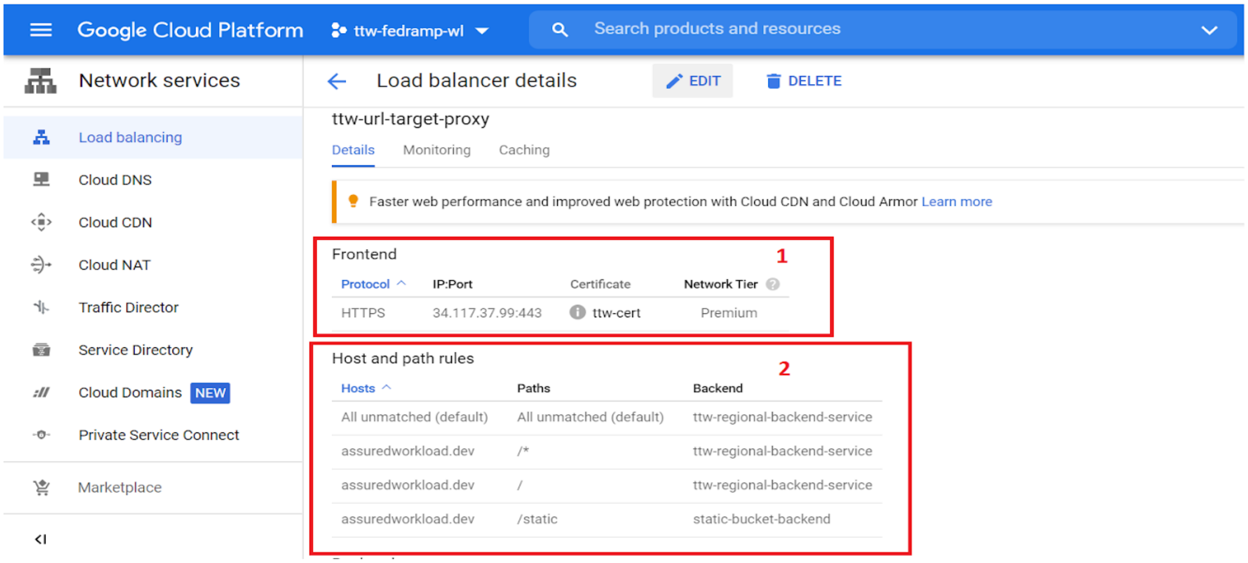
Network Services - Load Balancer Console: Shows the HTTPS external load balancer (Figure 5) with the Google managed frontend SSL certificate (Figure 6) and backend services (Figure 7).



□	Name	Protocol	Region	Backends
□	ttw-url-target-proxy	HTTPS	Global	✓ 1 backend service (1 instance group, 0 network endpoint groups) 1 backend bucket

To edit load balancing resources like forwarding rules and target proxies, go to the advanced menu.

Figure 5. HTTPS load balancer.



ttw-url-target-proxy

Faster web performance and improved web protection with Cloud CDN and Cloud Armor Learn more

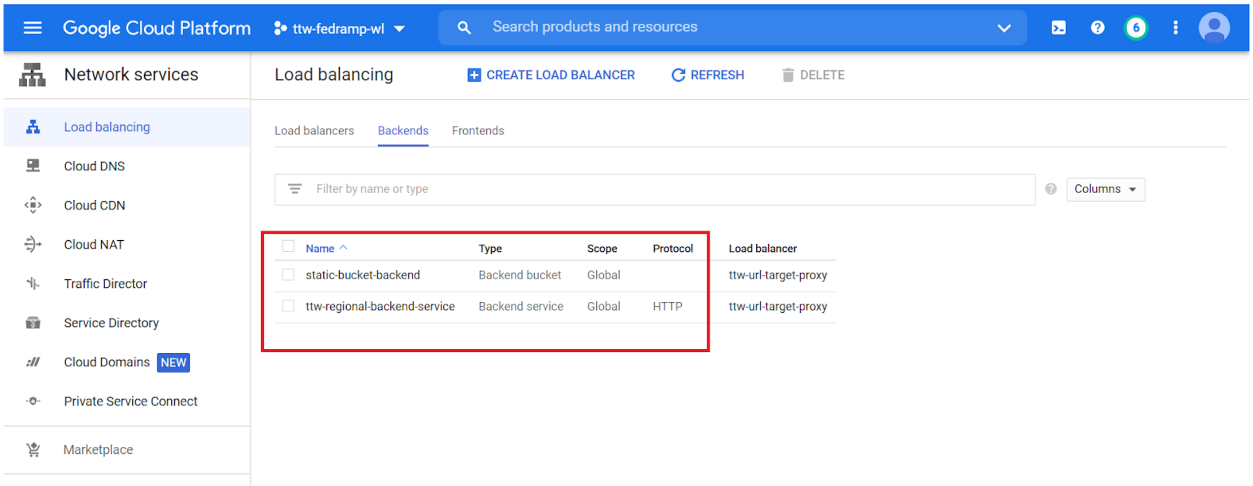
Frontend

Protocol	IP:Port	Certificate	Network Tier
HTTPS	34.117.37.99:443	ttw-cert	Premium

Host and path rules

Hosts	Paths	Backend
All unmatched (default)	All unmatched (default)	ttw-regional-backend-service
assuredworkload.dev	/*	ttw-regional-backend-service
assuredworkload.dev	/	ttw-regional-backend-service
assuredworkload.dev	/static	static-bucket-backend

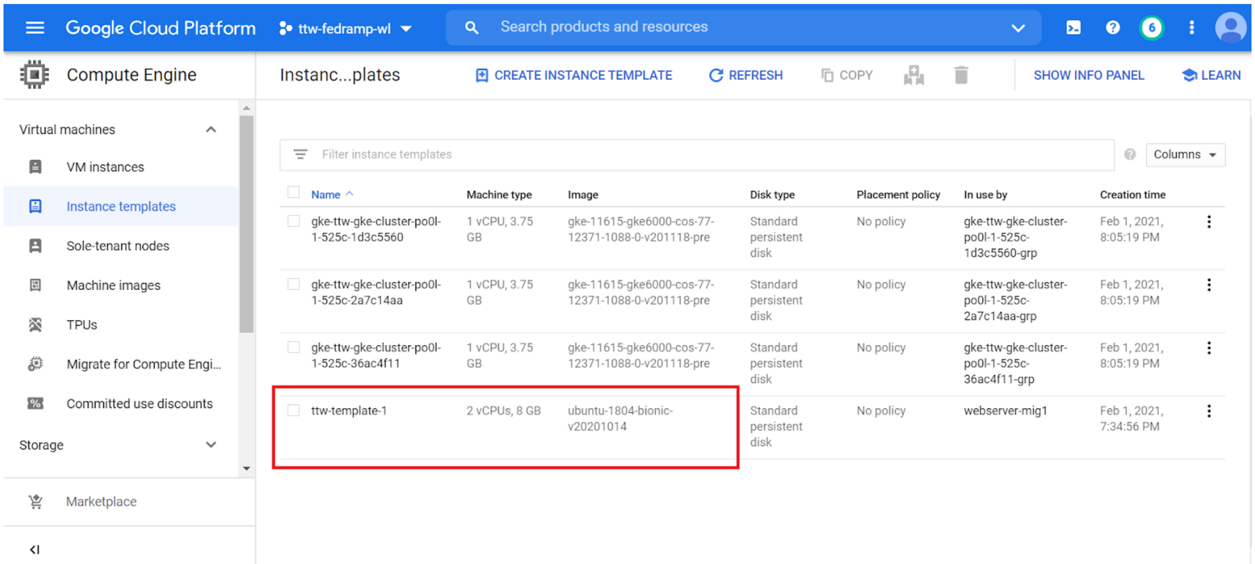
Figure 6. 1: Frontend SSL certificate 2: URL path.



