

Technical Infrastructure Design Document

Authors: Brad Svee, Anuradha Bajpai, Ryan Przybyl, Davis Pynadath, Richard Foltak, Jason Beavans

Prepared for: American Express

Contents

[**About the document**](#_9k1jtlt62l0i) **9**

[**Document Control**](#_2qgp1cbqu9nf) **10**

[**1. Introduction**](#_t2keswfxub54) **12**

[**2. Resource management**](#_d14qi3i54po4) **12**

[2.1 Current environment](#_i7zwpskhwndf) 13

[2.2 GCP Resource Management](#_991h1zpbol7k) 13

[2.2.1 Resources](#_k15kbx7cw1ry) 13

[2.2.2 Naming Conventions](#_7pw4mv18r2yz) 13

[X 2.2.3 Resource hierarchy](#_p46h931xlyw3) 17

[2.2.4 Regions and Zones](#_pxsroxhqnrzb) 19

[2.2.5 Resource Quota](#_hlda9eq6xdko) 19

[2.2.6 Constraints with Organization Policies](#_cvc2a5i2wffl) 19

[2.2.7 Labels](#_i0ele21aylhx) 23

[2.2.7.1 Services currently support labels](#_nfsgzkdd258w) 23

[2.2.7.2 Defining your Labels](#_rwxmm2709zw) 24

[2.2.7.3 Use cases for labeling](#_5c4jtmdex7pm) 25

[2.2.7.4 Proposed solutions for label enforcement:](#_seq8apxexg2f) 26

[2.2.8 Resource Provisioning](#_d4705u455qyn) 26

[2.2.9 Change Management](#_vk5jxyvmua2v) 26

[2.2.9.1 Tracking Changes within GCP](#_4l7cgvda62pf) 27

[**3. Identity and access management**](#_fu7zm2nyrn2w) **29**

[3.1 Current Architecture](#_yaw7qsx21cig) 29

[3.2 Cloud Identity](#_a3tywnx97tkp) 30

[3.2.1 Admin Console](#_lt8bqbrxsl6z) 30

[3.2.2 Identity](#_1ow1b1p81o5) 31

[3.2.3 Super Admin](#_s8oyk6gajik9) 32

[3.2.4 Organization Admin](#_b9k816jpq5pi) 33

[3.2.5 Users](#_qh2exd7zq9j6) 33

[3.2.5.1 Provisioning](#_ovc248x69n3h) 33

[3.2.5.2 Authentication](#_9orbgg5kmb9d) 34

[3.2.5.3 Conflicting accounts](#_49fffmvzuuud) 34

[3.2.6 Groups](#_q56m2l91p9pu) 35

[3.3 Authorization](#_3w47whan5sq8) 38

[3.3.1 IAM Roles](#_tq41popa5ytz) 39

[3.3.2 Project Permissions](#_ifoz9dj61u5v) 39

[3.3.3 Access Control Auditing](#_g5udb79jm7x) 40

[3.3.4 Service Account](#_5ioge2bxq6ke) 41

[3.3.4.1 Using Service accounts outside GCP](#_ah5gkh76m7ip) 42

[3.3.4.2 Using short lived service accounts credentials](#_kpyfmwker8xo) 42

[3.3.4.3 Service account impersonation](#_l8fngymerz1l) 42

[3.3.4.4 Service account best practices](#_wcbru7ip8ev) 42

[3.3.5 Service Account Key Management](#_lx71l9tauew) 44

[3.3.6 Cloud IAM Policies with Conditions](#_4p7xgqe8yd66) 45

[**4. Networking**](#_ex13uamus6aq) **46**

[4.1 Networking requirements](#_fu0oje162999) 48

[4.2 Proposed VPC Design](#_5yeugb1zloby) 49

[4.2.1 Current Phase](#_auzkpwdyyaj2) 49

[4.2.2 Future Phase](#_h5ge04r4hvws) 52

[4.3 Connection to Google](#_tou4cuqpiofc) 54

[4.3.1 Internal](#_fmmm20mfz6xr) 56

[4.3.2 External](#_86uq6i008lf) 57

[4.4 IP address space](#_nszcl657oaxb) 58

[4.4.1 GKE IP Addresses](#_g6sxdlc5bfi) 58

[4.4.2 Dataproc IP Addresses](#_8psxaglabygb) 59

[4.4.3 Other IP Addresses](#_c23rzaicl85f) 59

[4.5 Cloud DNS](#_auh1ovf5hp37) 60

[4.5 Cloud Load Balancing](#_31eb11kb7bcr) 61

[4.7 Firewall rules](#_ufkpljdoypw9) 62

[4.7.1 Evaluation Logic](#_4ii1pg9rjx6s) 63

[Firewall Layers](#_8xri5ftyv3f8) 65

[Priority 0-9,999](#_r2m04p3xecmq) 65

[Priority 10,000](#_me81932tngm9) 66

[Priority 10,001-29,999](#_qffr3g6o41wt) 66

[Priority 30,000](#_utk0qnr4ywl2) 66

[Priority 30,001-65,529](#_hqhc0hmpxhkr) 68

[Priority 65,530](#_6dpb437b96y1) 68

[4.8 VPC Service Controls](#_310coxe5um0f) 68

[4.8.1 Service Perimeter](#_pkzrl8aywil6) 68

[4.8.1.1 Overview](#_kcb3wduo3e6) 68

[4.8.1.2 Prod Perimeter - Configuration](#_2sug7knswnkw) 69

[4.8.1.3 Non Prod Perimeter - Configuration](#_k36eq6pxgxjz) 69

[4.8.1.4 Shared Services Perimeter - Configuration](#_k7xiho8k73lf) 70

[4.8.1.5 Anonymized Data Perimeter - Configuration](#_9cuuoojbcdc9) 70

[4.8.2 Access Levels](#_hhcitodqrqw8) 71

[4.8.3 Ingress & Egress Rules](#_9kzxpba4ry02) 72

[4.8.3.1 Prod Perimeter](#_fy58y17937l) 72

[4.8.3.2 Non-Prod Perimeter](#_ni824deu8soc) 74

[4.8.3.3 Shared Services Perimeter](#_rqv2nzjq3g1) 76

[4.8.3.4 Anonymized Data Perimeter](#_8t52byi8npd7) 77

[**5. Logging and Monitoring**](#_d5cwktvstaqf) **80**

[5.1 Current architecture](#_lia4xxia9106) 80

[5.2 Logging requirements](#_hy6ua9i6p2qk) 81

[5.3 Logging findings and recommendations](#_84wpsf2cgst9) 81

[5.4 Categories of Logs](#_9ftb6d3uv3hg) 81

[5.4.1 Google Cloud platform logs](#_tz2edxqsfgff) 82

[5.4.3.1 VPC Flow Logs](#_ap8x8lqip069) 82

[5.4.3.1 Firewall Rule Logs](#_4e1dc6555uyw) 82

[5.4.2 User written logs](#_qo4v0d20f2a3) 82

[5.4.3 Security logging](#_pqoj8o2wu4yd) 82

[5.4.3.1 Audit logs](#_71tc9jiw5mdf) 83

[5.4.3.2 Access Transparency logs](#_i0e8a1bug32f) 85

[5.4.4 Multi-cloud and hybrid-cloud logging](#_6tpowilx3gq1) 86

[5.5 Using and managing Logs](#_kcgj8nl2g5ou) 86

[5.5.1 Log entry structure](#_59r0k71dzbwp) 86

[5.5.2 Viewing logs](#_l8ewvu2lppdn) 86

[5.5.3 Routing logs](#_p97n9r9ytl2m) 87

[5.5.3.1 Log Router](#_ycrctmb3r23c) 89

[5.5.3.2 Sinks](#_t79wyvb7ja3q) 89

[5.5.3.3 Regionalization](#_5ccaesvyxpzh) 90

[5.5.4 Retention](#_6zn0hvnd6t1s) 90

[5.6 Log Compliance](#_nciewl10jfvo) 91

[5.6.1 Inadvertent PII data within Logs](#_9dafvwsk2ll) 91

[5.7 Monitoring requirements](#_mr3n1rjztw4) 91

[5.8 Monitoring findings and recommendations](#_ekk3xm34y7im) 92

[5.8.1 Alerting](#_w7v25qk4r7zh) 92

[5.8.2 Log-based alerting](#_lu6tsrhp61f) 92

[5.8.3 Dashboards and Charts](#_ltqoormji2qq) 93

[**6. Security**](#_105upw9geifp) **93**

[6.1 Current Architecture](#_z5stw6x3f43) 94

[6.1.1 Amex Approved Services](#_dfd1ap42muvi) 95

[6.2 Requirements](#_goib4gu6cmm) 96

[6.2.1 Encryption in transit](#_aa19iz97dx4o) 96

[6.2.2 Encryption at rest](#_e5n85vynvfm2) 96

[6.2.3 Organization security](#_9zvd95b7fhpb) 96

[6.2.4 Data security](#_m7mja1emnq03) 96

[6.2.5 Security events](#_9kk0l4aq8mou) 96

[6.3 Design options](#_cfsb4ldncw3f) 97

[6.3.1 Identity and Access Management](#_6dc1gra19owt) 97

[6.3.2 Organization Policies](#_wwquskste4pn) 97

[6.3.2.1 Resources hierarchy](#_3bnkvaem4cr6) 97

[6.3.2.2 Constraints](#_qatv60y8enpw) 98

[6.3.2.3 Violations](#_htk11p713o92) 101

[6.3.3 VPC service controls](#_oxqo0wnjrmq) 102

[6.3.3.1 Perimeter bridge](#_kqxuz3dqj35r) 102

[6.3.3.2 Perimeter definition](#_mg49egndvn1s) 102

[6.3.3.3 Perimeter shared resources](#_ye04rfud6clt) 104

[6.3.3.4 Access context manager (access level)](#_6ae6m8j7jrqp) 104

[6.3.3.5 Services Restrictiction](#_iphm5m5latpl) 104

[6.3.3.6 Custom perimeter definition](#_1gyjl4y87935) 106

[6.3.4 Encryption in transit](#_ej4jfgf1w1fs) 107

[6.3.4 Encryption at rest and key management](#_8bht5fwxa8ip) 109

[Key States](#_g8asc2779jtn) 110

[6.3.4.1 KMS Project & roles segregation](#_vsa4c9i0smkf) 110

[6.3.4.2 Key rotation](#_akvndnb6dq8y) 111

[6.3.4.3 Key Deletion](#_f58fl8uyrsni) 112

[6.3.4.4 Software vs HSM operations](#_ghuaaa3ozj30) 112

[6.3.4.5 CMEK Integration](#_bnk1592b0146) 113

[6.3.5 Google Secret Manager](#_pdr7xnl5egtz) 114

[6.3.6 Google Access Transparency](#_51s7441xmvdk) 115

[6.3.6.1 Access approval](#_hhcejy9qlm5) 115

[6.3.6.2 Access justifications](#_uuvng3i55qv) 115

[6.3.6.3 Resource and method identification](#_wfq8uiiqw64) 115

[6.3.6.4 Cloud Logging integration](#_jcc25kivlboy) 115

[6.3.6.5 Accessor location](#_7k3jklqy7h5x) 115

[6.3.6.6 Data protection controls](#_jsxtrctdmlot) 115

[6.3.6.7 Near real-time publication](#_wa5mslad69ak) 115

[6.3.7 Cloud Data Loss Prevention (DLP)](#_az1fe6wgmc84) 116

[6.4 Security Design](#_oi4pshl97wwz) 117

[6.4.1 Network Security](#_s9qys1nixpcu) 117

[6.4.1.1 Traffic Director](#_ft91y2m2ok1f) 117

[Policy-driven security](#_cnfubcqhjip5) 117

[Securing traffic (authentication)](#_ft3ccybt2azr) 117

[Encryption & supported authentication modes](#_a8ygfirwtng3) 117

[6.4.1.2 Certificates & certificate authorities](#_sgbf6l4m25qo) 118

[6.4.1.3 Private Google Access](#_q0fr5evhky2s) 118

[6.4.2 Security Monitoring](#_awx7xnr32ux8) 119

[6.4.2.1 Cloud Security Command Center (SCC)](#_uxeggstopd35) 119

[6.4.3 Third Party Integrations](#_s7r4g53wsjer) 122

[6.4.3.1 ServiceNow](#_fxj9qjh2y3li) 122

[6.4.3.2 Prisma Cloud](#_gu2w6r2k5ajq) 123

[6.4.3.2.1 Prisma Cloud Service Exception](#_97r8tio46fs0) 124

[6.4.3.3 Hashicorp Vault](#_8et2lxc8zh3u) 124

[6.5 Application Security](#_i21d3ue8k29m) 126

[6.5.1 Application-layer Secret Encryption (Kubernetes)](#_uqolwoshyne7) 126

[**7. Billing and Quota**](#_alr0pjhal3ak) **126**

[7.1 Cloud Billing Account & Payments Profile](#_rt5czm1d73sn) 127

[7.1.1 Overview](#_hv2pspbc5ye2) 127

[7.1.2 Billing contacts](#_6a6bx4s6n7p) 128

[7.1.3 Access control](#_yrhyh6yzveik) 128

[7.2 Cloud Billing Resource Hierarchy](#_v7yc2fvmra7x) 129

[7.3 Billing Accounts](#_ijd49fpj69ue) 130

[7.4 Cloud Billing Budget](#_payr06n9fzd8) 131

[7.4.1 Overview](#_ygvdakk3i27c) 131

[7.4.2 Budget alerts](#_1a8wht9galik) 132

[7.5 Billing Labels](#_gcxldq2k3mxv) 132

[7.6 Billing Reporting](#_frplrk62erxk) 133

[7.6.1 Budget report](#_oi3dr5dixj1e) 133

[7.6.2 Cost table report](#_o70p1cz4au60) 134

[7.6.3 Cost breakdown report](#_up9l8q4p9ttp) 134

[7.6.4 Prices per SKU pricing report](#_jz7hl9rcwo68) 134

[7.6.5 Committed use discounts analysis reports](#_4vjrn42pcbj8) 134

[7.6.6 Exported billing data](#_28umbxillie) 135

[7.7 Quota](#_e8qwtoo7gun1) 135

[**8. Compute and Storage**](#_s2kqoef6xcl6) **135**

[8.1 Google Cloud Storage](#_frvhj5yzu9ic) 135

[8.1.1 Buckets](#_hnhqehwpj4sk) 136

[8.1.2 Objects](#_r1doh0miqh4d) 136

[8.1.3 Storage Classes](#_5omfqq8a0v6a) 136

[8.1.4 Data Retention Lifecycle](#_kpxmo8b42mld) 136

[8.1.4.1 Object Versioning](#_eqjle776equs) 136

[8.1.4.2 Object Lifecycle Management](#_bkcggcw9xnbb) 137

[8.1.5 Geo-Redundancy](#_up3vysisvcty) 137

[8.1.6 Access Control](#_3t4j14k7hr3m) 138

[8.1.6.1 Access Control Overview](#_3y1iml870sqy) 138

[8.1.6.2 Recommendations:](#_a3va20nmiiz0) 139

[8.2 BigQuery](#_6dhc0hq0ao9u) 139

[8.2.1 BigQuery Storage](#_m9nvq6avxuy5) 139

[8.2.2 BigQuery Analytics](#_rjihyvfrn7wh) 140

[8.2.3 BigQuery Analytics](#_q1bttl8ye2v8) 140

[8.3 Dataproc](#_ti9u18etgjwj) 140

[8.3.1 Key Features](#_bs5maaaqtkvq) 140

[**9. Incorporating a DevOps culture**](#_i8ovdue333uy) **141**

[9.1 Recommendations](#_l7rr0ciisc7f) 142

[9.2 CI/CD process for AMEX](#_njzyeyiaishk) 142

[9.3 Infrastructure As Code](#_lmd0q385em1j) 142

[9.4 IaC Testing](#_2j5299qkroyb) 143

[9.4.1 Testing Methods](#_1pjet5ij2inv) 143

[9.4.2 Testing best practices](#_hjemwtw8cvrh) 144

[9.4.2.1 Optimizing testing runtime](#_wy402w18lvcr) 145

[**10. High Availability and Disaster Recovery**](#_1f4jddkbjfw6) **145**

[10.1 How Google Cloud is designed for resilience](#_5l7xejpfbrhd) 145

[10.1.1 Google data centers](#_i2wz46s6cl0c) 145

[10.1.2 Regions and zones](#_c4z5qaapiist) 146

[10.2 Design for scale and high availability](#_dv2c3ok6qjx9) 146

[10.2.1 Create redundancy for higher availability](#_n02zgoeyvssr) 147

[10.2.2 Design a multi-zone architecture with failover for high availability](#_ellc3hgcmwjv) 147