Name	Type	Scope	Protocol	Load balancer
static-bucket-backend	Backend bucket	Global		ttw-url-target-proxy
ttw-regional-backend-service	Backend service	Global	HTTP	ttw-url-target-proxy

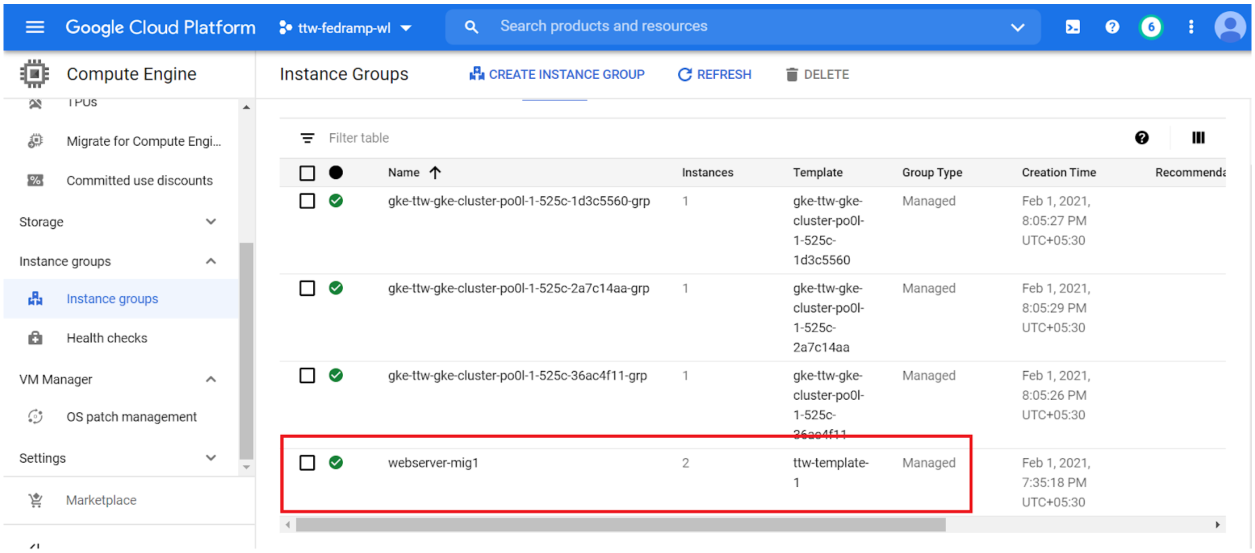
Figure 7. Backend services.

Compute Engine Console: Shows the Instance template (Figure 8), and the Instance group manager with auto scaling (Figure 9).



	Name	Machine type	Image	Disk type	Placement policy	In use by	Creation time
□	gke-ttw-gke-cluster-po0l-1-525c-1d3c5560	1 vCPU, 3.75 GB	gke-11615-gke6000-cos-77-12371-1088-0-v201118-pre	Standard persistent disk	No policy	gke-ttw-gke-cluster-po0l-1-525c-1d3c5560-grp	Feb 1, 2021, 8:05:19 PM
□	gke-ttw-gke-cluster-po0l-1-525c-2a7c14aa	1 vCPU, 3.75 GB	gke-11615-gke6000-cos-77-12371-1088-0-v201118-pre	Standard persistent disk	No policy	gke-ttw-gke-cluster-po0l-1-525c-2a7c14aa-grp	Feb 1, 2021, 8:05:19 PM
□	gke-ttw-gke-cluster-po0l-1-525c-36ac4f11	1 vCPU, 3.75 GB	gke-11615-gke6000-cos-77-12371-1088-0-v201118-pre	Standard persistent disk	No policy	gke-ttw-gke-cluster-po0l-1-525c-36ac4f11-grp	Feb 1, 2021, 8:05:19 PM
□	ttw-template-1	2 vCPUs, 8 GB	ubuntu-1804-bionic-v20201014	Standard persistent disk	No policy	webserver-mig1	Feb 1, 2021, 7:34:56 PM

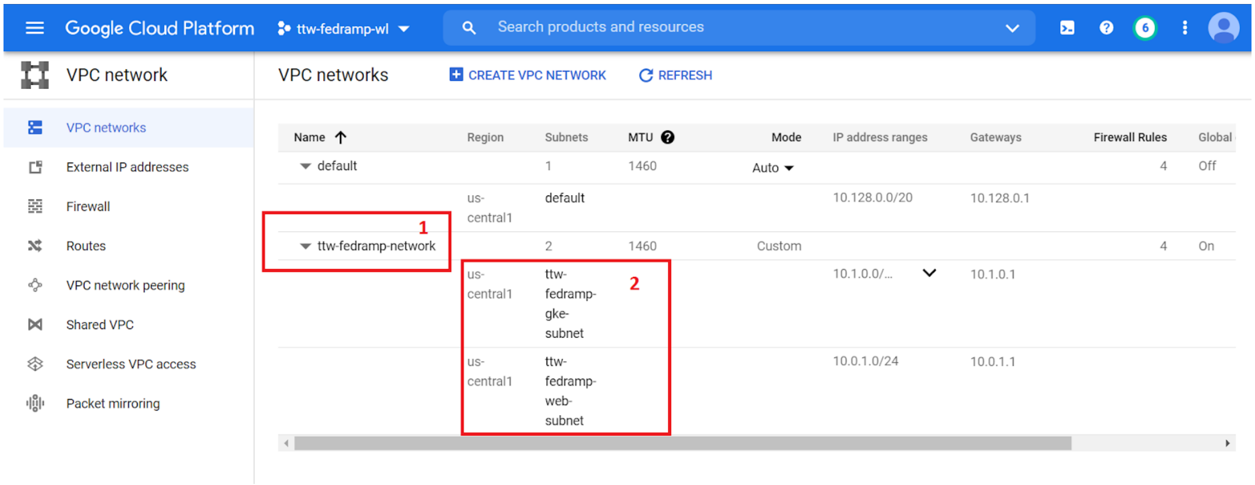
Figure 8. Instance template.



□	●	Name ↑	Instances	Template	Group Type	Creation Time	Recommenda
□	✓	gke-ttw-gke-cluster-po0l-1-525c-1d3c5560-grp	1	gke-ttw-gke-cluster-po0l-1-525c-1d3c5560	Managed	Feb 1, 2021, 8:05:27 PM UTC+05:30	
□	✓	gke-ttw-gke-cluster-po0l-1-525c-2a7c14aa-grp	1	gke-ttw-gke-cluster-po0l-1-525c-2a7c14aa	Managed	Feb 1, 2021, 8:05:29 PM UTC+05:30	
□	✓	gke-ttw-gke-cluster-po0l-1-525c-36ac4f11-grp	1	gke-ttw-gke-cluster-po0l-1-525c-36ac4f11	Managed	Feb 1, 2021, 8:05:26 PM UTC+05:30	
□	✓	webserver-mig1	2	ttw-template-1	Managed	Feb 1, 2021, 7:35:18 PM UTC+05:30	

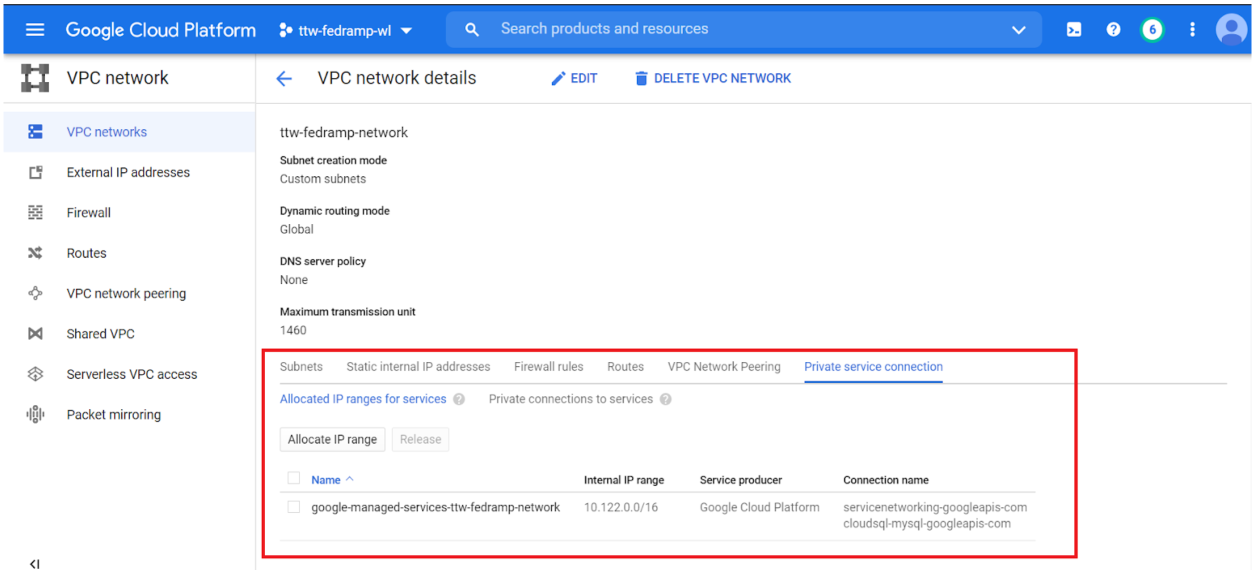
Figure 9. Instance group manager.

VPC Console: Shows the VPC that is created for the three -tier workload (Figure 10). A private service connection used by Cloud SQL is also shown in (Figure 11).



Name	Region	Subnets	MTU	Mode	IP address ranges	Gateways	Firewall Rules	Global
default		1	1460	Auto			4	Off
	us-central1	default			10.128.0.0/20	10.128.0.1		
ttw-fedramp-network		2	1460	Custom			4	On
	us-central1	ttw-fedramp-gke-subnet			10.1.0.0/...	10.1.0.1		
	us-central1	ttw-fedramp-web-subnet			10.0.1.0/24	10.0.1.1		

Figure 10. 1: VPC 2: Subnets.



ttw-fedramp-network

Subnet creation mode
Custom subnets

Dynamic routing mode
Global

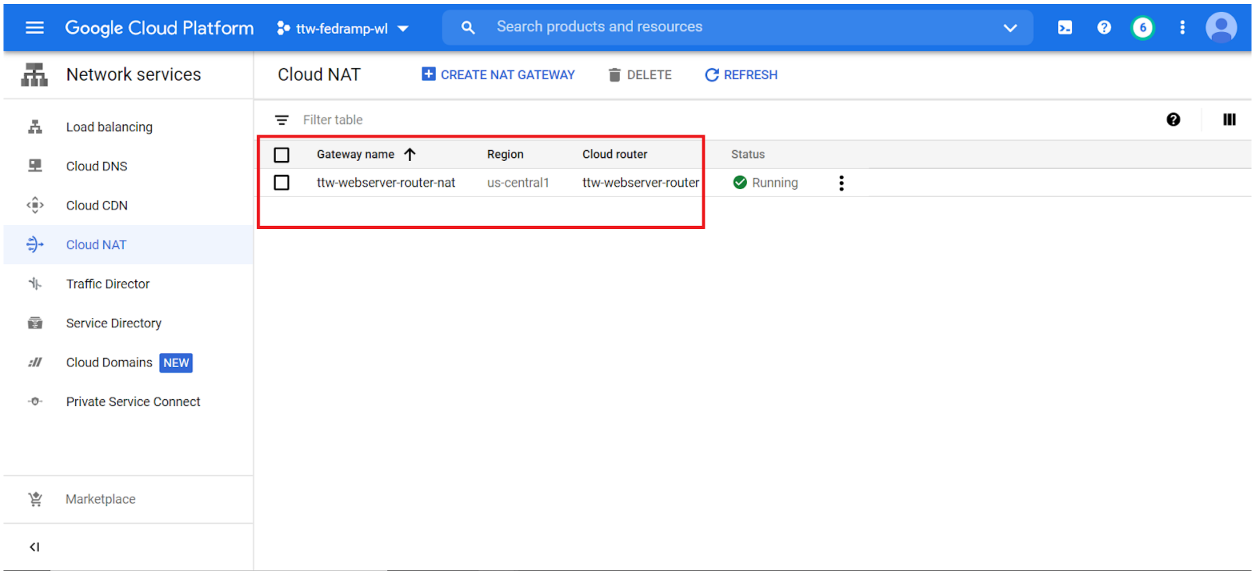
DNS server policy
None

Maximum transmission unit
1460

Name	Internal IP range	Service producer	Connection name
google-managed-services-ttw-fedramp-network	10.122.0.0/16	Google Cloud Platform	servicenetworking-googleapis-com cloudsql-mysql-googleapis-com

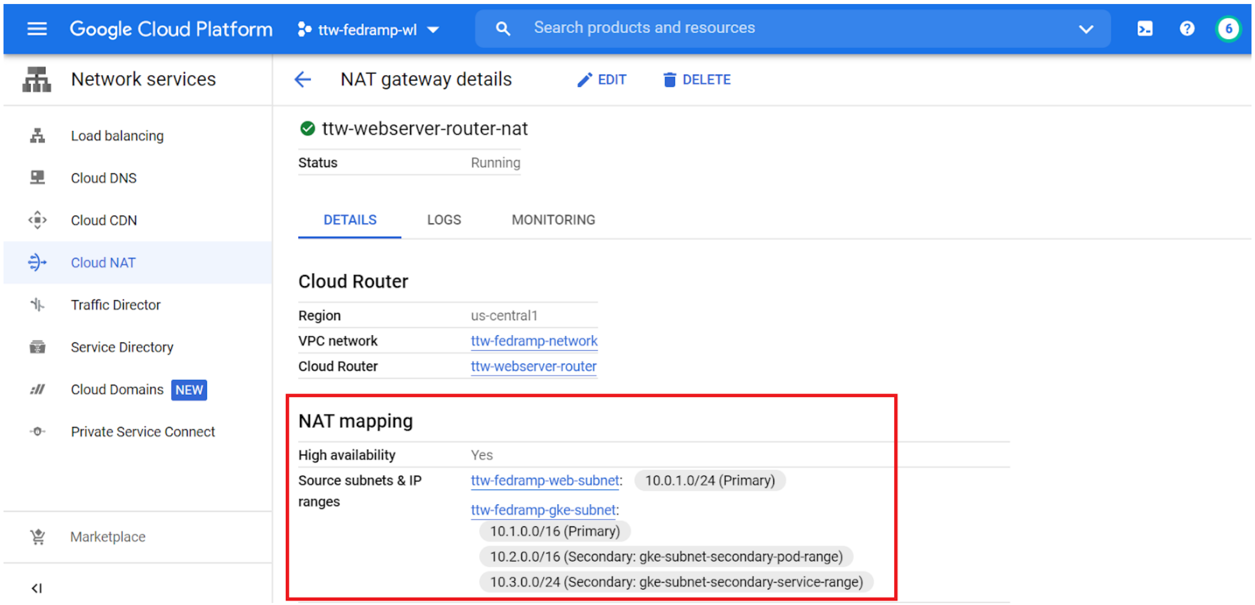
Figure 11. Private service connection.

Network Services Console: Shows the Cloud NAT with a Cloud Router deployed by the template (Figure 12).



□	Gateway name ↑	Region	Cloud router	Status
□	ttw-webserver-router-nat	us-central1	ttw-webserver-router	✓ Running

Figure 12. Cloud NAT.