[10.2.3 Replicate data across regions for disaster recovery](#_rna6tds05cg7) 147

[10.2.4 Design a multi-region architecture for resilience to regional outages](#_ia5efrlul5v0) 147

[10.3 Product reference](#_l36sr9lrfl9) 148

[10.3.1 Common themes](#_nnic9my57ss) 148

[10.3.2 Compute Engine](#_7lqrirkahvqa) 149

[10.3.2.1 Networking for Compute Engine](#_qcl2o2wqdtm7) 149

[10.3.2.2 Cloud Load Balancing resilience](#_kk9ifvk3yijr) 150

[10.3.3 Dataproc](#_yty7pc3e6kyb) 150

[10.3.4 BigQuery](#_9yu0d9gv86bz) 151

[10.3.4.1 Single region configuration](#_6dlqdkck7cui) 151

[10.3.4.2 Multi-region (US / EU) configuration](#_f8r2dduky8mm) 151

[10.3.5 Google Kubernetes Engine](#_hglbfp1g4orz) 151

[10.3.6 Cloud Key Management Service](#_9a6ep6vvw16u) 152

[10.3.7 Cloud Storage](#_c2cyss1utxb4) 152

[10.3.8 Pub/Sub](#_87z1rvxx633i) 153

[10.3.9 Cloud Logging](#_puo6sl1wbmhh) 154

[10.4 Create reliable operational process and tools](#_a9x029firkex) 154

[10.4.1 Automate build, test, and deployment](#_o2nkiyq8j7i0) 154

[10.4.2 Test failure recovery](#_qcdkytvhza5) 154

[10.4.3 Conduct disaster recovery tests](#_cyttu2w88p9h) 155

[10.5 Disaster Recovery Planning](#_tn43u4gq0t3a) 155

[10.5.1 Basics of DR planning](#_sbjkdmy45xd0) 155

[10.5.2 DR patterns](#_dmrh7jvzg2r9) 156

# 

# 

# About the document

| **Highlights** | |
| --- | --- |
| **Purpose** | To introduce the reader to the architectural design, implementation and components of a highly scalable, robust and secure American Express (AMEX) deployment on Google Cloud Platform |
| **Intended audience** | AMEX application stakeholders, including: security teams, compliance teams, data teams, infrastructure teams ,application teams, architects, and engineers, who are looking to understand and/or deploy and/or implement and/or support the targeted workloads’ infrastructure policies, security standards and governance policies in deployment across production and non-production environments. |
| **Key**  **assumptions** | The audience has technical knowledge of Google Cloud Platform along with understand security and governance policies and the relevant components liking of cloud architecture, a basic understanding of Infrastructure setup needs like IAM and networking, the importance of storage, pipelines and warehousing |

# 

# Document Control

**Document source**

https://docs.google.com/document/d/1tH3pkMCH8XPE\_kramAHRNFjGEKNXnRCsLyj5kiN7Ai0/edit?usp=sharing

**Document Approver(s)**

All approvers are required. Record of each approver must be maintained.

| **Approver Name** | **Role** | **Date Approved** | **Version Approved** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

**Document Reviewer(s)**

Record of each reviewer must be maintained.

| **Reviewer Name** | **Role** | **Date Reviewed** | **Version Approved** |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |

**Revision History**

The document author is authorized to make the following types of changes to the document without requiring approval:

* Editorial, formatting, and spelling errors
* Clarification

Changes to this document are summarized in the following table in reverse chronological order (latest version first).

| **Version** | **Date** | **Comment** |
| --- | --- | --- |
| 2.0 | January 28, 2022 | Added sections 9.3 Infrastructure as Code and 9.4 IaC Testing. 2.2.3 Resource Hierarchy updated. 7.6.1 Budget Report updated. |
| 1.6 | December 13, 2021 | Updated section 3. Identity and access management incorporating AMEX comments. Updated section 5.5.3.2 Sinks to include updated diagram. Added section 10. High Availability and Disaster Recovery |
| 1.5 | November 30, 202 | Added section 7. Billing and Quota, section 8. Compute and Storage, and section 9. Incorporating a DevOps culture |
| 1.4 | November 18, 2021 | Updated section 3.2.2 Identity and section 4. Networking |
| 1.3 | November 12, 2021 | Updated section 4. Networking incorporating AMEX feedback. Section 5. Logging and Monitoring added |
| 1.2 | October 19, 2021 | Updates to Security and Networking sections. Delivered to AMEX |
| 1.1 | September 23, 2021 | Delivered to AMEX |
| 1.0 | August 3, 2021 | Initial version |

# 

# 1. Introduction

This document details the foundational architecture design and Google’s recommendations for American Express’s Google Cloud Platform (GCP) implementation. It is intended for use by American Express’s central IT team managing the foundational platform, as well as the data teams and application teams working with the data and onboarding onto the platform. This document describes the best practices, the recommended architecture, along with key decisions made throughout the design process, in order to provide American Express with a reliable, compliant, secure, and scalable cloud foundation. This document can be used as Amex’s GCP Information System Document, Security Architecture and Security Plan.

Through this engagement, American Express (AMEX) team would like to build a Google Cloud Platform “secure landing zone” in order to enable their internal teams to deploy and develop data analysis tools and applications in the cloud with minimal human intervention throughout the deployment of the underlying infrastructure.

The engagement should be treated as a green field activity, enabling the organization to follow GCP best practices going forward for all new projects, while leaving as-is any potential existing GCP workloads.

At Google Cloud, we understand a landing zone encompasses a number of different tools, technologies, and processes that allow data engineers, data analysts, developers and other employees to focus on their technical work, while at the same time reducing toil and risk for the operations and security teams managing, maintaining, and monitoring the platform. The goal of this engagement is to capture the functional business and technical objectives, document and analyze these requirements, and provide design, recommendations and code to deploy the GCP foundational elements based on GCP best practices. Finally we will handoff all developed assets to the AMEX team and guide the Customer through deployment of a pilot environment through a series of demos and by providing associated documentation.

The landing zone being developed for AMEX makes use of modern “configuration as code tools”, with Hashicorp’s [Terraform](https://www.terraform.io/cloud), with modules based on Google’s [Cloud Foundation Toolkit](https://cloud.google.com/foundation-toolkit). The intention is to create all resources in GCP using Terraform, creating a no-touch production environment that is fully automated provisioning through tight integrations with their Servicenow change management processes. Overall, this represents industry-wide best practices for cloud infrastructure deployment and automation.

# 2. Resource management

All cloud resources belong to a [Google Cloud Platform project](https://cloud.google.com/compute/docs/projects). Projects form the basis for enabling and using the Google Cloud Platform services, including managing APIs, enabling billing, adding and removing collaborators, managing quota, and enabling other Google services. Each project is a separate compartment and each resource belongs to exactly one project. Projects can have multiple owners and users. Each project is managed and billed separately, though multiple projects may have the same billing account.

From a risk management perspective, a principle requirement for American Express is to establish the most restrictive configuration mode following the Google Cloud Platform (information system) best practices that enable the desired operational business outcomes. [NIST 800-53 CM-6](https://csrc.nist.gov/projects/risk-management/sp800-53-controls/release-search#!/control?version=4.0&number=CM-6)

## 2.1 Current environment

AMEX is conducting a pilot of the Google Cloud Platform’s (GCP) BigQuery product for Customer’s Enterprise Data Warehouse (EDW) needs, and AMEX has engaged Google Cloud Professional Services for guidance. The GCP Foundation will be designed and built for this pilot project and for future migrations.

## 2.2 GCP Resource Management

### 2.2.1 Resources

In the GCP environment, all the cloud resources that are created must belong to a [project](https://cloud.google.com/compute/docs/projects). Projects form the basis for enabling and using the Google Cloud Platform services, including managing APIs, enabling billing, granting and revoking authorizations to collaborators, and enabling other Google services. Each project is a separate container and each resource belongs to exactly one project, though these resources can communicate with each other if their VPC networks are configured to permit it. Multiple authorizations can be granted to users and groups in each project. Each project is managed and appears itemized separately in billing invoices, though multiple projects may share the same billing account.

Because resources cannot be migrated between projects (they are sticky to the project in which they are created), care must be taken to ensure that each resource is created in the correct project. Resources can be placed in separate projects in order to:

* allow separate teams to have control over different instances of the same type (separation of duties)
* allow separate resources to be itemized separately on an invoice (billing separation)
* exceed the number of resources permitted by a project-based quota

### 2.2.2 Naming Conventions

We recommend that you have a standardized naming convention across various Google Cloud resources. Table 1.1 lays out conventions for creating names for elements in Google Cloud for the example.com reference architecture. Table 1.2 describes the field values that are used in the names.

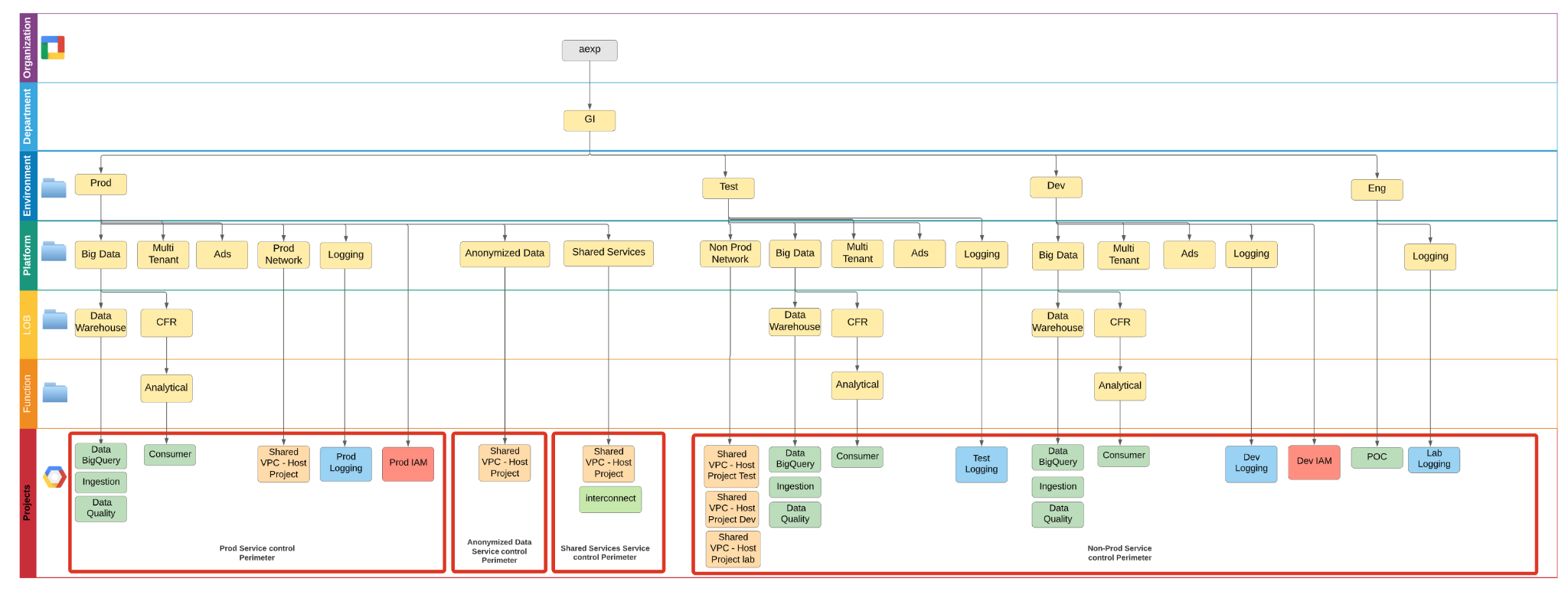
| **Resource** | **Naming Convention** |
| --- | --- |
| Folder | fldr-<environment-code>-<folder-description>  Examples:  fldr-p-analytical  fldr-p-engineering |
| Project | prj-<business-code>-<environment-code>{-<project-description>}-<unique-number>\*  Examples:  prj-amex-p-kubernetes  prj-amex-p-inventory |
| VPC | vpc-<environment-code>-<vpc-type>{-<vpc-description>}  Example: vpc-p-shared-base |
| Subnet | sb-<vpc-name>-<region>{-<subnet-description>}  Example: sb-p-shared-base-us-east1-net1 |
| Firewall | fw-<vpc-name>-<direction>-<action>-<src-label>-<dest-label>-<protocol-port>{-<firewall-description>}  Example: fw-p-shared-base-i-a-all-all-tcp-80 |
| Cloud router | cr-<vpc-name>-<region>{-<cloud-router-description>}  Example: cr-p-shared-base-us-east1-cr1 |
| Route | rt-<vpc-name>-<instance-tag>-<next-hop>{-<route-description>}  Example: rt-p-shared-base-all-default-windows-activation |
| Cloud Interconnect Connection | ic-<onprem-dc>-<colo>  Example: ic-dal-lga-zone1-1422 |
| Cloud interconnect VLAN attachment | vl-<ic-name>-<cr-name>  Example: vl-dal-da1-zone1-p-shared-base-us-east1-cr1 |
| Group | gcp-<team-name>-<role>-<region>  Example: gcp-sre-engineers-na |
| Role | rl-<function>{-<role-description>}  Example: rl-compute-admin |
| Service Account | sa-<app-name>-<service-account-description>  Example: sa-p-acde-shared-base-data-bkt-reader |
| Storage | bkt-<project-name>{-<bucket-description>}  Example: bkt-amex-p-inventory-assets |