✓ ttw-webserver-router-nat

Status Running

Cloud Router

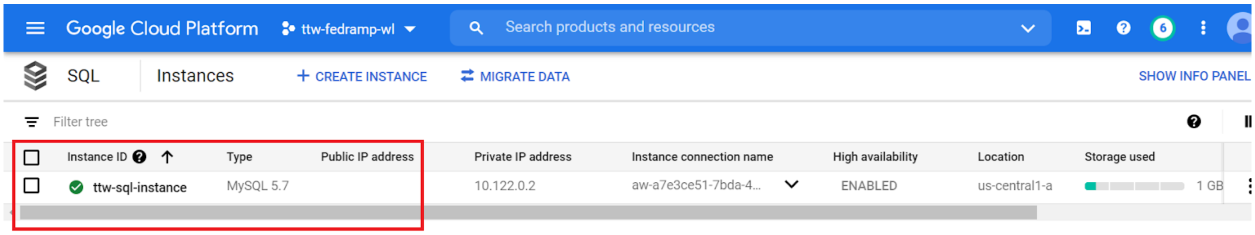
Region	us-central1
VPC network	ttw-fedramp-network
Cloud Router	ttw-webserver-router

NAT mapping

High availability	Yes
Source subnets & IP ranges	ttw-fedramp-web-subnet: 10.0.1.0/24 (Primary) ttw-fedramp-gke-subnet: 10.1.0.0/16 (Primary) 10.2.0.0/16 (Secondary: gke-subnet-secondary-pod-range) 10.3.0.0/24 (Secondary: gke-subnet-secondary-service-range)

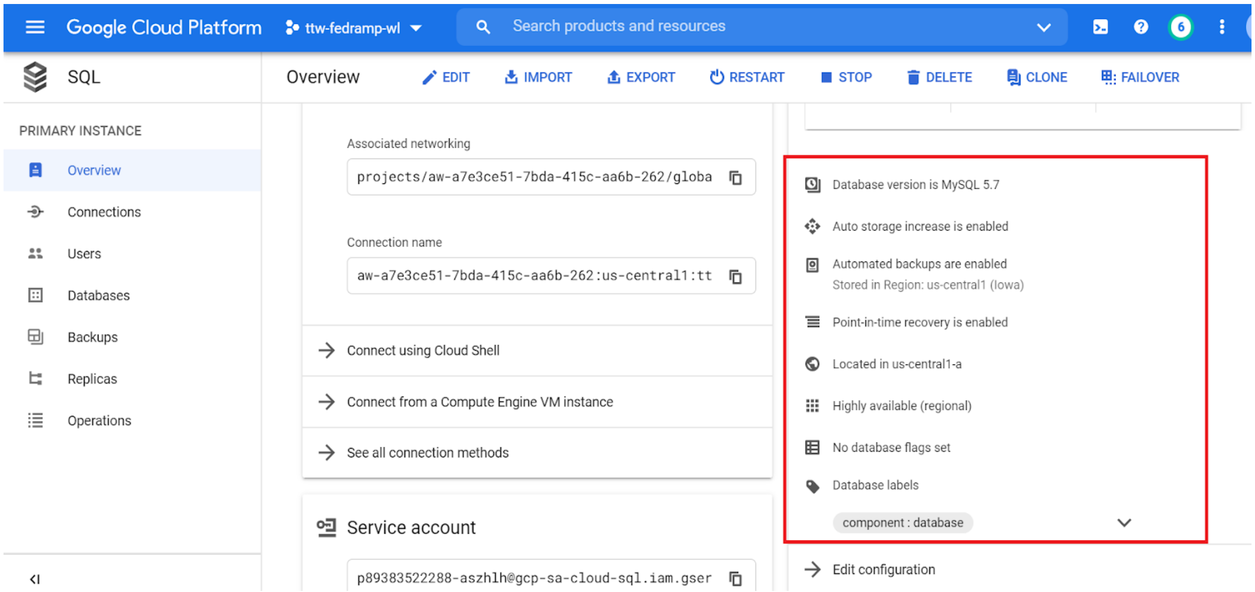
Figure 13. NAT mapping

Cloud SQL Console: Shows the private Cloud SQL instance that is created by the template (Figure 14).



Instance ID	Type	Public IP address	Private IP address	Instance connection name	High availability	Location	Storage used
ttw-sql-instance	MySQL 5.7		10.122.0.2	aw-a7e3ce51-7bda-4...	ENABLED	us-central1-a	1 GB

Figure 14. Cloud SQL instance.



Associated networking
projects/aw-a7e3ce51-7bda-415c-aa6b-262/globa

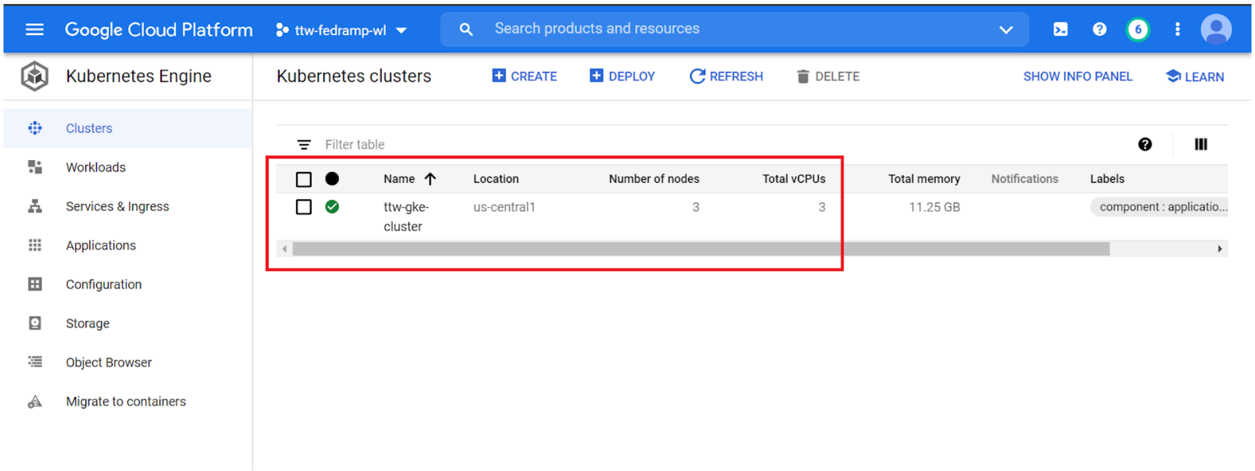
Connection name
aw-a7e3ce51-7bda-415c-aa6b-262:us-central1:tt

Service account
p89383522288-aszhlh@gcp-sa-cloud-sql.iam.gser

- Database version is MySQL 5.7
- Auto storage increase is enabled
- Automated backups are enabled
  Stored in Region: us-central1 (Iowa)
- Point-in-time recovery is enabled
- Located in us-central1-a
- Highly available (regional)
- No database flags set
- Database labels
  component : database

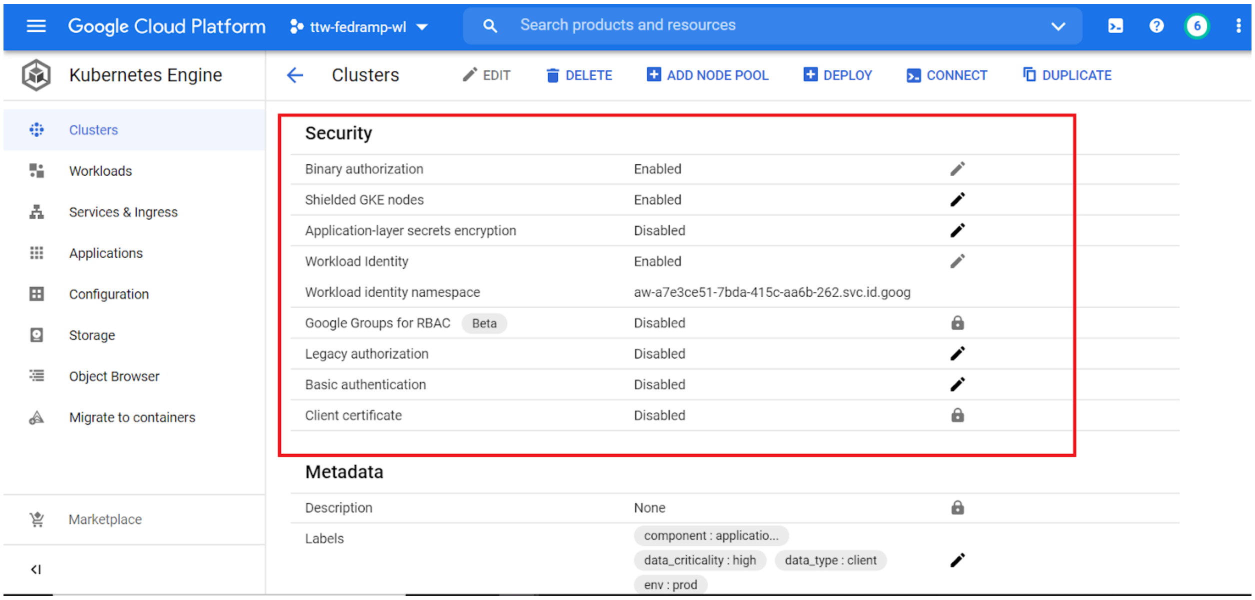
Figure 15. Cloud SQL enabled configurations.

Google Kubernetes Engine Console: Shows the private GKE cluster that is created by the template.



□	●	Name	Location	Number of nodes	Total vCPUs	Total memory	Notifications	Labels
□	✓	ttw-gke-cluster	us-central1	3	3	11.25 GB		component : applicatio...

Figure 16. GKE Cluster.



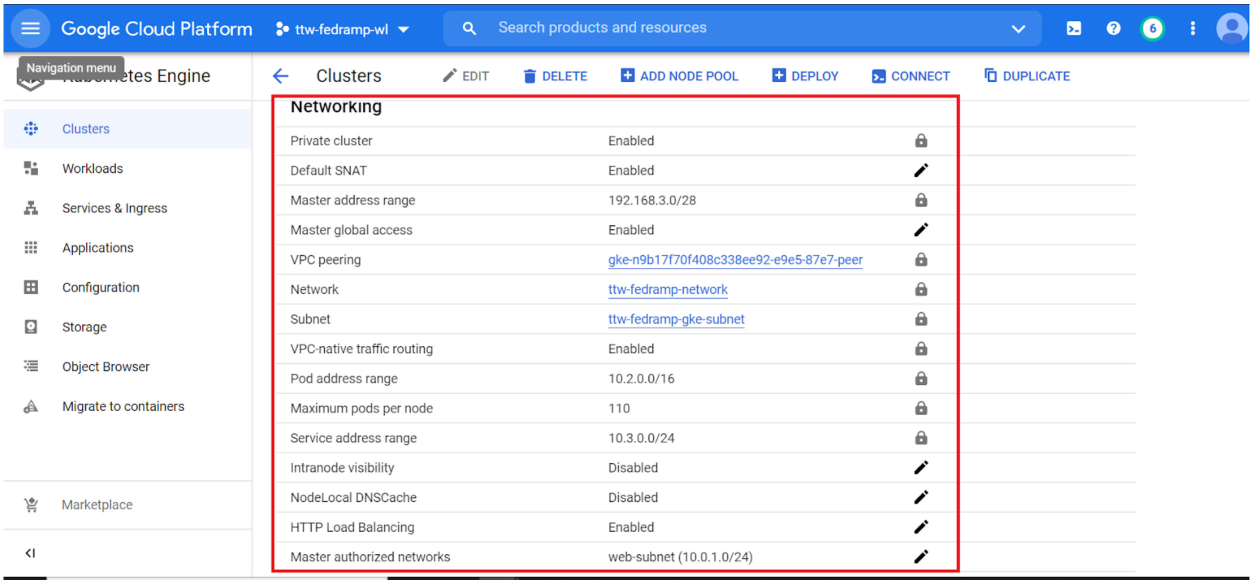
Security

Binary authorization	Enabled
Shielded GKE nodes	Enabled
Application-layer secrets encryption	Disabled
Workload Identity	Enabled
Workload identity namespace	aw-a7e3ce51-7bda-415c-aa6b-262.svc.id.goog
Google Groups for RBAC Beta	Disabled
Legacy authorization	Disabled
Basic authentication	Disabled
Client certificate	Disabled

Metadata

Description	None
Labels	component : applicatio... data_criticality : high data_type : client env : prod

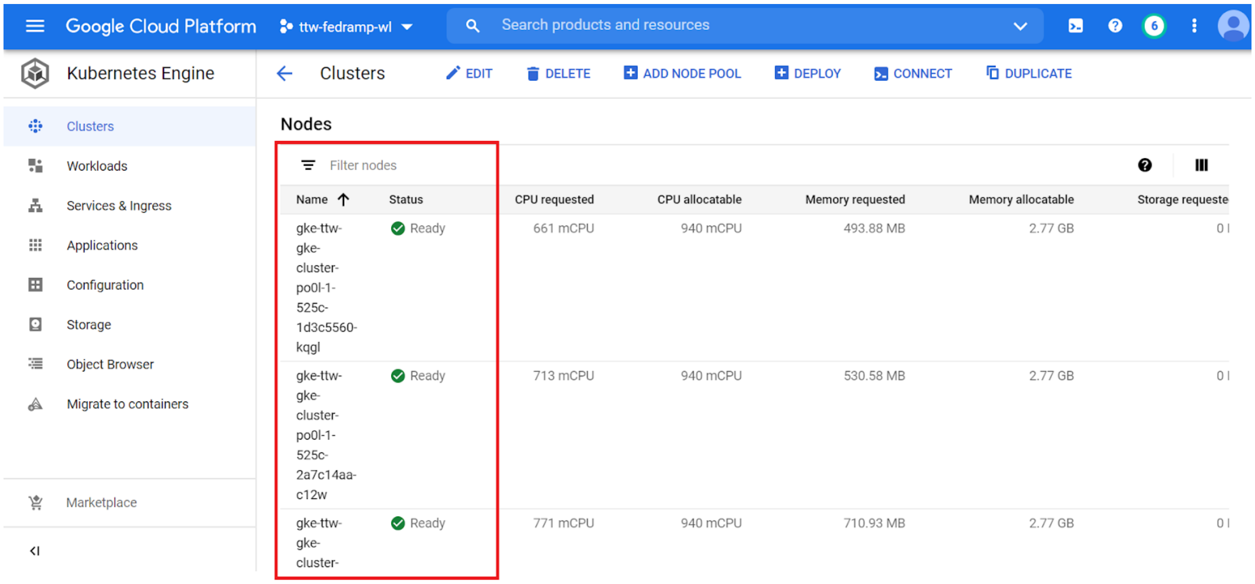
Figure 17. GKE security enabled configurations.



Networking

Private cluster	Enabled	🔒
Default SNAT	Enabled	✎
Master address range	192.168.3.0/28	🔒
Master global access	Enabled	✎
VPC peering	gke-n9b17f70f408c338ee92-e9e5-87e7-peer	🔒
Network	ttw-fedramp-network	🔒
Subnet	ttw-fedramp-gke-subnet	🔒
VPC-native traffic routing	Enabled	🔒
Pod address range	10.2.0.0/16	🔒
Maximum pods per node	110	🔒
Service address range	10.3.0.0/24	🔒
Intranode visibility	Disabled	✎
NodeLocal DNSCache	Disabled	✎
HTTP Load Balancing	Enabled	✎
Master authorized networks	web-subnet (10.0.1.0/24)	✎

Figure 18. GKE network enabled configurations.