Table 1.1 Naming conventions for reference

| **Field** | **Description** | **Values** |
| --- | --- | --- |
| environment | A description of the folder-level resources within the Google Cloud organization. | prod, dev, staging, sb1, tools, cicd, shared, nap2 |
| environment-code | A short form of the environment field | p, d, s, sb1, t, cicd, sh, p2 |
| business-code | A 4-character code that's used to associate a project with a business unit or group. | A uniquely identifiable 4-character code. For common projects that are not related to a business unit, zzzz is used. In the example 2-character code au is used. |
| unique-number | A globally unique identifier. | A 5-digit integer |
| vpc-type | The type of VPC network that's being established. | shared, service, float, nic, peer |
| region | The region that the resource is located in. | Any valid [Google Cloud region](https://cloud.google.com/compute/docs/regions-zones). Short forms are used for some names and directions: Australia (au), North American (na), South America (sa), Europe (eu), southeast (se), and northeast (ne). |
| priority | A numerical value that specifies the priority of the Google Cloud route or firewall rule. | For details, see the documentation for [firewall](https://cloud.google.com/vpc/docs/firewalls#priority_order_for_firewall_rules) [priorities](https://cloud.google.com/vpc/docs/firewalls#priority_order_for_firewall_rules) and [routing order](https://cloud.google.com/vpc/docs/routes#routeselection). |
| direction | The direction of traffic relative to Google Cloud that the firewall applies to. | i for ingress, e for egress |
| action | The action to take if a firewall rule matches. | a for allow, d for deny |
| src-label | The instance source label to which a firewall is applied. | all (indicating 0.0.0.0/0 ), the source IP range, or source tags (list) |
| dest-label | The instance destinations label to which a firewall is applied. | all (indicating 0.0.0.0/0 ), the source IP range, or source tags (list) |
| protocol | The protocols to which a firewall is applied. | all, a single protocol, or a combination of protocols (tcp, udp, tcpudp , and so on) |
| port | The port or port range on which a firewall is applied. | A port number or port number range |
| instance-tag | The instances to which a route is applied. | Instance tags |
| next-hop | The next-hop destination where the route will direct traffic, if applicable. | default , an instance ID, an IP address, a VPN tunnel name, oran internal load-balancer address |
| onprem-dc | The name of your data center to which an interconnect is connected. |  |
| colo | The colocation facility name that the interconnect from the on-premises datacenter is peered with. | A valid Google Cloud [colocation facility code](https://cloud.google.com/interconnect/docs/concepts/colocation-facilities#locations-table) |
| function | The name of the resource type that a custom role is associated with. | Resource dependent |
| app-name | The name of the application using this service account | Resource dependent |
| \*-name | The name of a resource without its prefix. | Resource dependent |
| \*-description | A descriptive field to enhance the description of the resource. | Resource dependent |

Table 1.2 Naming conventions field values

### X 2.2.3 Resource hierarchy

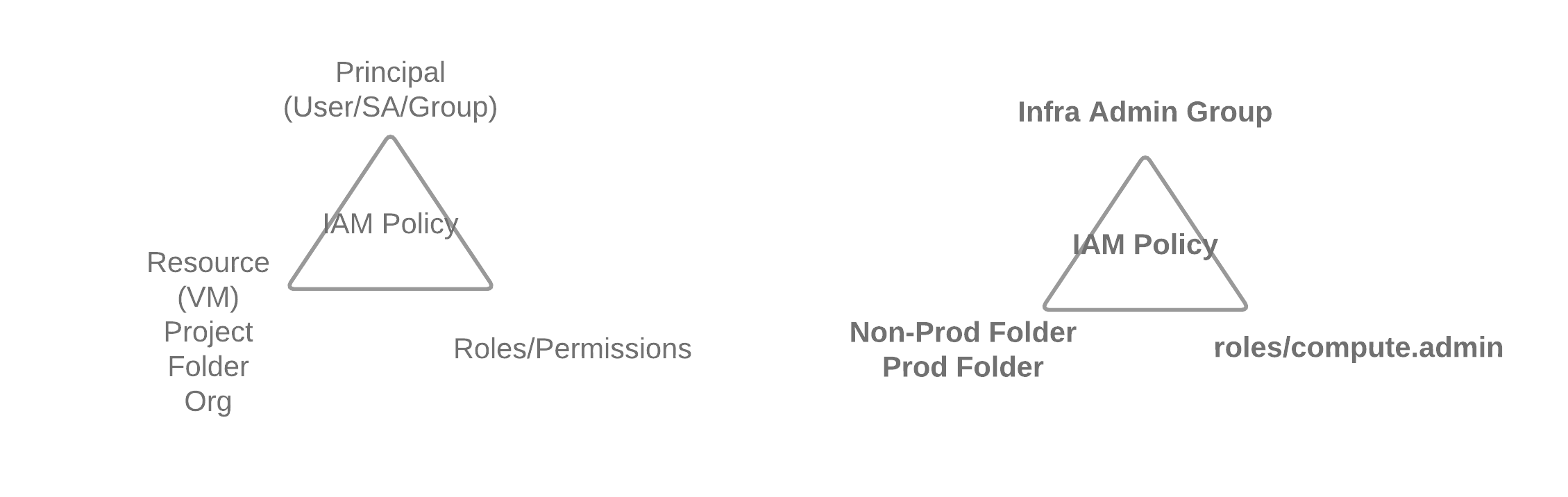


AMEX will be leveraging a strategy of many projects with folders to segment and group their projects by like-projects (i.e. projects which require similar organizational policies and user access).

* American Express has a primary root organization node at aexp.com which contains all of their GCP workloads.
* The first level of the folders will group different **departments** responsible and accountable for the applications ex GI.
* The second level of the folders will group different **environments** managed by the central IT team like prod, dev, test, Eng.
* The third level of the folders will group the **platforms** like Big Data, General Purpose and Ads. In prod, along with platforms Shared Service and Anonymized Data folder are added.
* The fourth level groups **LOBs** like Data Warehouse.
* The fifth level groups “like” applications by **functional owner** at the configuration level.
* Separate projects will be used to delineate each application and/or application component.
* Prod and Non Prod are set up similarly and are replica, so they will have very similar folders and project structures.
* Prod Network folder will contain a project for hosting the Shared VPCs for Prod environments. This node is under Prod to inherit prod policies.
* Non Prod Network folder will contain 3 host projects for hosting the Shared VPCs for Dev, Test and Engg environments. This node is under Test to inherit test folder policies
* Shared Services folder will contain a project for hosting the Shared VPC for Shared Services.
* Anonymized data folder contains projects for anonymized data.
* Logging will be centralized based on the environment i.e. all prod logs will be centralized in the logging project of prod. All test logs will be centralized in the logging project for test. Similar setup for Dev and Eng.
* Currently there are three major Project Types.
* Shared Data projects - used for data analysis and processing of data shared across multiple teams.
* BigQuery projects - For structured data analysis of data shared across multiple teams specifically with BigQuery.
* Dedicated team centric projects - For data analysis of data that cannot be shared across teams.
* We need to create two IAM projects(one for Prod and one for Non-Prod) that will contain all the Service Accounts.

Authorizations, policies, and constraints can be granted at each level of the resource management hierarchy. Any authorization, policy, or constraint is automatically inherited by all following child resource management layers. For example, if a policy is applied to the organization node, all folders and projects will inherit this policy, and if an authorization is applied at a deeper folder level, only the following folders and projects will inherit the authorization.

Authorizations are generally granted in the form: **principal** gets assigned a **role** to a **resource**, generally folders or a project, but in many cases can be granted to individual resources, like virtual machines.



Authorizations inherited through the hierarchy structure are additive (allow). **There is no way to revoke (deny) an authorization granted at a parent level to any following child resources.**

### 2.2.4 Regions and Zones

Currently all applications are hosted out of 2 US regions

### 2.2.5 Resource Quota

Google Cloud Platform enforces [quotas](https://cloud.google.com/compute/docs/resource-quotas) on resource usage for a variety of reasons. Quotas are managed at project level. For example, quotas protect the community of GCP users by preventing unforeseen spikes in usage.

Not all projects have the same quotas. As your use of GCP expands over time, your quotas may increase automatically. If you expect an upcoming increase in usage that exceeds your existing quotas, you can proactively request quota increases from the [quotas](https://console.cloud.google.com/compute/quotas) page in the Cloud Platform Console. To view all quota details and request increases, log in to the Cloud Console and select *IAM & Admin -> Quota*.

Capacity planning and quotas should be a recurring process that is repeated at regular intervals or any time a new application is being added into an existing project to ensure that the growth can smoothly proceed. Reasonable quota increase requests are usually processed within three business days. Large quota requests may take additional time to procure resources.

At the current time, no quota needs to be increased. We recommend proactively monitoring resource quota availability as new projects are planned and as customer projects scale.

### 2.2.6 Constraints with Organization Policies

The Organization Policy service gives AMEX centralized, programmatic control over their organization's Cloud resources. It provides a simple mechanism for AMEX to restrict allowed configurations across the entire Cloud Resource hierarchy. These policies are intended to provide effective preventative security controls around undesirable configurations. A word of caution not to confuse organization policies with IAM policies associated at the organization level.

Organization policies provide the following benefits:

* Policies can be set per project, per folder, or per organization.
* Policies are inherited down the resource hierarchy, and can place additional restrictions at any level on which an org policy can be set.
* Policies are not necessarily managed by the owner of the resource. Instead, policies are managed by your organization's policy administrator (IAM role: roles/orgpolicy.policyAdmin).
* This means that individual users and project owners cannot override policies.

AMEX should define a set of default Organization policies that define how [the constraints](https://cloud.google.com/resource-manager/docs/organization-policy/org-policy-constraints) are applied by their organization. Google recommends setting the following constraints:

| **Policy Constraint** | **Description** | **Value** |
| --- | --- | --- |
| gcp.resourceLocations | Resource Location Restriction | Allow <list> |
| serviceuser.services | Restrict allowed Google Cloud APIs and services  Note: this policy does not allow all the available services | Allow all |
| appengine.disableCodeDownload | GAE - Disable Source Code Download | true |
| cloudfunctions.allowedIngressSettings | GCF - Allowed ingress settings | Allow <list> |
| cloudfunctions.allowedVpcConnectorEgressSettings | GCF - Allowed VPC Connector egress settings | Allow <list> |
| cloudfunctions.requireVPCConnector | GCF - Require VPC Connector | true |
| sql.restrictPublicIp | Cloud SQL - Restrict Public IP access | true |
| compute.disableGuestAttributesAccess | GCE - Disable Guest Attributes of Compute Engine metadata | false |
| compute.disableInternetNetworkEndpointGroup | GCE - Disable Internet Network Endpoint Groups | true |
| compute.disableNestedVirtualization | GCE - Disable VM nested virtualization | true |
| compute.disableSerialPortAccess | GCE - Disable VM serial port access | true for non GKE projects |
| compute.disableSerialPortLogging | GCE - Disable VM serial port logging to Stackdriver | false |
| compute.requireOsLogin | GCE - Enables OS Login to VMs on all newly created Projects. This constraint prevents metadata updates that disable OS Login at the project or instance level | false (unless OSLogin becomes only path) |
| compute.requireShieldedVm | GCE - Requires that all new Compute Engine VM instances use Shielded disk images with Secure Boot, vTPM, and Integrity Monitoring options enabled | true |
| compute.restrictAuthenticatedGoogleConnection | GCE - VMs that are allowed to use Authenticated Google Connection | Allow all |
| compute.restrictCloudNATUsage | GCE - Restrict Cloud NAT usage | Allow all |
| compute.restrictDedicatedInterconnectUsage | GCE - Restrict Dedicated Interconnect usage in subnetworks | Allow all |
| compute.restrictDirectGoogleAccess | GCE - Restrict Compute Engine VMs that are allowed to use Direct Google Access | Allow all |
| compute.restrictLoadBalancerCreationForTypes | GCE - Restrict Load Balancer Creation Based on Load Balancer Types | Allow in:INTERNAL |
| compute.restrictNonConfidentialComputing | GCE - Restrict Non-Confidential Computing | Allow all |
| compute.restrictPartnerInterconnectUsage | GCE - Restrict Partner Interconnect usage | Allow all |
| compute.restrictProtocolForwardingCreationForTypes | GCE - Restrict Protocol Forwarding Based on type of IP Address (internal/external) | Allow INTERNAL |
| compute.restrictSharedVpcHostProjects | GCE - Restrict Shared VPC Host Projects | Allow all |
| compute.restrictSharedVpcSubnetworks | GCE - Restrict Shared VPC Subnetworks | Allow all |
| compute.restrictVpcPeering | GCE - Restrict VPC peering usage | Allow all |
| compute.restrictVpnPeerIPs | GCE - Restrict VPN Peer IPs | Allow all |
| compute.skipDefaultNetworkCreation | GCE - Skip default network creation | true |
| compute.storageResourceUseRestrictions | GCE - Compute Storage resource use restrictions (Compute Engine disks, images, and snapshots) | Allow all |
| compute.trustedImageProjects | GCE - Projects that can be used for image storage and disk instantiation for Compute Engine | Allow project\_id |
| compute.vmCanIpForward | GCE - Restrict VM IP Forwarding | Deny all |
| compute.vmExternalIpAccess | GCE - Restrict external IPs for VM instances | Deny all |
| compute.restrictXpnProjectLienRemoval | RM - Restricts users that can remove a Shared VPC project lien without organization-level permission | true |
| iam.allowServiceAccountCredentialLifetimeExtension | IAM - Allow extending lifetime of OAuth 2.0 access tokens to up to 12 hours | Deny all |
| iam.allowedPolicyMemberDomains | IAM - Domain users that can be added to Cloud IAM policies | Allow aexp.com |
| iam.disableServiceAccountCreation | IAM - Disable service account creation | Allow All |
| iam.disableServiceAccountKeyCreation | IAM - Disable service account key creation | Deny All |
| iam.disableServiceAccountKeyUpload | IAM - Disable uploading public key to Service Accounts | Deny All |
| iam.disableWorkloadIdentityClusterCreation | IAM - Disable Workload Identity Cluster Creation | false |
| iam.automaticIamGrantsForDefaultServiceAccounts | IAM - Disable Automatic IAM Grants for Default Service Accounts | false |
| gcp.detailedAuditLoggingMode | GCS - Detailed Audit Logging Mode (both, request and response are included in Cloud Audit Logs) | true |
| storage.retentionPolicySeconds | GCS - Retention policy duration in seconds | Allow all |
| storage.uniformBucketLevelAccess | GCS - Enforce uniform bucket-level access | true |
| storage.publicAccessPrevention | GCS - Restrict public access for all buckets and objects, both new and existing, under that resource. | true |

### 2.2.7 Labels

Following pages can be referenced for details on GCP Labels:

* [GCP labels overview](https://cloud.google.com/resource-manager/docs/creating-managing-labels)
* [How to: labeling resources](https://cloud.google.com/compute/docs/labeling-resources)
* [Using labels to organize GCP resources](https://cloudplatform.googleblog.com/2015/10/using-labels-to-organize-Google-Cloud-Platform-resources.html)
* [Most common use cases for labels - security marks, labels and tags](https://cloud.google.com/blog/products/gcp/labelling-and-grouping-your-google-cloud-platform-resources)

#### 2.2.7.1 Services currently support labels

Following services support labeling at the moment. Most recent list can be found [here](https://cloud.google.com/resource-manager/docs/creating-managing-labels#label_support):

* BigQuery
* Cloud Bigtable
* Dataflow
* Dataproc
* Cloud Deployment Manager
* Cloud Functions
* Cloud Healthcare API
* Cloud Key Management Service
* Pub/Sub
* Cloud Spanner
* Cloud SQL
* Cloud Storage
* Compute Engine
* Google Kubernetes Engine
* Cloud Run (fully managed)
* Networking
* Resource Manager (projects only)
* App Engine (both Flexible and Standard environments)

#### 2.2.7.2 Defining your Labels

AMEX has finalized [these](https://drive.google.com/file/d/1Egp84JmEkVSRrMeViVLgTRtgFV1X24DJ/view?usp=sharing) labels. Following table is a sample (for reference, not the finalized list) to define labels:

| **Label Key** | **Label Value** | **Owner** | **Target Location** | **Exception** |
| --- | --- | --- | --- | --- |
| amex-environment | prod, sb1, staging, dev, tools | finance | Project | No |
| amex-data-classification | na, low, moderate, high | legal | Buckets, Databases, PDs with GCE | No |
| amex-cost-center | fin-us, mkt-eu, it-jp | sre | Project | sandbox-folder |
| amex-team | Operator-team-oncall-alias (link to be added) | sre | Project | Non production environments. Non critical components. |
| component (expand for each component) | frontend, cache, application, database | finance | All | Optional |
| amex-app | [1:1 mapping with microservice] | finance | All | No. Exception for multi-tenant resources where there is no 1:1 mapping with app |
| amex-compliance | pci, hipaa |  | Project | Optional |

For highly regulated environments such as Amex, a best practice is to leverage labels across the Resource Hierarchy for both operational and compliance reasons. It is strongly recommended that both Mandatory and Optional Labels be standardized within Amex.

For change management purposes, a best practice is to capture changes to resources. This includes both Projects and Resources to quickly identify who made a given change and being able to link that change back to a specific change request (ITSM recordID). Using Google Asset Inventory, one can quickly identify a resource’s change history with proper label management.

| **Label Key** | **Label Value** | **Owner** | **Target Location** | **Exception** |
| --- | --- | --- | --- | --- |
| changed-by | john.smith\_aexp.com  terrafrom\_proj\_create\_001 | sre | All | No |
| record-id | 0000001 | sre | All | No |

Optional labels are also often used to provide operational transparency over resource requirements. Labels such as the following are often used:

| **Label Key** | **Label Value** | **Owner** | **Target Location** | **Exception** |
| --- | --- | --- | --- | --- |
| criticality | DR, HA, None | sre | All | Yes |
| retention | GCS retention standard | sre | GCS | Yes |
| backup | VM backup standard | sre | GCE | Yes |

Many enterprises define additional custom operational labels such as “state: active, readytodelete, archive '' or other labels that can be target identifiers for automation to perform some action.

NOTE: When using multiple classifications within the context of a single project, one should always use the highest classification of any resource at the Project level and classify resources to what they specifically contain. Google supports conditional access with [IAM at the resource level.](https://cloud.google.com/iam/docs/tags-access-control) This allows for granular access to resources based on labels. However, in practice, it is generally easiest to control access to Projects based on the overall Project classification.

#### 2.2.7.3 Use cases for labeling

* Identify microservices associated with the resource. This assumes that the resource maps 1:1 with the resource.
* Identify which environment this resource is associated with.
* Identify which team is responsible for this resource.
* Identify the data classification of any data stored in the system. Only applicable to stateful systems.

#### 2.2.7.4 Proposed solutions for label enforcement:

Currently, labels can not be enforced using org policies or IAM . Following solutions can be used:

**Proactive Governance**

* IAM permission to limited users for creating labels.
* [Terraform Validator](https://github.com/GoogleCloudPlatform/terraform-validator) - Automate label creation using centralized CI/CD pipeline and enforce labels using Terraform validator.

AMEX will use Terraform validator for labels as proactive governance to enforce labels in production.

**Reactive Governance**

* Stackdriver Alerts. This is not available out of the box but we can leverage user defined log-based metrics. [POC Needed]
* [Cloud Asset Inventory](https://cloud.google.com/asset-inventory/docs/overview) real time notification to trigger a cloud function that checks if the resources are properly labeled, then that cloud function could send an alert into some other system [POC Needed]
* Custom solution to enforce labels - A custom solution was implemented for a large customer to assign labels using Audit Logs -> PubSub -> Cloud Function, where the Cloud Function performs the remediation. In this solution, labels are stored in Firestore by folder id. Cloud Function fetches it and assigns it to all projects in the folder. Code [here](https://source.cloud.google.com/cloud-professional-services/usaa-foundations/+/master:auto-label/README.md).
* Custom solution to routinely scan all resources and enforce labels. [POC Needed]

### 2.2.8 Resource Provisioning

AMEX resources are being created using the terraform [Google Cloud Foundation Toolkit](https://cloud.google.com/foundation-toolkit). This practice will be further described in the Infrastructure as Code section.

### 2.2.9 Change Management

A mature Change Management governance process is a central requirement to satisfying most cybersecurity frameworks such as NIST 800-53, ISO 27001, etc. The process is responsible for controlling the lifecycle of all changes with minimum disruption to IT services. The objective is to ensure that all policies and approvals are met before system changes are implemented.

Within the context of Immutable Infrastructure environments, Change Management needs to be viewed from three different perspectives: **manual change**s to systems, **emergency changes** to systems, and **infrastructure as code changes** to systems.

Manual changes are day to day operational changes to systems that are created as a result of a change request. Change requests are submitted into an IT Service Management system such as ServiceNow and follow an approval process before being permitted to execute. A human then makes the necessary changes to the system to fulfill the change request requirements.

Emergency changes are changes that are typically made as part of an unexpected event requiring an urgent response. This can be as a result of an Incident Management action or other unforeseen event. Such changes may not be able to request a change record ID from IT Service Management at the time of the event, but still require some sort of follow-up to document said changes.

Infrastructure as Code (IaC) is a formal resource provisioning practice that establishes necessary resources and configurations to deliver a service. The practice involves managing infrastructure or services as version controlled code. The versioning practice involves significant testing and approvals before code can be released into production. Strict Change Management practices are typically built into IaC processes. For best outcomes, the IT Service Management system can trigger IaC directly as part of a Service Catalog, fulfilling all change management requirements via automation. For Immutable Infrastructure environments, IaC is the primary approach to all changes to that environment.

#### 2.2.9.1 Tracking Changes within GCP

Foundational to all Change Management frameworks is an effective means by which IT services are requested, approved, and finally executed. No change should take place within an IT environment without having a documented trail ensuring that at minimum dual controls (the person making the change or approving the change is not the same person requesting the change) are used for every configuration change.

Change management always begins with a change request. This request establishes a traceable record for the underlying justification for a change to IT systems. Each request creates a change ID or “record-id” that is a unique identifier for the change over its lifecycle. This record-id should be viewed as an auditable identifier for why a change was needed, who approved it, and who implemented it. One of the biggest challenges in managing a complex IT environment is that current practices for tracking changes are only “top-down”. The change management system of record maintains visibility to all approved changes. But, the underlying system assets rarely provide their own view of why actual changes to the systems were made. This creates configuration misalignment and changes creep in between what the source of record expects to be implemented and what is really implemented.

To counter this problem, an effective strategy is to ingrain Change Management records (change id/record id, changed by) into the underlying assets themselves. All Public Cloud vendors support tagging or labeling resources as part of their capabilities. The tags/labels are metadata that provide operational visibility to the underlying resource.

As [suggested in the Labels section](#kix.o2leqaxzkyto), establishing mandatory labels for change management purposes can help enterprises stay within compliance from the resource perspective. One can quickly audit all resources against missing change records. Such a process would immediately identify non-authorized changes within an organization, ensuring continuous compliance to regulatory requirements. Moreover, leveraging Cloud configuration history tools such as “[GCP Cloud Asset Inventory](https://cloud.google.com/asset-inventory/)” can quickly identify how IT assets got to their current state from the resource side (bottom-up).

# 3. Identity and access management

Identity and access management (IAM) is the discipline that enables approved identities (both person and non-person) to access appropriate resources at the appropriate times for approved reasons. IAM is the central information system security control used within GCP to ensure that users have appropriate access to authorized services or resources in a manner that is consistent with an enterprise’s access policies. Proper enterprise access policies ensure that adequate controls are in place to validate that users are who they claim to be, are granted the bare minimum access privileges required to perform their job function, are prevented from being able to individually escalate their own privileges, and whose activities are accounted for and audited.

Identity and access management (IAM) policies are available for configuration in the Cloud Console. IAM roles are available for users, groups of users, and service accounts that allow granular control of permissions to access resources

## 3.1 Current Architecture

Amex has an existing IAM practice that they use on Amazon Web Services and are leveraging into Google Cloud. It has its root in Amex’s historic Red Hat on-premise PaaS. Critical in that practice is to maintain uniform consistency in naming conventions across all cloud environments. Internally, Amex uses the following conventions to identify their different environments.

| E0 | Lab environment |
| --- | --- |
| E1 | Dev environment |
| E2 | Test/QA environment |
| E3 | Prod environment |

Amex has identified a number of groups as they use them on AWS. These groups are implemented in a particular manner as a result of AWS not supporting the folder construct. As such, many of the Groups are very “AWS Account” centric and don’t leverage organization, environment, business unit and project level access inheritance. Within GCP, it is recommended that Groups be identified considering how access to specific resources is organized across the folder structure identified within the Resource Management section to prevent an overload of Groups as GCP projects proliferate. The table below lists the identified roles and PRC groups for GCP Foundation.

## 3.2 Cloud Identity

Google Cloud Identity is the product used for managing users, groups, and domain-wide security settings for G Suite and Google Cloud Platform. Cloud Identity is tied to a unique DNS domain that needs to be enabled for receiving email (e.g., has an appropriate MX configured) so that users and groups configured with responsibilities in GCP can receive generated notifications.

Cloud Identity configurations are made in the Admin Console. ExistingG Suite customers can use their G Suite Admin Console for Cloud Identity. Customers without an existing G Suite account can create a Cloud Identity in the "IAM" section of the GCP Cloud Console.

AMEX will use aexp.com for the identities of users and groups configured in GCP.

 Domains

The customer selects a primary domain and has the option to create a secondary domain.

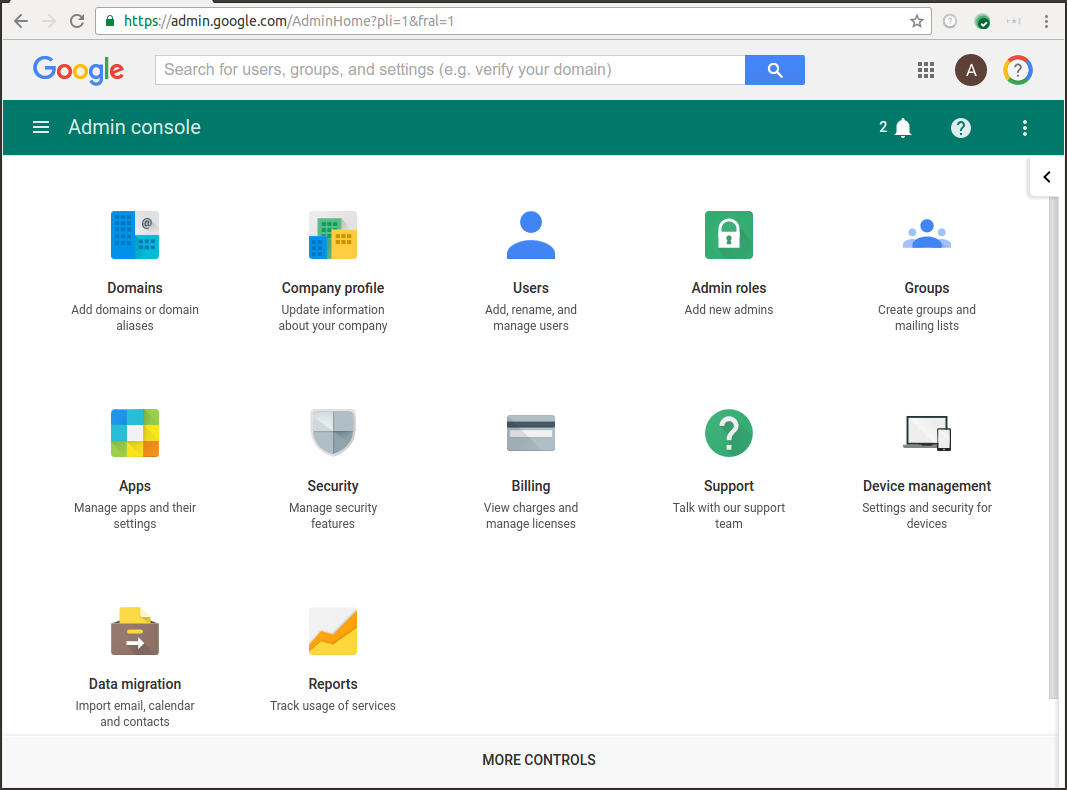


AMEX will use only the primary domain.

### 

### 3.2.1 Admin Console

The Google Admin Console (<https://admin.google.com/>) is the centralized console to manage a Google instance's users, groups, and security settings. For G Suite customers, the Admin Console provides additional functionality to configure their user’s G Suite experiences.

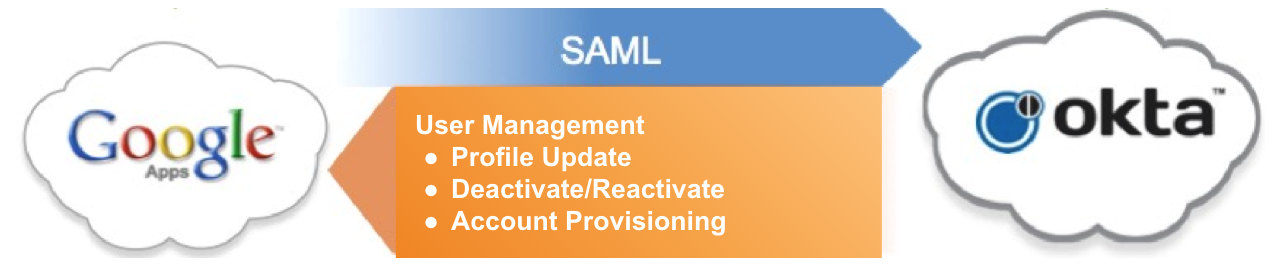


### 3.2.2 Identity

Identity for American Express users is provided by OKTA, which is integrated with an on-premise Active Directory (AD). Identities are synchronized from Active directory to Okta and then from Okta to Google Cloud Identity and this process can take upto 4 hours.

OKTA is used as the Identity Provider (IdP) and for Single Sign-On (SSO). Access to OKTA is currently only available within the American Express internal network. This requires users to connect into Amex’s on-prem network using a client VPN before being able to access Google Cloud. Providing access without a client VPN is being considered, but is outside of the scope of the current design.

Google strongly recommends that all users be verified with 2FA (2 Factor Authentication). The use of usernames and passwords is simply not secure enough as a mechanism to prevent hijacking of user identities. OKTA supports 2FA, and this design delegates the responsibility of 2FA to OKTA as part of the IdP Single Sign-ON (SSO) process. This design expects that OKTA will validate all non-Super Admin user identities with additional factors beyond username/passwords.



### 3.2.3 Super Admin

This Google instance will be managed by the engineering team at AMEX. The engineering team will designate limited members with the Super Admin role managed and administered by the Google instance.

[Super Admins](https://support.google.com/a/answer/2405986) have the ability to set permissions (and passwords) for any user managed in their Google instance's Admin Console. Within GCP, a Super Admin also has the ability to establish [Organization Admins](https://cloud.google.com/resource-manager/docs/creating-managing-organization#adding_an_organization_admin).

To prevent abuse or an interruption in administrational capacity due to an inability for a Super Admin to perform their responsibilities, Google recommends that more than one Super Admin be created. AMEX will have at least 2 <@TODO Recommendation 4. To be validated with AMEX> Super Admins:

| **Super Admin user name** | **Full name** |
| --- | --- |
| [daniel.n.greenfield@aexp.com](mailto:daniel..n.greenfield@aexp.com) | Daniel Greenfield |
| [abhjeet.gandharwar1@aexp.com](mailto:abhjeet.gandharwar1@aexp.com) | Abhjeet Gandharwar |

It is recommended to set up auditing and alerting capability to prevent misuse of Super Admin role wherein an AMEX Super Admin will have the ability to impersonate any amex.com user in GCP by resetting their password, and thus accessing amex.com GCP projects. The Admin Console provides an auditing and alerting capability to identify if and when a Super Admin has performed a password change. In the Admin Console, this capability is configured under 'Reports' > 'Manage alerts', using the 'User's password changed' alert. Google recommends setting this alert on Google instance.