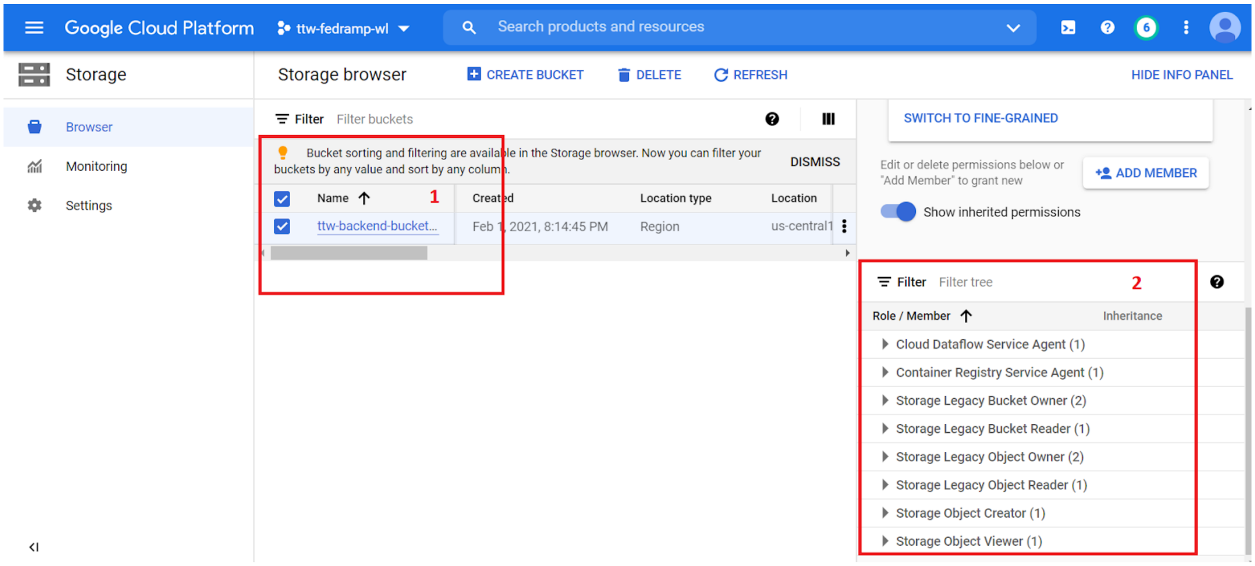


Nodes

Name ↑	Status	CPU requested	CPU allocatable	Memory requested	Memory allocatable	Storage requeste
gke-ttw-gke-cluster-po0l-1-525c-1d3c5560-kqgl	✓ Ready	661 mCPU	940 mCPU	493.88 MB	2.77 GB	0 I
gke-ttw-gke-cluster-po0l-1-525c-2a7c14aa-c12w	✓ Ready	713 mCPU	940 mCPU	530.58 MB	2.77 GB	0 I
gke-ttw-gke-cluster-	✓ Ready	771 mCPU	940 mCPU	710.93 MB	2.77 GB	0 I

Figure 19. GKE nodes.

Cloud Storage Console: Shows the backend storage bucket that is deployed by the template (Figure 20). User can view the storage bucket and its corresponding permissions by selecting the respective buckets in the list.



Bucket sorting and filtering are available in the Storage browser. Now you can filter your buckets by any value and sort by any column.

☑	Name ↑	Created	Location type	Location
☑	ttw-backend-bucket...	Feb 1, 2021, 8:14:45 PM	Region	us-central1

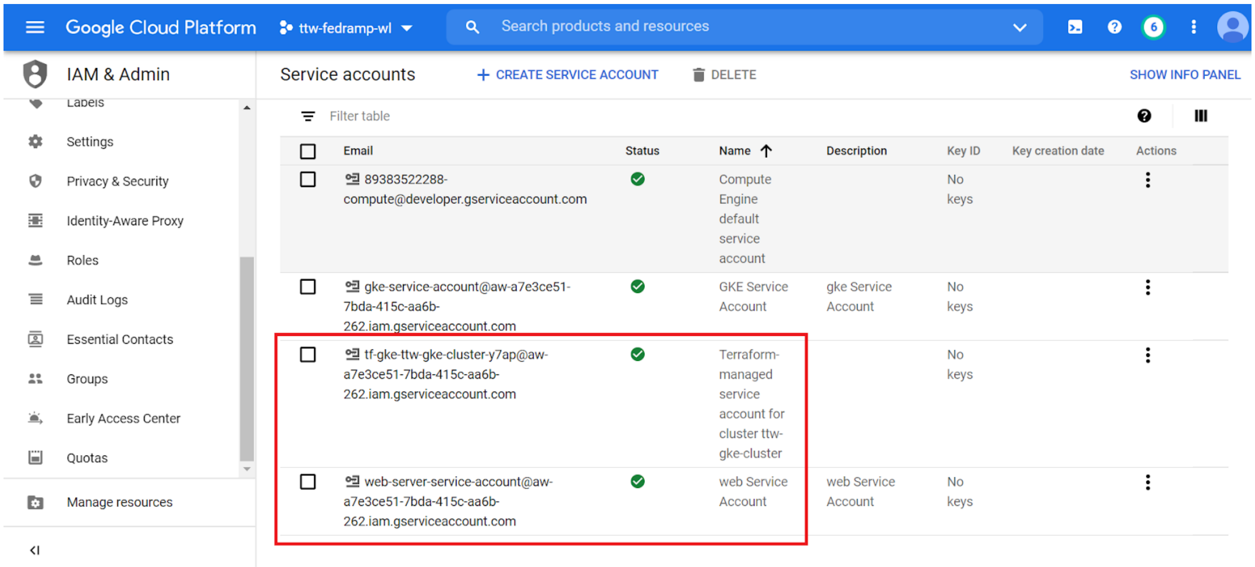
Edit or delete permissions below or "Add Member" to grant new

● Show inherited permissions

Role / Member ↑	Inheritance
Cloud Dataflow Service Agent (1)	
Container Registry Service Agent (1)	
Storage Legacy Bucket Owner (2)	
Storage Legacy Bucket Reader (1)	
Storage Legacy Object Owner (2)	
Storage Legacy Object Reader (1)	
Storage Object Creator (1)	
Storage Object Viewer (1)	

Figure 20. 1: State storage bucket 2: Bucket permissions.

Service Accounts console: Shows the custom service accounts for GKE and Managed Instance Group (MIG) (Figure 21).

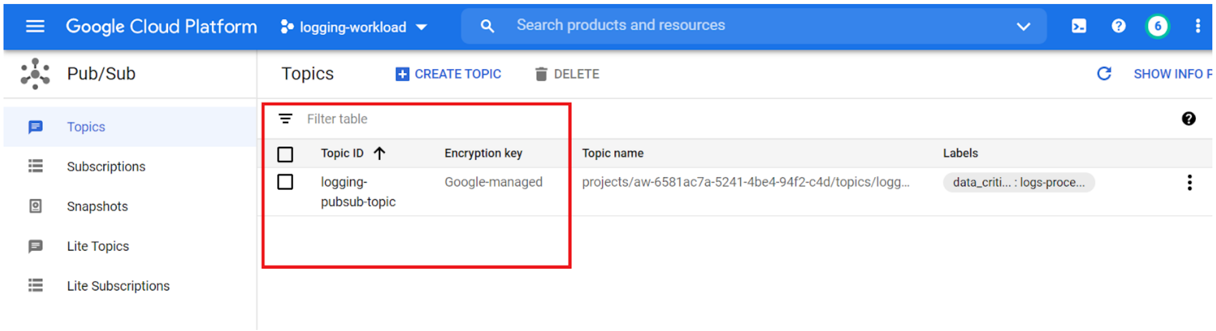


□	Email	Status	Name ↑	Description	Key ID	Key creation date	Actions
□	89383522288-compute@developer.gserviceaccount.com	✓	Compute Engine default service account		No keys		⋮
□	gke-service-account@aw-a7e3ce51-7bda-415c-aa6b-262.iam.gserviceaccount.com	✓	GKE Service Account	gke Service Account	No keys		⋮
□	tf-gke-ttw-gke-cluster-y7ap@aw-a7e3ce51-7bda-415c-aa6b-262.iam.gserviceaccount.com	✓	Terraform-managed service account for cluster ttw-gke-cluster		No keys		⋮
□	web-server-service-account@aw-a7e3ce51-7bda-415c-aa6b-262.iam.gserviceaccount.com	✓	web Service Account	web Service Account	No keys		⋮

Figure 21. Service accounts.

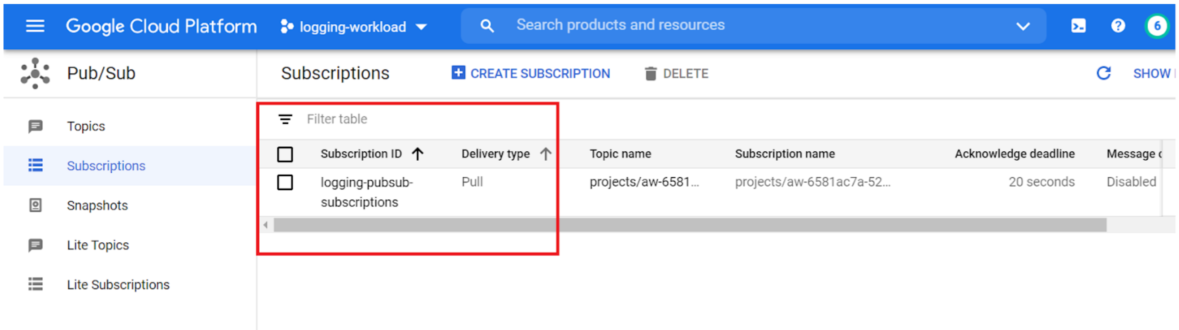
3.7.3 Logging project

Pub/Sub Console: Shows the Pub/Sub topic and the subscription that are created by the template (Figure 22).