Super Admin Account Best Practices:

* Create a new email address that is not specific to a particular user as the Google Workspace or Cloud Identity super admin account. Secure access for this user with 2F and physical security keys.
* Create a private Google Cloud administrator group in your Google Workspace or Cloud Identity super admin account. Add your organization administrator users to this group, but not your super admin user. Grant this group the Organization Administrator IAM role or a limited subset of the role's permissions.
* Enforce a short sign-in period for admin account <https://support.google.com/a/answer/7576830?hl=en>
* Use Google Cloud's operations suite to [set up alerts](https://www.youtube.com/watch?v=LdGqmnoowc8) that will notify you when a [SetIamPolicy()](https://cloud.google.com/resource-manager/reference/rest/v1/projects/setIamPolicy) API call is made. This will send an alert when anyone modifies any IAM policy.
* Ensure that the organization administrators are familiar with the super admin [account recovery process](https://support.google.com/a/answer/33561?hl=en). This process will help you recover your account in the event that super admin credentials are lost or compromised.
* Follow [security best practices for super admin accounts](https://support.google.com/a/answer/9011373?hl=en)

### 3.2.4 Organization Admin

Organization Admin has access to administer all resources belonging to the organization. The Organization Admin role is the most permissive role inside GCP. Organization Admins can grant IAM roles to team members so that they can access an organization's resources and APIs. Organization Admin can grant roles to a Google Account email, a Google Group, a service account, or a G Suite domain.

It is also recommended to remove the project creator role from org node after it is created. This will limit project creation to just the Organization Admin.

<@TODO: Email Ids and Fill name to be replaced with Organization Admins>

| **Organization Admin username** | **Full name** |
| --- | --- |

| [orgadmin@aexp.com](mailto:superadmin@hdsupply.com) | Organization Admin |
| --- | --- |
| Other Org Admin accounts |  |

Org admins will login via Okta, use 2FA and will have appropriate ADS PRC groups with admin logging turned on.

### 3.2.5 Users

#### 3.2.5.1 Provisioning

Admin Console allows Super Admins to provision [users](https://support.google.com/a/answer/179832) either manually or using an automated process by reading from an organizational LDAP or AD system.

For AMEX, users and groups are synchronized with Google Cloud Identity using an OKTA Workspace Sync tool scheduled every 6 hours.

#### 3.2.5.2 Authentication

In the Admin Console, a Google instance needs to be configured to use either Google's authentication system or an [SAML-based Single Sign On (SSO) system](https://support.google.com/a/answer/60224) for authentication. While a Google instance will permit the addition of secondary DNS domains for user management purposes, Google recommends using only a single domain within an instance for authentication as it minimizes effort in managing user accounts and permissions.

Google strongly recommends using an authentication system that is capable of enforcing a second factor, especially one that supports the use of a Security Key as they are much harder to phish. Google's authentication system does support 2-Step Verification and Security Keys. It is important that Admins do not have access to the second factor configurations in order to prevent a malicious Admin from impersonating a user by merely changing their primary password.

Google strongly recommends that all users be verified with 2FA (2 Factor Authentication). The use of usernames and passwords is simply not secure enough as a mechanism to prevent hijacking of user identities. OKTA supports 2FA. This AMEX implementation design delegates the responsibility of 2FA to OKTA as part of the IdP Single Sign-ON (SSO) process. It is expected that OKTA will validate user identities with additional factors beyond username/passwords.

AMEX is using OKTA as the Identity Provider (IdP) and for Single Sign-On (SSO). Access to OKTA is currently only available within the American Express internal network. This requires users to connect into Amex’s on-prem network using a client VPN before being able to access Google Cloud. Providing access without a client VPN is being considered, but is outside of the scope of the current design.

#### 3.2.5.3 Conflicting accounts

[Conflicting accounts](https://support.google.com/a/answer/185186?hl=en) may be created by the user provisioning process in the Admin Console. They occur when an email address used to provision a new user in a Google instance (managed by your organization) is the same email address as is used by a consumer (i.e., personal) Google account. If a conflicting account is created during the provisioning process, the next time the user logs into their consumer account they will be presented with additional information about how to resolve the conflicting account. The user will be directed to this Google Help Center article for further information: <https://support.google.com/accounts/troubleshooter/1699308?hl=en>

Because only AMEX’s IT group will provision users into Cloud Identity, no relevant conflict accounts have been identified.

## 

### 3.2.6 Groups

Google recommends using [groups](https://support.google.com/a/answer/33343?hl=en) to grant roles and permissions to users. Managing access to GCP projects via groups is easier than managing access via individual users, as once project owners have properly associated groups with roles in a GCP project, granting and revoking a user's access to multiple projects can be managed centrally in the Admin Console. Additionally, group membership is auditable through the Admin Console.

For organizations that adhere to the principle of least privilege, groups can be created and associated with narrower levels of access to the services within a GCP project.

Groups for managing access to Google Cloud Platform will be provisioned using OKTA in AMEX.

To be able to easily associate a group with the role that membership to the group grants, Google recommends establishing a naming convention for group addresses similar to the following format: ***gcp-{team name}-{role}<-{environment}><-{project}>@aexp.com***

For example, gcp-sre-engineers-e0@aexp.com would be used to grant access to Engineers working on the SRE.

At AMEX, following groups have been identified for cloud roles and access management:

| **#** | **Role Type** | **Description** | **Role Name** | **Binding** | **PRC Group** |
| --- | --- | --- | --- | --- | --- |
| 1 | Cloud Security Admin | This role will be responsible for Cloud Security Policy  This Team will also PoC general security capabilities control not covered  by network or IAM. | org\_secu  rity\_admin | PRC\_group  : lab  TFE\_servic  e\_account:  org | PRC-AXP-BAE2-AppAdminGCP-CloudSec |
| 2 | Cloud Security Poweruser |  | org\_secu  rity\_powe  ruser | PRC\_group  : org | PRC-AXP-BAE2-AppAdminGCP-CloudSec |
| 3 | Infosec Audit | This role will have list access to all resources, it does not need to access  data within a resource. | org\_infos  ecaudit\_v  iewer | PRC\_group  : org | PRC-AXP-AAE3-Compl-SecGCPInfosecAudit |
| 4 | IAM  Admin | This team is responsible for all okta and identity integration and will do  PoCs related to IAM tools(cloudknox,etc,access analysis) | org\_iam\_  admin | PRC\_group  : lab  TFE\_servic  e\_account:  org | PRC-AXP-BAE2-AppAdminGCP-IAM |
| 5 | IAM  Poweruser |  | org\_iam\_  poweruser | PRC\_group  : org | PRC-AXP-BAE2-AppAdminGCP-IAM |
| 6 | Network Security Eng  Admin | This team will do Firewall administration and will be responsible for VPC  Firewall and PaloAlto appliances. This team will also manage Service  Perimeters and Access Context Manager. | org\_nets  ec\_admin | PRC\_group  : lab  TFE\_servic  e\_account:  org | PRC-AXP-BAE2-AppAdminGCP-NetSec  PRC-AXP-BAE2-AppAdminGCPNetSecOps |
| 7 | Network  Security  Eng  poweruser |  | org\_nets  ec\_power  user | PRC\_group  : org | PRC-AXP-BAE2-AppAdminGCP-NetSec  PRC-AXP-BAE2-AppAdminGCPNetSecOps |
| 8 | SIEM  Admin | This team will responsible for all stackdriver audit related actvities  including pub/sub subscription to stackdriver at org level. | org\_siem  \_admin | PRC\_group  : lab  TFE\_servic  e\_account:  org | PRC-AXP-BAE2-AppAdminGCP-SIEM |
| 9 | SIEM  Poweruser |  | org\_siem  \_powerus  er | PRC\_group  : org | PRC-AXP-BAE2-AppAdminGCP-SIEM |
| 10 | IR Admin | The incident response team will be responsible for Security tooling  (security center)excluding network and iam. | org\_ir\_ad  min | PRC\_group  : lab  TFE\_servic  e\_account:  org | PRC-AXP-BAE2-AppAdminGCPInfosecMonitori  ng |
| 11 | IR  Poweruser |  | org\_ir\_po  weruser | PRC\_group  : org | PRC-AXP-BAE2-AppAdminGCPInfosecMonitori  ng |
| 12 | Crypto  Admin | This team is responsible for PKI infrastructure including KMS,  certificates, Customer managed keys, and related capabilities. | org\_crypt  o\_admin | PRC\_group  : lab  TFE\_servic  e\_account:  org | PRC-AXP-BAE2-AppAdminGCP-Crypto |
| 13 | Crypto  Poweruser |  | org\_crypt  o\_poweru  ser | PRC\_group  : org | PRC-AXP-BAE2-AppAdminGCP-Crypto |
| 14 | CloudEng  Admin | This team is responsible for overall cloud operation and integration,  enablement of services, resource management, iam management,  troubleshooting all resource types, creating and testing modules for all  resource types | org\_coref  oundation  \_admin | PRC\_group  : org?  TFE\_servic  e\_account:  org | PRC-AXP-BAE2-AppAdminGCP-CF |
| 15 | CloudEng  Poweruser |  | org\_coref  oundation  \_powerus  er | PRC\_group  : org | PRC-AXP-BAE2-AppAdminGCP-C |
| 16 | DDI  Admin | This team is responsible for CloudDNS. | org\_ddi\_  admin | PRC\_group  : lab  TFE\_servic  e\_account:  org | PRC-AXP-AAE2-AppAdminGCP-DDI |
| 17 | DDI  poweruser |  | org\_ddi\_  poweruser | PRC\_group  : org | PRC-AXP-AAE2-AppAdminGCP-DDI |
| 18 | CloudOps  Admin | This team is responsible for troubleshooting and operations activities | org\_cloud  ops\_admin | PRC\_group  : lab  TFE\_servic  e\_account:  org | PRC-AXP-PUE2-AppAdminGCP-Dev-Ops |
| 19 | CloudOps  Poweruser |  | org\_cloud  ops\_pow  eruser | PRC\_group  : org | PRC-AXP-PUE2-AppAdminGCP-Dev-Ops |
| 20 | Billing  Admin | This team is responsible for billing, chargeback and pulling billing data | org\_billin  g\_admin | PRC\_group  : lab  TFE\_servic  e\_account:  org | PRC-AXP-BAE2-AppAdminGCP-Billing |
| 21 | Billing  Poweruser |  | org\_billin  g\_viewer | PRC\_group  : org | PRC-AXP-BAE2-AppAdminGCP-Billing |
| 22 | Network  Admin | This team will be responsible for all non-security related aspects of  network management including VPCs, subnets, routers, will work with  netsec for internet gateways, nat, etc.  This team is also responsible for LoadBalancers. | org\_netw  ork\_admin | PRC\_group  : lab  TFE\_servic  e\_account:  org | PRC-AXP-BAE3-AppAdminAWS-DCE  PRC-AXP-BAE3-AppAdminAWS-GNOC |
| 23 | Network  Poweruser |  | org\_netw  ork\_powe  ruser | PRC\_group  : org | PRC-AXP-BAE3-AppAdminAWS-DCE  PRC-AXP-BAE3-AppAdminAWS-GNOC |
| 24 | CloudSolu  tions  poweruser | This team will assist with onboarding application team to cloud, they will  help with app deployment, testing, loadbalancers,etc | org\_soluti  ons\_pu  E0:  E1-E3: | PRC\_group  : lab  TFE\_servic  e\_account:  lab | PRC-AXP-PUE2-AppAdminGCP-Dev-Eng |
| 25 | CloudQE  Poweruser | This team will perform testing of capabilities provided to application and  application platform teams.  This team should only need poc/lab access to create all non-infrastucture  resource types(dataproc,bigquery,gce,etc) and do load/performance  testing on the same. | org\_qualit  y\_pu  E0:  E1-E3: | PRC\_group  : lab  TFE\_servic  e\_account:  lab | PRC-AXP-PUE2-AppAdminGCP-TESTEng |
| 26 | CICD  Poweruser | This team manages CICD pipline and will be responsible for pushing  artifacts, deploying any CICD related tools. | org\_cicd\_  pu  E0:  E1-E3: | PRC\_group  : lab  TFE\_servic  e\_account:  lab | PRC-AXP-PUE2-AppAdminGCP-CICD |
| 27 | Product  poweruser | This group is for project managers and needs the ability to view the  presence of resources and billing. | org\_prod  uct\_pu  E0:  E1-E3: | PRC\_group  : org  TFE\_servic  e\_account:  org | PRC-AXP-PUE2-AppSuppGCP-CEProduct |

For details on Groups, Roles and Permissions, @TODO Please add the confluence link of the sheet containing a list of groups and GCP pre-defined roles.

## 3.3 Authorization

Authorization to systems is achieved by assigning an Okta-sourced user Organization Unit (OU) to individuals. These OUs assign users to GCP groups which themselves are provided GCP roles that provide granular permissions required to perform their business functions. GCP Groups will be the central means by which users will be granted privileges to resources and services within the environment. As a member of a GCP Group, the user will be granted all roles and permissions assigned to said GCP Group. As a best practice, GCP users should never be explicitly granted roles directly.

## 3.3.1 IAM Roles

Permissions to all resources are granted by setting policies that grant roles to a user, group, or service account. To provide more granular controls in lieu of the legacy roles ([owner, editor, and viewer](https://cloud.google.com/iam/docs/understanding-roles#primitive_roles)), you can leverage the groups created and assign them to one or more roles.

Roles should be investigated as applications are migrated into the Google Cloud environment and users are granted access to them. Please refer to this [link](https://cloud.google.com/iam/docs/understanding-roles) for the available list of roles.

GCP supports 3 types of roles within its environment: Primitive, Predefined and Custom.

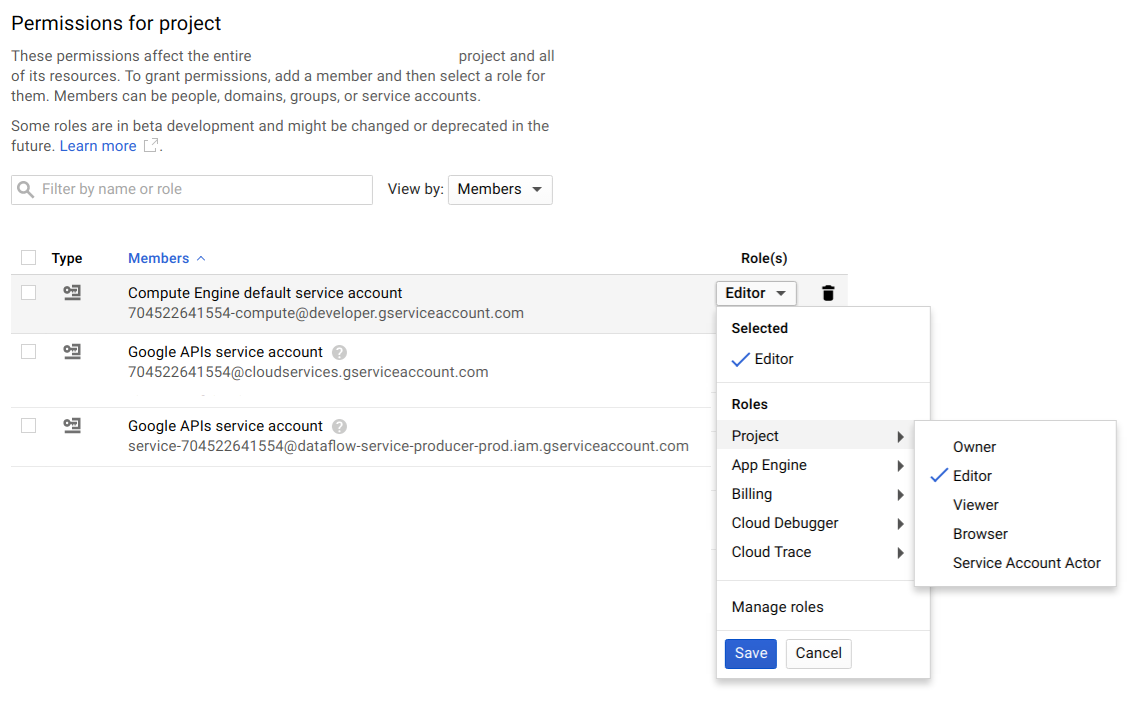
It is strongly recommended not to use any Primitive roles that have write permissions (Create, Updated, Delete) within production environments. These roles were created within GCP before the IAM service was developed. It is strongly recommended that Predefined or Custom roles be utilized to effectively manage user, group or service account permissions in production environments (with the exception of Viewer). Least privilege access is a fundamental security practice that is very strongly advised by Google and mandated in all cybersecurity frameworks.

Please see details on AMEX Roles on <@TODO: link to IAM template> sheet. In future, additional roles will be created for application development teams.

## 3.3.2 Project Permissions

Once a project is created, users need to be granted roles for resource usage. However, Google recommends leveraging groups for these role assignments. To accomplish this, first identify each set of roles that need to be individually granted to users for the given project. For each of these roles, work with the Google instance's Organization Admin to establish or identify existing appropriate groups for the role, keeping in mind that groups should be named according to the designated naming scheme.

The project creator should add the appropriate groups and associate the correct roles for them using terraform in the IAM module.



Cloud Resource Manager also offers organization-wide roles that may be applied and inherited onto all projects. Google recommends such roles for certain groups, for example, Network Admins and Security Admins. AMEX has finalized roles for admins. Please see details <@TODO: link to template>.

## 3.3.3 Access Control Auditing

AMEX will be deploying immutable infrastructure as their default approach. What this means is that very few users should have CUD (Create, Update, Delete) access within Production (E4) environments. Other environments will be more permissive for all CRUD (Create, Read, Update, Delete) operations. However, even in those environments, users and groups should generally be managed in such a way as to limit unnecessary access to underlying resources that are not needed by said teams to perform their jobs. An effective operational strategy is to have the Organization Admin act as the group role administrator. When teams require additional roles to accomplish their jobs, the Organization Admin reviews and enables these privileges as needed. However, the Organization Admin would still be required to provide the least amount of privileges needed to individual teams to perform their business functions. They should be the ones responsible for maintaining least privilege requirements on the environment, The following Google tools can help an Organization Admin in maintaining IAM configuration optimizations across their environment:

[Policy Intelligence](https://cloud.google.com/policy-intelligence) is an automated toolset used to assist security teams to keep the GCP environment secure and help manage IAM permissions. The tools will let you know of users who have roles that aren’t fully utilized based on resource usage and access patterns, can help troubleshoot access issues, and will allow you to simulate policy changes prior to rollout to ensure minimal issues.

[IAM Recommender](https://cloud.google.com/iam/docs/recommender-managing) uses machine learning to make smart access control recommendations that can help admins remove unwanted access to GCP resources. IAM Recommender can help reduce attack surface by analyzing users’ access patterns and recommending less permissive roles or full on revocation if unused for 90 days.

[Policy Troubleshooter](https://cloud.google.com/iam/docs/reference/policytroubleshooter/rest) is used when a user is denied access to a resource. Policy Troubleshooter enables security administrators to understand why requests were denied by visualizing the policies around the API calls. This allows administrators to see which policies would block the call in question, and can reduce time to resolution on IAM access issues.

## 3.3.4 Service Account

A [service account](https://cloud.google.com/compute/docs/access/service-accounts) is a specialized account that can be used by services and applications running on your Google Compute Engine (GCE) instance to interact with other GCP APIs. These are similar to “Programmatic Users” supported on other Public Cloud Platforms. Applications can use service account credentials to authorize themselves to a set of APIs and perform actions within the permissions granted to the service account and virtual machine instance. Additionally, GCE instances, including instances used for Google Container Engine, use service accounts to interact with other Google services and their underlying APIs.

Example Service Accounts

The following is an illustrative example of the roles/permissions of the [Dataflow](https://cloud.google.com/dataflow/docs/concepts/security-and-permissions#controller_service_account) service accounts that may be required for each environment. There will be additional service accounts depending on the application and GCP services being utilized.

|  | **Dataflow Service Accounts** | **Roles/Permissions** |
| --- | --- | --- |
| **E0** | svc-example@e0-example-1234.iam.gserviceaccounts.com | BigQuery Admin  Cloud Storage Admin  Cloud Pub/Sub Publisher/Subscriber/Viewer |
| **E1** | svc-example@e1-example-1234.iam.gserviceaccounts.com | BigQuery Admin  Cloud Storage Admin  Cloud Pub/Sub Publisher/Subscriber/Viewer |
| **E2** | svc-example@e2-example-1234.iam.gserviceaccounts.com | BigQuery Admin  Cloud Storage Admin  Cloud Pub/Sub Publisher/Subscriber/Viewer |
| **E4** | svc-example@e4-example-1234.iam.gserviceaccounts.com | BigQuery Admin  Cloud Storage Admin  Cloud Pub/Sub Publisher/Subscriber/Viewer |

#### 3.3.4.1 Using Service accounts outside GCP

For accessing and using a service account outside the GCP environment it is recommended to use a workload identity pool as a container for a collection of external identities.

A project can contain multiple workload identity pools, and each pool can have access to different resources. This lets you follow the principle of least privilege by grouping related identities in the same pool, and then granting them fine-grained access to resources.

In general, we recommend creating a new pool for each non-Google Cloud environment that needs to access Google Cloud resources, such as development, staging, or production environments.

#### 3.3.4.2 Using short lived service accounts credentials

Service accounts can use short-lived credentials to authenticate calls to Google Cloud APIs and other non-Google APIs. Short-lived credentials have a limited lifetime, with durations of just a few hours or shorter. Short-lived service account credentials are useful for scenarios where you need to grant limited access to resources for trusted service accounts. They also create less risk than long-lived credentials, such as service account keys.

#### 3.3.4.3 Service account impersonation

You can let other users or service accounts impersonate a service account. This allows members and resources to impersonate, or act as, an Identity and Access Management (IAM) service account. Google’s recommendation is to establish a well defined logging and tracking mechanism before using service account impersonation as a primary way to use elevated access.

A cloud audit log entry looks like this when a user runs with the impersonate-service-account flag:

| authenticationInfo: {  principalEmail: "xxxxx@xxxx" ----> This has the service account addresss  serviceAccountDelegationInfo: [  0: {  firstPartyPrincipal: {  principalEmail: "xxxxx@xxxx" -------> This has the user's email address   }  }  ]  } |
| --- |

#### 3.3.4.4 Service account best practices

Following best practices are recommended to be implemented while creating, using and managing service accounts.

| **Topic** | **Best Practices** |
| --- | --- |
| Default Service Accounts | Avoid using the Default Compute Engine service account. Use dedicated custom service-accounts for running VM’s with minimal required permissions |
| Service Account Usage | Specify who can act as service accounts. Users who are Service Account Users for a service account can indirectly access all the resources the service account has access to. Therefore, be cautious when granting the serviceAccountUser role to a user. |
| Least Privilege | Grant the service account only the minimum set of permissions required to achieve their goal. |
| Per Service Account | Create service accounts for each service/use case with only the permissions required for that service. |
| Centralized Service Account Project | Create all shared service accounts within a single shared project, making it easier to manage and audit. |
| Naming | Define a naming convention for your service accounts. Use the display name of a service account to keep track of the service accounts. When you create a service account, populate its display name with the purpose of the service account. |
| Key Rotation | Implement processes to automate the rotation of user-managed service account keys. Take advantage of the IAM service account API to implement key rotation. |
| Auditing | Regularly check service account permissions to make sure they are up-to-date. Audit service accounts and keys using either the serviceAccount.keys.list() method or the Logs Viewer page in the console. |
| Deletion | Do not delete service accounts that are in use by running instances on App Engine or Compute Engine unless you want those applications to lose access to the service account. |
| Service Accounts Keys | Avoid creating long lived service account keys and always have short lived service account access |
| VPC Service Controls | Utilize VPC Service Controls to create a perimeter around who can authenticate with Google services |
| Identities | Use workload and VM identities wherever possible. GKE workload identity is the recommended way for GKE applications to consume services provided by Google APIs. This is enabled by configuring a Kubernetes service account to act as a Google service account. Any Pods running as the Kubernetes service account then use the Google service account to authenticate to cloud services. |

Google recommends that each use case instance that needs to call a Google API should run as a service account with the minimum permissions necessary for that instance to do its job. In practice, this means service accounts for GKE, Cloud Functions or VM instances should be configured with the following process:

1. Create a new service account rather than using the Compute Engine default service account.
2. Grant IAM roles to that service account for only the resources that it needs.
3. Configure the instance to run as that service account.
4. Grant the instance the https://www.googleapis.com/auth/cloud-platform scope to allow full access to all Google Cloud APIs, so that the IAM permissions of the instance are completely determined by the IAM roles of the service account.

**AMEX Decision:**

1. Service accounts will be used extensively and centralized in a single project
2. There is a desire to avoid using Service Account impersonation, even in break glass scenarios, if used, then audit logs during impersonation should include both actor and impersonator information
3. Service accounts will be used for both the Customer Tracking and Customer Service applications.
4. American Express has decided to use VM and Workload identity for all their workloads. <@TODO To be validated>
5. AMEX has decided to service account impersonation for break glass access. There will be two custom roles created for application elevated access and organisation elevated access. <@TODO To be validated>
6. Hashicorp Vault will be used to maintain and manage a majority of Service Accounts within AMEX.

## 

## 3.3.5 Service Account Key Management

All service account keys used in the GCP environment leverage AES-256 symmetric encryption. There are two approaches to key management — Google-managed or user-managed keys. Google App Engine supports Google-managed keys, which are automatically maintained and rotated daily. AMEX will leverage Google-managed keys wherever possible.<@TODO: TO BE VALIDATED>

## 3.3.6 Cloud IAM Policies with Conditions

Cloud IAM policies allow you to add context to the role based on various context conditions. Cloud IAM policies comprise one or more role bindings, which have the following structure:

| "bindings": [  {  "role": ...  "members": ...  "condition": ...  },  ... ] |
| --- |

Condition objects are optional, and each role binding can contain zero or one condition. A role binding with no condition object is considered as an unconditional grant, as no condition check is necessary. The condition object has the following structure:

| "condition": {  "title": ...  "description": ...  "expression": ... } |
| --- |

Title is required, but the description is optional. The expression field is required. It defines an attribute-based logic expression using a subset of the [Common Expression Language (CEL)](https://opensource.google/projects/cel).

Google recommends using the [Cloud IAM condition](https://cloud.google.com/iam/docs/conditions-overview) feature to grant elevated access for user accounts or service accounts for scenarios like time bound role/access to users/service accounts. This can also be used as a way to provide time bound access to groups or people like contractors etc.

AMEX has decided that Service account impersonation will not be used as a primary method for elevated access, IAM conditions or groups expiration features will be used for time bound access.

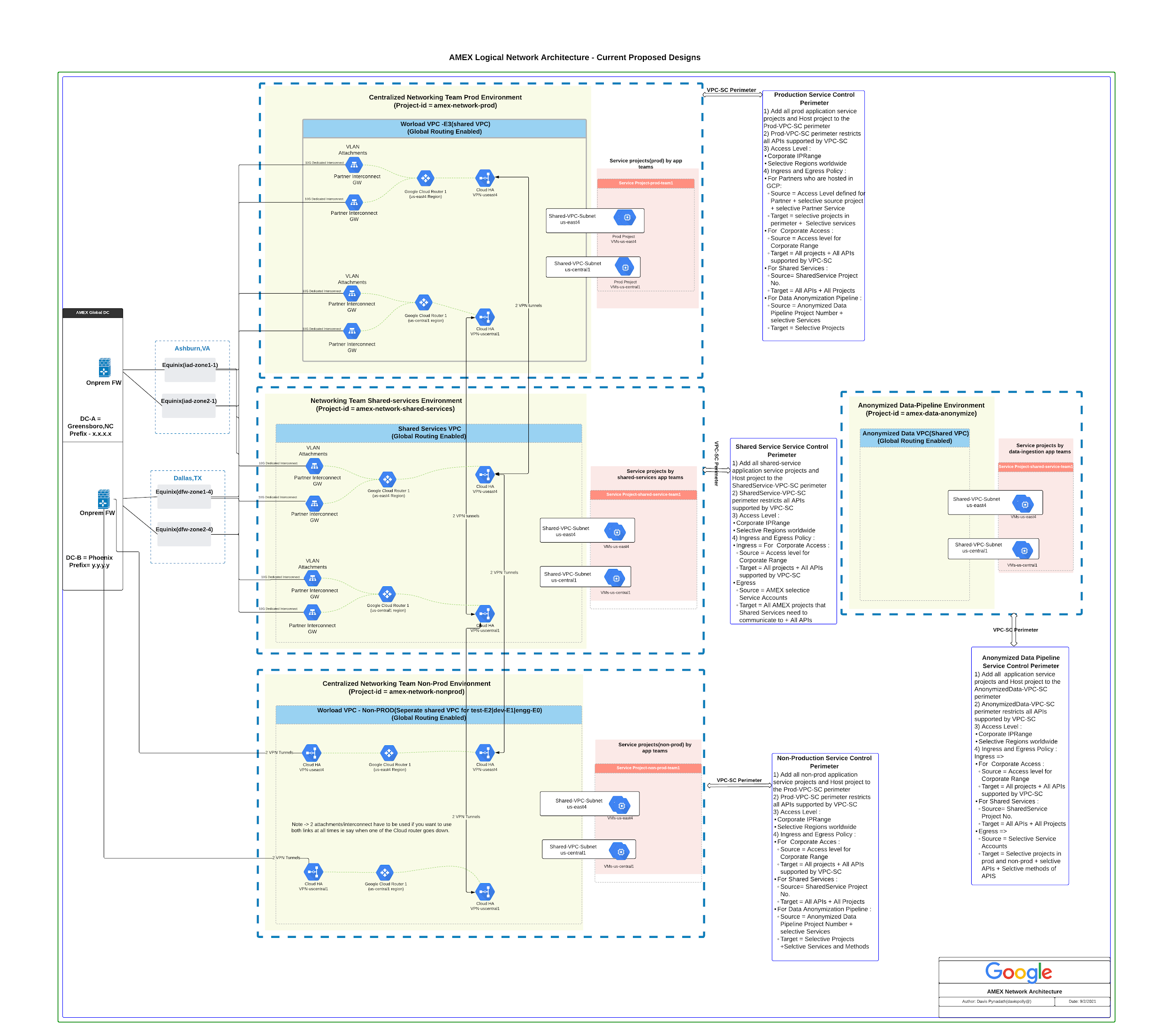
IAM Policies with conditions are widely used in other Public Cloud Platforms to isolate resource access within the same environment. Although supported on GCP, when using Shared VPCs, it is often operationally easier to separate resources into different service projects and have dedicated IAM controls around the project boundary than try to manage resource groups within the same project. Google VPCs and projects are not the same as Accounts and Subscriptions on other Public Cloud Platforms. So, when using IAM Policies within conditions, make sure you are not overloading your IAM boundary around the project.