□	Topic ID ↑	Encryption key	Topic name	Labels
□	logging-pubsub-topic	Google-managed	projects/aw-6581ac7a-5241-4be4-94f2-c4d/topics/logg...	data_criti... : logs-proce...

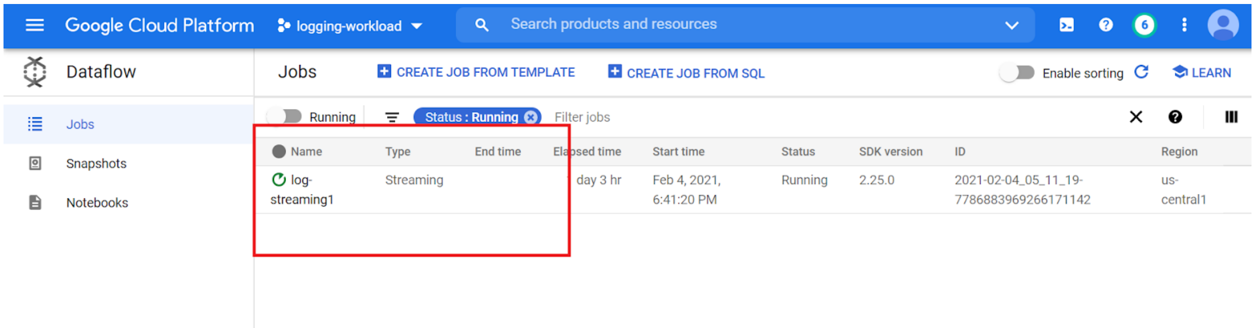
Figure 22. Pub/Sub topic.



□	Subscription ID ↑	Delivery type ↑	Topic name	Subscription name	Acknowledge deadline	Message (
□	logging-pubsub-subscriptions	Pull	projects/aw-6581...	projects/aw-6581ac7a-52...	20 seconds	Disabled

Figure 23. Pub/Sub subscription.

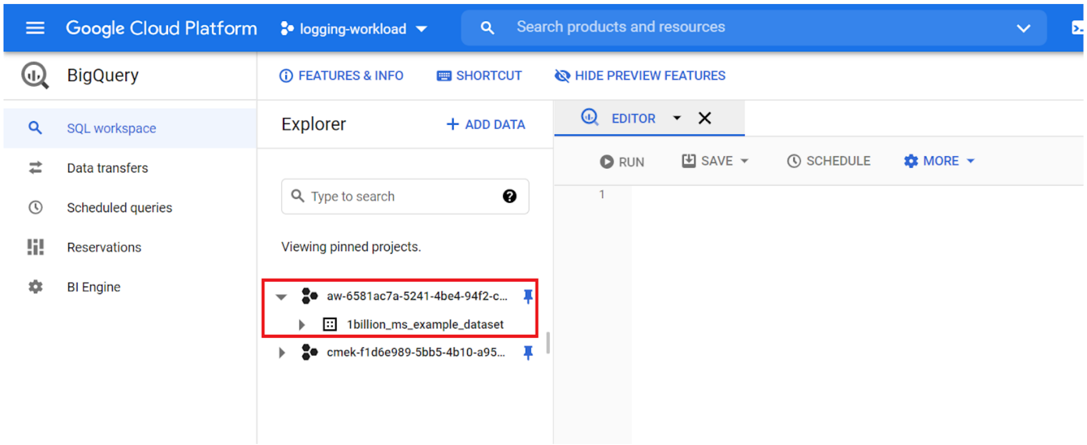
Dataflow Console: Shows the Dataflow job that is created by the template (Figure 24).



Name	Type	End time	Elapsed time	Start time	Status	SDK version	ID	Region
log-streaming1	Streaming		day 3 hr	Feb 4, 2021, 6:41:20 PM	Running	2.25.0	2021-02-04_05_11_19-7786883969266171142	us-central1

Figure 24. Dataflow job.

Bigquery Console: Shows the BigQuery dataset that is created for analysing the logs that are generated by the three-tier workload (Figure 25).

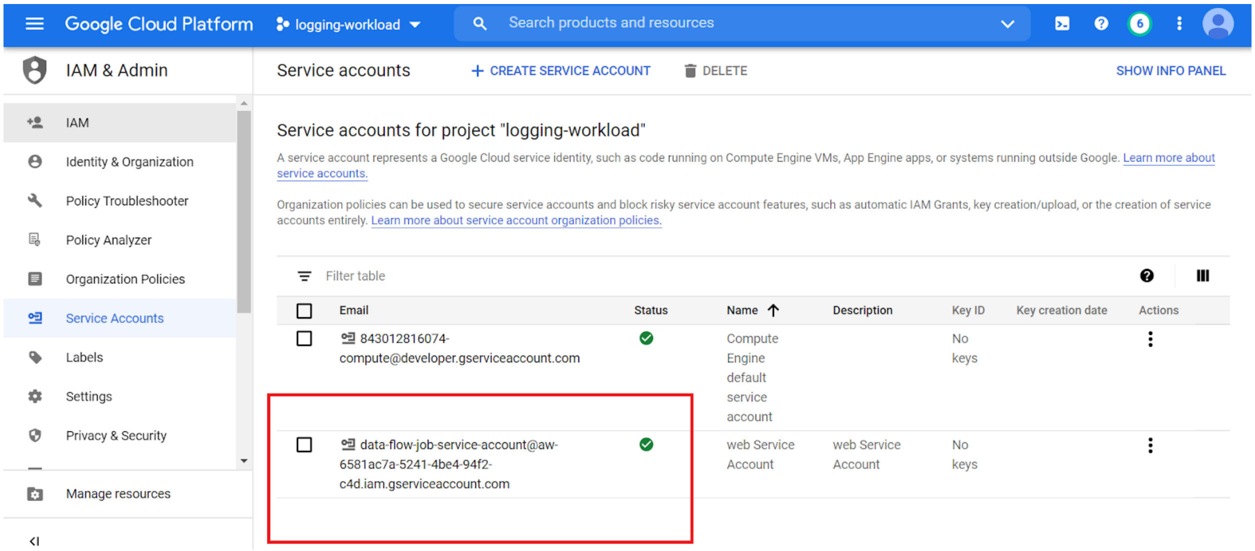


Viewing pinned projects.

- aw-6581ac7a-5241-4be4-94f2-c...
  - 1billion_ms_example_dataset
- cmek-f1d6e989-5bb5-4b10-a95...

Figure 25. BigQuery dataset.

Service Accounts console: Shows the custom service accounts for Dataflow and the default service accounts.



Service accounts for project "logging-workload"

A service account represents a Google Cloud service identity, such as code running on Compute Engine VMs, App Engine apps, or systems running outside Google. Learn more about service accounts.

Organization policies can be used to secure service accounts and block risky service account features, such as automatic IAM Grants, key creation/upload, or the creation of service accounts entirely. Learn more about service account organization policies.

□	Email	Status	Name ↑	Description	Key ID	Key creation date	Actions
□	843012816074-compute@developer.gserviceaccount.com	✓	Compute Engine default service account		No keys		⋮
□	data-flow-job-service-account@aw-6581ac7a-5241-4be4-94f2-c4d.iam.gserviceaccount.com	✓	web Service Account	web Service Account	No keys		⋮

Figure 26. Service accounts.

A. Appendix

A.1 FedRAMP and Google Cloud

Google Cloud Platform supports FedRAMP compliance and provides specific details on the approach to security and data protection in the Google security whitepaper and in the Google Infrastructure Security Design Overview.