# 

# 4. Networking

AMEX networking team provides the networking and security infrastructure for all AMEX applications. The layout for the GCP network design is outlined below :

**Current Phase(no direct Internet Ingress and Egress into GCP) :**

****

## 

## 4.1 Networking requirements

The network group within AMEX has a number of key requirements for the GCP network architecture design. The main requirements are as follows:

1. Centralized network team would be governing VPC routing, connectivity to DC and subnet allocation for app-teams. The NetSec would govern firewalling and communication requirements from application-team VMs(BU-1) to on-premises or other application-team VMs(BU-2).
2. Exposure of application team VMs to the internet is not allowed as part of the initial phase (current phase). When native internet ingress/egress directly into GCP is allowed(later phase), a FW appliance in GCP will be needed. This appliance will also perform internal WAF capabilities.
3. As part of the current phase, all internet egress traffic from GCP hosted applications has to go through DC and needs to be inspected by an Onprem FW appliance. Appliance also provides WAF capabilities. Internet ingress into GCP hosted applications is not allowed.
4. All DC traffic (on prem servers) to GCP hosted applications must be inspected by an Onprem FW appliance. This appliance should have WAF capabilities.
5. Inter-application traffic hosted in a distinct GCP environment ( prod and non-prod) needs to be inspected by the Onprem FW appliance with WAF Capabilities.
6. VPC Service controls will be leveraged by AMEX for the prevention of data-exfiltration. There will be a separate perimeter for prod, non-prod applications. There will also be a defined set of ingress/egress access rules for partner-services consuming prod/non-prod GCP applications.
7. Production Data-Sets need to be anonymized and copied over to non-prod Data-sets. This transaction must be enforced with the right security controls.
8. Inter VM communication should be encrypted by default and all traffic-paths must be controlled by the Net-Sec team.

Note : GCP hosted services that require direct internet exposure need to go through an exception process. **This is not part of the current phase deployment model in GCP and is part of a** [**future phase**](#_h5ge04r4hvws). When approving such a request in the future, the following conditions have to be met :

* Hosted services must be exposed using HTTPS LB and have a WAF such as Cloud Armor enabled with the right set of preconfigured rules.
* PAN-VM Series/Security-Appliance must be deployed in the architecture.
* Hosted services must be spun up ideally in a shared-VPC(**external Shared VPC**) that is different from the shared-VPC used for internal applications. This external Shared-VPC will be connected to the untrust/connectivity VPC using PAN-GCP/security appliance.
* Need for inspection of direct internet ingress and egress, DC ingress and egress is met through the PAN-GCP appliance for IPS/IDS purposes.
* Creation of GCP FW Rule for HTTPS LB range for ingress and egress with the target=backendserviceVM/service-account.

## 4.2 Proposed VPC Design

### 4.2.1 [Current Phase](#_6ze32o7hh6pb)

This section documents the network architecture that has been proposed to the AMEX Network team.

AMEX DCs are located at Greensboro, NC and Phoenix, Arizona. Based on DC location and latency to GCP regions, the recommended GCP region mapping would be :

| **DC** | **GCP Region** |
| --- | --- |
| Greensboro | us-east4 |
| Phoenix | us-central1 |

Based on the Network requirements stated [above](#_fu0oje162999), AMEX will leverage the **On-Prem FW Appliance** during the current/initial phase to :

* Inspect all traffic from DCs to GCP
* Inspect all traffic from Internet egress into GCP hosted applications
* Inspect all traffic across GCP prod and non-prod VPCs if any traffic-flows are ever allowed(not recommended)

It is important that this onprem FW appliance also provides WAF capabilities.

Onboard application into Shared VPC :

AMEX will leverage GCP Shared VPC for hosting workloads owned by different teams that would :

* Need to privately communicate (RFC1918) with each other
* Need high bandwidth access to google-apis today. In the future when Appliance/PAN-VM Series is deployed, traffic path would be the following :
  + Google apis direct through High-BW fiber
  + Workload API calls through PAN-VM Series/Appliance
* Have the requirements as outlined [above](#_fu0oje162999)

It is in the above circumstances that AMEX would enable teams to onboard into the shared-VPC. Onboarding the team into the shared VPC would work by :

1. Creating a **distinct subnet/region** for the particular team. Identification of the IP-block range is crucial so the workloads can scale.
2. Share the above defined subnet from the host-project to the application-team project (service-project)
3. Add the **service project to the** **VPC service control perimeter of the environment**
4. Create FW rules in DC to enable communication from the above subnet to onprem-clients/servers.
5. Create GCP ingress and egress FW rules :
   1. To enable communication between the subnets of the distinct workload-teams
   2. To enable communication from DC.

AMEX will create the following VPCs within GCP as part of the [current phase](https://docs.google.com/document/d/1V-7c57ZzmETrrH4TnGdEl-0FvW-Mu48A_u3ieQuY7iQ/edit#heading=h.w5d5e6l98vj2) :

* **Amex-shared-services VPC :** This is AMEX’s shared services VPC and will host all steams. Network team needs to build a workflow to accommodate this onboarding through the existing CI/CD pipeline. Some of the key requirements that the application teams need to fill for onboarding are -
  + - Connectivity to DC requirement
    - Connectivity to alternate application-team services requirement
    - Leveraging GKE ? If so, service-cidr, pod-cidr and node-cidr are required. If not, primary subnet CIDR is required based on the number of application-instances at peak traffic volumes.
    - GKE Service accounts and other service-accounts that need to be given <x-IAM-permissions> in the shared-vpc host project.
    - Determine the set of googleapis required by the App team.
  + Interconnect attachments from DCs will land on this VPC for Onprem/DC Connectivity
  + Google PSO Recommends exploring Private Service Connect(PSC) as an option to enable RFC1918 communication between shared-services and environmental VPCs (if feasible). Please refer [here](https://cloud.google.com/vpc/docs/configure-private-service-connect-services) for implementation details for leveraging PSC for exposing a service to an alternate VPC. If PSC cannot be used, Cloud HA VPN GW will be created in us-east4 and us-central1 to facilitate RFC1918 communication between amex-shared-services VPC and other environmental VPCs(environmental VPCs are described below).

Note -> VPC-peering is not recommended as an interconnectivity method because of the [aggregated network quotas that VPC-peering creates](https://cloud.google.com/vpc/docs/quota#vpc-peering-effective-limit) creating scale concerns.

* + Service projects of amex-network-shared-services Shared-VPC should be added to the **Shared-services VPC Service control perimeter.**
* **Amex-prod-shared VPC** : This is AMEX’s production GCP VPC and will host prod applications. This is **Amex’s E3 environment.**
  + This VPC will be created in project-id = **amex-network-prod**
  + This is a shared VPC. Subnet-allocations and application team onboarding would function similarly to the Amex-shared-services VPC
  + Interconnect attachments from DCs will land on this VPC
  + If Private Service Connect cannot be leveraged, Cloud HA VPN GW will be used in us-east4 and us-central1 to enable RFC1918 communication between amex-shared-services VPC and Amex-prod-shared VPC
  + Service projects of amex-prod-shared VPC should be added to the **Prod VPC Service control perimeter.**
* **Amex-dev-shared VPC** : This is AMEX’s Development GCP VPC and will host dev applications.This is **Amex’s E1 environment.**
  + This VPC will be created in project-id = **amex-network-nonprod**
  + This is a shared VPC. Subnet-allocations and application team onboarding would function similarly to the Amex-shared-services VPC
  + Cloud HA VPN GWs will be used to interconnect this environment to Onprem using IPSec tunnels
  + If Private Service Connect cannot be leveraged, Cloud HA VPN GW will be used in us-east4 and us-central1 to enable RFC1918 communication between amex-shared-services VPC and Amex-dev-shared VPC
  + Service projects of amex-dev-shared VPC should be added to the **Non-Prod VPC Service control perimeter.**
* **Amex-test-shared VPC** (not shown in above diagram, to maintain simplicity) : This is AMEX’s Test/Integration GCP VPC. This is **Amex’s E2 environment.**
  + This VPC will be created in project-id = **amex-network-nonprod**
  + This is a shared VPC. Subnet-allocations and application team onboarding would function similar to the Amex-shared-services VPC
  + Cloud HA VPN GWs will be used to interconnect this environment to Onprem using IPSec tunnels
  + If Private Service Connect cannot be leveraged, Cloud HA VPN GW will be used in us-east4 and us-central1 to enable RFC1918 communication between amex-shared-services VPC and Amex-test-shared VPC
  + Service projects of amex-test-shared VPC should be added to the **Non-Prod VPC Service control perimeter.**
* **Amex-engg-shared VPC** (not shown in above diagram, to maintain simplicity) : This is AMEX’s Engg/Lab GCP VPC. This is **Amex’s E0 environment.**
  + This VPC will be created in project-id = **amex-network-nonprod**
  + This is a shared VPC. Subnet-allocations and application team onboarding would function similar to the Amex-shared-services VPC
  + Cloud HA VPN GWs will be used to interconnect this environment to Onprem using IPSec tunnels
  + If Private Service Connect cannot be leveraged, Cloud HA VPN GW will be used in us-east4 and us-central1 to enable RFC1918 communication between amex-shared-services VPC and Amex-engg-shared VPC
  + Service projects of amex-engg-shared VPC should be added to the **Non-Prod VPC Service control perimeter.**
* **Amex-data-anonymize-shared VPC :** This VPC will host data-ingestion pipelines(GKE/dataproc clusters) that would anonymize production datasets. The anonymized data-sets can then be written into non-prod environments for consumption
  + This VPC will be created in project-id = **amex-data-anonymize**
  + This is a shared VPC. Subnet-allocations and application team onboarding would function similar to the Amex-shared-services VPC
* Service projects of Amex-data-anonymize-shared VPC should be added to the **Anonymous Data VPC Service control perimeter**.hared services (such as Active directory, logging and monitoring servers, <>)
  + This VPC will be created in project-id = **amex-network-shared-services**
  + This is a shared VPC
  + Distinct subnets from this VPC would be shared to shared-services application

### 4.2.2 [Future Phase](#_jvrl2ab7oc4v)

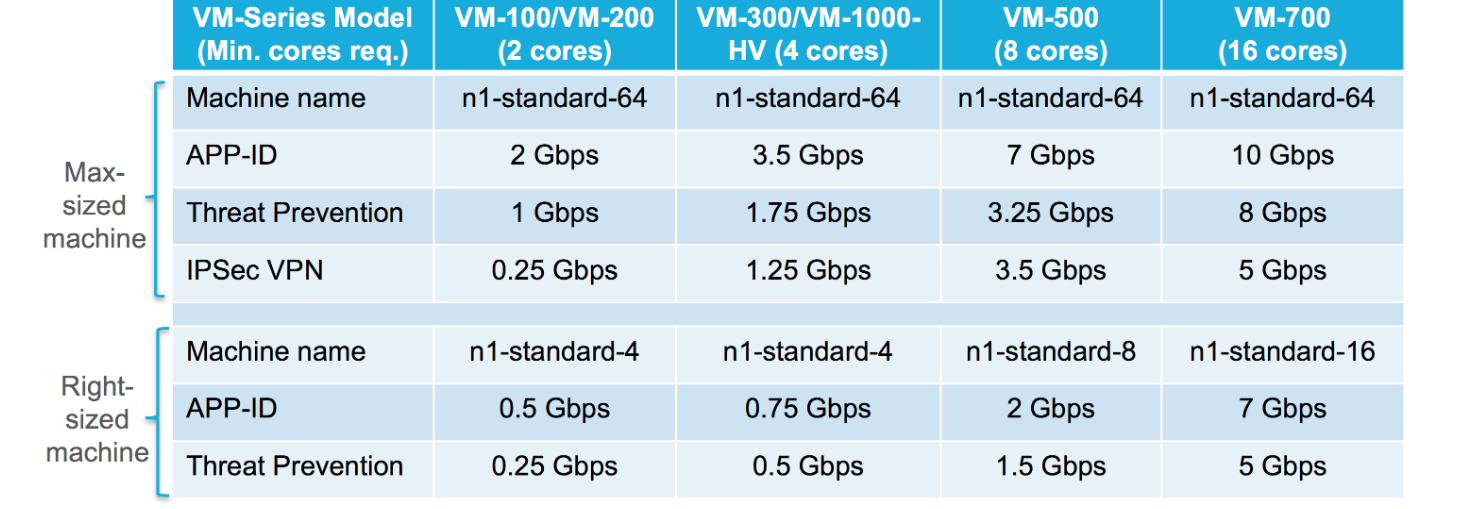
As part of this phase, direct Internet ingress and egress will be enabled into GCP hosted applications. This design is an **extension of the current-phase design.**

**Note** : ​​ Below design is an initial proposal from Google and should be treated as such – the final design is a subject of further discussions prior of providing a final solution.

Based on the Network requirements stated [above](#_fu0oje162999), AMEX will leverage the **PAN-VM Appliance** within GCP to :

* Inspect all traffic from Internet ingress/egress into GCP hosted applications
* To perform IPS
* Inspect all traffic across GCP prod and non-prod VPCs natively in GCP instead of onprem-FW
* WAF solution should be utilized for all web traffic. This needs to be further discussed.

Based on the PAN-appliance chosen, throughput capabilities are follows:



Note : A VM Appliance in GCP can only have a max of 8 NIC(network Interfaces) and hence can interconnect only upto 8 VPCs.

In addition to the VPCs outlined in the [current-phase](#_auzkpwdyyaj2), AMEX will create the following VPCs :

* **Amex-connectivity-prod VPC** : This is the untrust(DMZ)/connectivity VPC and will act as a transit VPC across AMEX prod-shared-VPC and DC.
  + This VPC will be created in project-id = **amex-network-prod**
  + This is a normal VPC and not a shared-VPC.
  + The untrust interface of the prod-PAN-VMSeries appliance (NIC-0) will land on this VPC. The trust-interface of prod-PAN-VMSeries appliance(non-NIC0) will land on the prod-shared-VPC
  + Create a route : destination=amex-prod-supernet, nextHop = ILB-PAN-Appliance. PAN-appliance will have a policy to send the received traffic to the trust-interface.
  + Interconnect attachments from DCs will land on this VPC. At this point, AMEX could decide to deprovision the interconnect attachments that land directly in the prod-shared-VPC and leverage the appliance as a transit ie DC -> amex-connectivity-vpc -> PAN-Appliance -> prod-shared-VPC
* **Amex-connectivity-nonprod VPC** : This is the untrust(DMZ)/connectivity VPC and will act as a transit VPC across AMEX’s nonprod shared-VPC(arena, dev, test, engg) and DC.
  + This VPC will be created in project-id = amex-network-nonprod
  + This is not a shared VPC.
  + The untrust interface of the nonprod-PAN-VMSeries appliance (NIC-0) will land on this VPC. The trust-interface of nonprod-PAN-VMSeries appliance(non-NIC0) will land on the diff environmental non-prod VPCs(dev, test, arena, engg) thereby using up 4 NICs of the appliance.
  + Create a route : destination=amex-nonprod-supernet(4 supernets for the 4 nonprod VPCs), nextHop = ILB-PAN-Appliance. PAN policy must exist that will route to the correct trust-interface of PAN.
  + Interconnect attachments(IC) from DCs will land on this VPC. Amex could decide to deprovision the IC attachments that land directly on the non-prod shared-VPCs and use Amex-connectivity VPC as transit to DC.
* **Amex-management VPC** : This VPC is used for management of the PAN-VMSeries Appliance. This will exist both in project-id=amex-network-nonprod and amex-network-prod.
* **Amex-heartbeat VPC** : This VPC is used for exchanging the heartbeat messages for High-availability requirements of the PAN-VMSeries appliance. This will exist both in project-id=amex-network-nonprod and amex-network-prod.

Communication from **amex-shared-services VPC** tothe environmental VPCs(amex-**prod**-shared, amex-**dev**-shared, amex-**test**-shared, amex-**engg**-shared, amex-**data-anonymize**-shared) can now be enabled by :

* VPC-Peering **Amex-connectivity-nonprod** VPCand **Amex-connectivity-prod** VPCto **amex-shared-services** VPC**.** VPC Peering willnotcause a quota aggregation issue(scale concerns) since there are very minimal resources in the 3 peered VPCs.