Google Cloud customers own their data and control how it is used. It is crucial to remember that enterprises and individuals that use Google Cloud are responsible for understanding FedRAMP and its implications concerning use of Google Cloud products and services hosting applications and services. Some of these aspects are listed below:

- Applicability of the provisions and requirements of FedRAMP across applications, platforms, and Google Cloud Infrastructure.
- Classification and inventory of data such as protected health information (PHI), personal identifiable information (PII) along with the business and information systems that process this data.
- Alignment of current controls, policies, and processes for managing and protecting data with FedRAMP requirements.
- Understanding of the existing data protection features on Google Cloud for meeting FedRAMP requirements.

Review these documents carefully for FedRAMP compliance guidance on Google Cloud:

- Google Cloud FedRAMP Implementation Guide
- FedRAMP compliance on Google Cloud | Assured Workloads
- FedRAMP Moderate shared security model | Assured Workloads
- Google Cloud Platform Terms of Service

A.2 Compliance

Security and privacy on the cloud is a shared responsibility. Google Cloud is responsible for the security of the cloud infrastructure and services. Google Cloud's customers are responsible for the security, privacy, and compliance of their workloads in the cloud. Google Cloud's focus on data security, privacy, and transparency has provided a foundation towards achieving FedRAMP compliance for Google Cloud. In addition, Google Cloud offers data security, data privacy, data portability, and threat protection products and features that can support FedRAMP compliance efforts, some of which have been described in this solution guide. These capabilities not only help prevent abuse or unauthorized access to personal data, but also to maintain security of data and meet FedRAMP requirements.

Google Workspace and Google Cloud Platform are regularly tested, assessed, and evaluated for the effectiveness of technical and organizational security and privacy measures via third-party audits and certifications, such as those listed below:

ISO/IEC 27001 for information security management systems.

ISO/IEC 27017 for cloud security controls.

ISO/IEC 27018 for protection of PII in public clouds acting as PII processors.

SOC 2 and SOC 3 for evaluating systems' and controls' security, availability, processing integrity, and confidentiality or privacy.

To learn more about security and compliance for the Google Cloud Platform, and view our comprehensive compliance documentation, refer to the Compliance resource center.

A.3 Google Cloud shared responsibility model

The shared responsibility model depends on the particular service model. This starts from the bottom of the stack and moves upwards, from the infrastructure as a service (IaaS) layer where only the hardware, storage, and network are Google's responsibility, up to software as a service (SaaS) where most components of the stack except the content (such as the data) and access policies are up to the provider.

To learn more about Google Cloud's shared responsibility model, refer to the Google Infrastructure Security Design Overview.

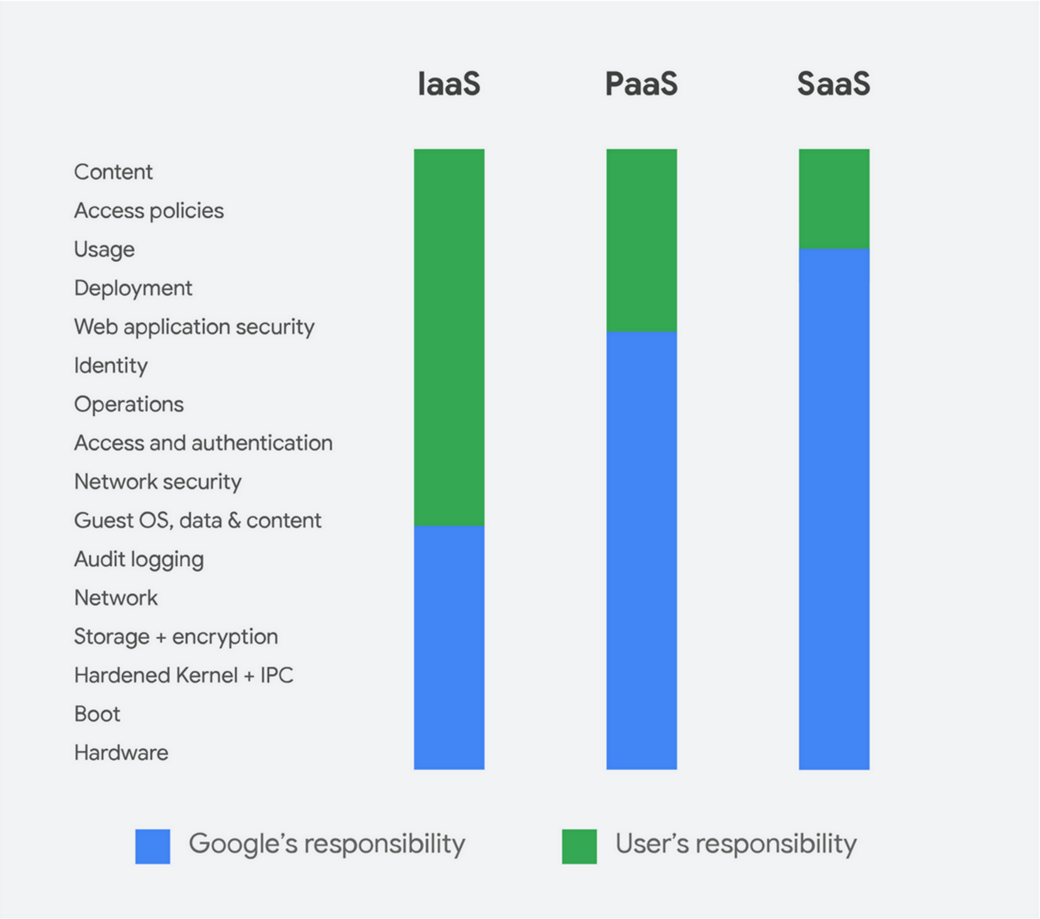


Figure 45: Google Cloud shared responsibility model.

In general, Google is responsible for the security of the underlying infrastructure, including hardware, firmware, kernel, OS, storage, network, and more. This includes encrypting data at rest by default, encrypting data in transit, using custom-designed hardware, laying private network cables, protecting data centers from physical access, and following secure software development practices.

Conversely, customer responsibility for security and compliance in the cloud is listed in Appendix A.4.

A.4 Customer responsibilities

The following items are typical examples of overall security and compliance capabilities for which the customer is responsible. For additional guidance, refer to the FedRAMP Shared Security Model and Google Cloud FedRAMP Implementation Guide.

Identity and access management

- User access provisioning.
- Custom roles to control access.
- Monitoring to detect unusual activity by users and administrators.
- Role-based access controls and separation of duties.
- Multi-factor authentication and two-step verification for access-critical environments and sensitive data.
- Periodic cadence for reviewing access lists.

Governance, risk, and compliance

- Governance and implementation of organization-specific security policies and standards.
- Definition of security-specific key performance indicators (KPIs) and key risk indicators (KRIs).
- Security awareness training and secure coding practices training and reporting.
- Background verification checks by authorized parties prior to granting access.

Data security

- Consent collection, logging, tracking, and monitoring by end-users for organizational access to their PII, PHI, or other sensitive data.
- Data governance lifecycle and data management strategy.
- Data classification, labeling, and handling as per regulatory requirements, service level agreements and operational continuity requirements.
- Policies and procedures for reuse, disposal, and deletion of resources (for example, like data, equipment, and digital media).
- Data destruction, obfuscation, and archival standards, including supporting tools and technologies.
- Key management, if you choose to use the CMEK or customer-supplied encryption keys (CSEK) capabilities of specific products.
- Communication of incidents and notifications about data breaches, including reporting to regulatory authorities and customers.

Infrastructure security

- Configuration and validation of firewall rules for both egress and ingress traffic.
- Penetration testing, closed-box testing, and red-teaming exercises.

Security operations

- Audit logs for security events, faults, exceptions, and data access violations.
- Continuous monitoring of security events.
- Enabling of data-read and data-write logs for all critical environments and projects, which are reviewed on a periodic basis.
- Event logs stored and backed up at a centralized storage location which is protected against tampering and unauthorized access.
- Clocks for all resources are synchronized with approved time sources like network time protocol (NTP) servers or domain controllers.