Routing rules to enable shared-Services consumption across environments :

1. When the networks are VPC-peered, the shared-services (active directory..) subnet will automatically be exchanged from Amex-shared-services VPC to Amex-connectivity-nonprod and Amex-connectivity-prod VPC.
2. In the non-prod VPCs (dev, test, engg and arena), you need to create a static route : destination=shared-services-subnet, nextHop=ILB-PAN-VMSeries
3. In the prod VPCs, you need to create a static route : destination=shared-services-subnet, nextHop=ILB-PAN-VMSeries

Once the above connectivity is verified to shared-services, the Cloud VPN tunnels can be de-provisioned.

**Note** : Connectivity between amex-shared-services VPC and amex-data-anonymize-shared VPC will continue to use Cloud VPN Gateways.

## 4.3 Connection to Google

Due to AMEX’s long term strategy of hybrid connectivity between on-premises and GCP, it is recommended that AMEX connect to GCP via [Dedicated Interconnect](https://cloud.google.com/interconnect/docs/concepts/dedicated-overview). AMEX currently has two data center locations in scope for connectivity: 1) Greensboro, NC 2) Phoenix, AZ.

Google has multiple [locations](https://cloud.google.com/interconnect/docs/concepts/colocation-facilities#locations-table) to connect on-premises with GCP. AMEX is currently co-located with Google at :

* **Equinix facility in Dallas,Texas** and connectivity from AMEX Phoenix DC will be enabled through this location.
* **Equinix facility in Ashburn, Virginia** and connectivity from AMEX Greensboro DC will be enabled through this location.

In order to support [99.99% availability](https://cloud.google.com/interconnect/docs/tutorials/dedicated-creating-9999-availability) AMEX will need to select two Interconnect zones in two locations(metros). Please refer to the [network design above](https://docs.google.com/document/d/1V-7c57ZzmETrrH4TnGdEl-0FvW-Mu48A_u3ieQuY7iQ/edit#heading=h.ivetd73wbnfc) for an overview of this connectivity with the recommended VPC setup.

Amex predicts the following throughput requirements to GCP :

* Total Ingress into GCP : 10-20 TB/day
* Total Egress from GCP : 2-4 TB/day.

Also, prior to initial application-VM setup, AMEX envisions transfer of few PB data into GCP. This will be done using a transfer appliance instead of using the network.

Outlined below is transfer-time based on data and Interconnect-capacity to GCP.



Based on AMEX’s data volume transfer requirements, Google recommends provisioning at least 2\*10G in Equinix Ashburn facility(2 for redundancy and Ashburn is [low-latency location](https://cloud.google.com/network-connectivity/docs/interconnect/concepts/choosing-colocation-facilities-low-latency?hl=en#locations-table)) and 2\*10G in Equinix Dallas,TX.

The Interconnect objects should be provisioned in a new project = amex-network-interconnect with no other resources within it. This project should be ideally under the shared-services folder. This aids with :

* determining bill for interconnect usage and enables easy charge-back to different Business units.
* Easy IAM implementation for Amex eng who are allowed to provision and land the Interconnect attachments on various GCP projects in AMEX org.

### 4.3.1 Internal

During design discussions, AMEX expressed the desire to route GCP API calls from on-premises to GCP over the Dedicated Interconnect connection. [Private Google Access from on-premises](https://cloud.google.com/vpc/docs/configure-private-google-access-hybrid), would allow on-premises users to call [most Google APIs and services](https://cloud.google.com/vpc/docs/private-google-access#pga-supported) through a private endpoint (199.36.153.8/30).

However, since AMEX would use [VPC-Service Controls](https://cloud.google.com/vpc-service-controls) for maintaining a strong security-posture, it requires leveraging **restricted.googleapis.com (199.36.153.4/30)** for on-prem to GCP API calls. Note - This private VIP (199.36.153.4/30) only enables calls to APIs that are supported by VPC-Service controls.

[Here](https://cloud.google.com/vpc-service-controls/docs/supported-products) is a list of detailed APIs and Services supported by VPC-Service control. **For onprem-client calls that are made to APIs outside this list, will be blocked.**

To make use of this capability for on-premises, AMEX will need to:

1. Ensure onprem-clients/applications make \*.googleapis.com calls to the restricted.googleapis.com. This can be done executing the steps [here](https://cloud.google.com/vpc/docs/configure-private-service-connect-apis)(i.e. Update to cloud SDK or application-code). This is the only agreed approach from Amex, since manipulation of googleapis is not possible in on-prem DNS NSs.
2. Advertise the prefix-199.36.153.4/30 from the **amex-shared-services** VPC to on prem over the cloud-router BGP Session attached to the dedicated interconnect-attachment.
3. Within GCP
   1. DNS Private zone to resolve (CNAME) \*.googleapis.com to restricted.googleapis.com. This should be done in the DNS-Hub(which is the shared-services VPC project)
   2. Add the following A records for restricted.googleapis.com in GCP Private zone
      1. 199.36.153.4
      2. 199.36.153.5
      3. 199.36.153.6
      4. 199.36.153.7
   3. Configure the route for 199.36.153.4/30 to have a next hop of “Default Internet gateway”

Note : The IPblock 199.36.153.4/30 is not internet routable and is purely a private endpoint available only within the GCP backbone. Also, some Google APIs and services are provided using additional domain names, including \*.gcr.io, \*.gstatic.com, \*.pkg.dev, and pki.goog. For these domains, a private zone having a CNAME would need to be created and A records described above (config details [here](https://cloud.google.com/vpc-service-controls/docs/set-up-gke)).

For connections initiated by a serverless GCP service which are not part of a VPC (e.g. Cloud Functions, App Engine Std, Cloud Run) a private IP-range must be assigned for it to communicate with VPC-resources having an Internal-IP (example - VMs/GKE nodes having only an internal IP). Configuring this private IP range for GCP Serverless services is enabled through [Serverless VPC Access Connector](https://cloud.google.com/vpc/docs/configure-serverless-vpc-access#connectors). In order for the VPC Serverless service to communicate privately with on prem servers, then the [private-ip range of the serverless VPC access connector connection](https://cloud.google.com/vpc/docs/configure-serverless-vpc-access#configuring) must also be exchanged over the CloudRouter BGP session of the interconnect. Per the scoping discussion, AMEX does not plan on using serverless GCP Products outlined, so there would be no need for Serverless VPC Access connector configuration.

**Routes**

Global dynamic routing should be enabled so that routes from both us-east4 and us-central1 are learned by on-premises routers. Priority for these routes is managed automatically where:

* Priority for region with Cloud Router is set up with value of base priority
* Other regions are set with base priority + region-to-region cost

Since internet traffic is going to be routed through on-premises to allow for advanced egress policies, the default internet gateway should be deleted or modified. On-premises routers should advertise routes to 0.0.0.0/0. The following routes will be created in GCP to support access to Google APIs through Private Google Access (ie without having to go to on prem):

| **Description** | **Destination** | Nexthop |
| --- | --- | --- |
| Restricted VIP | 199.36.153.4/30 | *default internet gateway* |
| Private VIP | 199.36.153.8/30 | *default internet gateway* |

Note : The VPC subnet also needs to have the Private-google-access flag ON (config detail [here](https://cloud.google.com/network-connectivity/docs/interconnect/concepts/partner-overview#9999_availability_topology)). Also, the private VIP is present in the table above for clients making calls to GCP APIs not supported by VPC Service-Controls. If no GCP-VPC-resource should be allowed to make such calls, then remove the Private API route and Cloud DNS Private zone config for private-api.

### 

### 4.3.2 External

Exposing application services hosted in GCP to the internet is not permitted per AMEX security posture. Therefore all application-services are to be exposed either using a GCP Internal L4/L7 Load Balancer(L4/L7 ILBs). AMEX team needs to ensure the following :

* L4/L7 ILB Frontend VIP are exchanged using [custom route advertisements](https://cloud.google.com/network-connectivity/docs/router/how-to/advertising-custom-ip) on the Cloud Router BGP Session present in the Amex-connectivity VPC.
* There must be a route within Amex-connectivity VPC that would enable traffic to go to the environmental VPC (application subnets) through PAN-VM-Series ie destination=Application-Subne/Application-ILB, NextHop=ILB-PAN-VMSeries.
* L4/L7 ILB Frontend VIP is NAT’d accordingly to an advertised AMEX Public-IP address in DC.
* On prem and GCP Firewall rules are present that would enable the communication from client to server VMs.

## 

## 4.4 IP address space

In order to facilitate connectivity between AMEX data centers IP space and GCP IP space, a block of IP addresses will be reserved for use in GCP. To reduce management complexity, these addresses will not be provisioned on the existing data center network. The recommended IP address allocation is outlined in the table below. It’s Google’s understanding that AMEX is planning to leverage managed services as much as possible to reduce IT overhead and increase efficiency. While the proposed approach may seem aggressive, it is important that proper planning is taken into account when deploying workloads on certain Google Cloud services, primarily Google Kubernetes Engine (GKE) and [Dataproc Clusters](https://cloud.google.com/dataproc/docs/concepts/overview).

### 4.4.1 GKE IP Addresses

GKE requires significant IP addresses for its various components, namely Nodes, Pods and Services. When creating a VPC-native cluster (cluster that uses [alias IP ranges](https://cloud.google.com/vpc/docs/alias-ip)), a subnet in VPC network is specified. The GKE cluster uses three unique subnet IP ranges:

* **Primary IP range for all node IP addresses:**
  + **Example**: if AMEX plans to create a 900 node cluster, the primary IP address range must be at least a /22 which results in 1024 addresses. Of those, 1020 are usable since four IP addresses are reserved for GCP.
* **One Secondary IP range for all Pod IP addresses:**
  + **Example**: For a 900 node cluster supporting up to 110 Pods per node, 900 x 256 = 230, 400 IP addresses for Pods are needed. 256 addresses are used in the above calculation because Kubernetes by default allocates twice as many IP addresses as the number of pods for pod scaling.
* **Another Secondary IP range for all Service addresses**:
  + **Example**: For a cluster that runs up to 3000 Services, 3000 cluster IP addresses are needed. Therefore, a secondary range of size /20 or larger is required which results in 4096 IP addresses.

When allocating IP address ranges for GKE, it’s important to note that secondary IP address ranges cannot be expanded once they’re created and replacing the address range has the potential to put the GKE cluster in an unstable state.

### 4.4.2 Dataproc IP Addresses

Dataproc Clusters consist of master and worker VMs. The detailed networking config for dataproc is outlined [here](https://cloud.google.com/dataproc/docs/concepts/configuring-clusters/network).

### 4.4.3 Other IP Addresses

In addition to dataproc clusters and GKE, there are subnet allocation for VM based workloads and Serverless VPC connectors (enables access from Serverless products : Cloud Run, App Engine, Cloud Functions)

| Environment | Region | Purpose | CIDR |
| --- | --- | --- | --- |
| Prod | us-central1 | Dataproc Cluster |  |
| VM Based workload |  |
| GKE: Primary |  |
| GKE: Cluster |  |
| GKE: Service |
| us-east4 | Dataproc Cluster |  |
| VM Based workload |  |
| GKE: Primary |  |
| GKE: Cluster |  |
| GKE: Service |

| Environment | Region | Purpose | CIDR |
| --- | --- | --- | --- |
| Dev | Test | Engg | Arena | us-central1 | Dataproc Cluster |  |
| VM Based workload |  |
| GKE: Primary |  |
| GKE: Cluster |  |
| GKE: Service |
| us-east4 | Dataproc Cluster |  |
| VM Based workload |  |
| GKE: Primary |  |
| GKE: Cluster |  |
| GKE: Service |  |

| Environment | Region | Purpose | CIDR |
| --- | --- | --- | --- |
| Test/Integration | us-central1 | Dataproc Cluster |  |
| VM Based workload |  |
| GKE: Primary |  |
| GKE: Cluster |  |
| GKE: Service |
| us-east4 | Dataproc Cluster |  |
| VM Based workload |  |
| GKE: Primary |  |
| GKE: Cluster |  |
| GKE:Service |

## 4.5 Cloud DNS

Internal DNS should be handled by Google Cloud internal DNS which is a scalable, reliable, and managed authoritative Domain Name System (DNS) service running on the same infrastructure as Google.

With the current Network Architecture, [private zones](https://cloud.google.com/dns/zones/) will be created in **each region** within the VPC (prod, dev, test, engg, arena).

Enabling GCP Cloud DNS service provides several benefits for the AMEX GCP setup. Use of Cloud DNS will allow [logging](https://cloud.google.com/dns/docs/monitoring#private_dns_logging) of resolution requests.

In order to have consistency throughout the environment, a naming standard should be defined for an organization. For the AMEX GCP environment :

* Cloud DNS should use the prefix “gcp” and the “GCP region name” before the domain in the Cloud DNS Private zone defined. Example - [gcp-us-east4.aexp.com](http://gcp-us-east4.aexp.com/)
* Vanity names as application/service aliases will be leveraged for the individual user facing applications deployed in GCP.

Please note, the scope of this design is for managed private zones—external-facing DNS is out of scope.

It is recommended that AMEX configures DNS Peering to obtain name-resolution of private-zone files defined in each environment Shared VPCs (prod, dev, test, engg, arena) to the **amex-shared-services VPC Project.** The amex-shared-services VPC would function as the DNS Hub.

For GCP to AMEX on prem name resolution, GCP DNS forwards to the NS on prem (outbound forwarding) through creation of a forwarding zone. The forwarding zones will be defined on amex-shared-services VPC(DNS Hub). The forwarding zone defined will have the domain/sub-domain of Amex’s choice of DC with the exact Onprem DNS NameServers(NSs) within the DC. The request will originate from prefix=35.199.192.0/19, therefore the prefix needs to be advertised over the BGP session of the Cloud Interconnect Attachment on amex-shared-services VPC and appropriate on prem FW rules needs to be created to allow this traffic.

## 4.5 Cloud Load Balancing

Since the current phase involves no direct internet ingress/egress, only Google’s Internal Load Balancer(ILB) are in scope for the deployment. GCP has two kinds of ILBS :

* L4 ILB which is regional in nature. The backend servers have to be in the same region as where the ILB is created. The clients making a call to the ILB can be from onprem or any other GCP region
* L7 HTTP(S)ILB is also regional in nature and is an Envoy based implementation. The difference between L7 ILB and L4 ILB :
  + Routing decisions(which backend service should receive the traffic, there can be multiple) can be made based on URL-maps/path-rules.
  + As of this writing, the clients making the call to the L7 ILB has to be in the same GCP-region as the L7 ILB
  + SSL offload is possible

Amex has decided to use only L4 ILB in the current deployment. When an application team exposes their service using L4 ILB, the following steps have to taken :

1. Create a private zone in Cloud-DNS with appropriate A records of ILB-VIP so that GCP-clients resolve to the app using FQDN
2. Exchange the ILB-VIP over the appropriate BGP Session to DC/Onprem
3. Update GTM zone/records that enables resolution of Apps FQDN to ILB-VIP(A-record creation). This enables onprem systems to resolve to the appropriate regional-ILB and essentially enables traffic flow to the hosted-GCP-appservice

Exposing Internet facing GCP-Apps through Amex DC is currently out of scope.

## 4.7 Firewall rules

Note : For more details on GCP Security, refer to the section [here](#_105upw9geifp).

Firewall rules allow or deny access to or from any server running on Google Compute Engine. AMEX is responsible for ensuring that firewall rules are configured to grant network access only to the systems, protocols, and ports that are required. Enabled GCP firewall rules are always enforced, protecting your instances regardless of their configuration and operating system, even if they have not started up.

Every VPC network functions as a distributed cloud firewall. While GCP firewall rules are defined at the network level, connections are allowed or denied on a per-instance basis. You can think of the GCP firewall rules as existing not only between your instances and other networks, but between individual instances within the same network.

GCP firewall rules are matched via IP address, arbitrary tags, or via service accounts. While GCP firewall rules are applied at the instance level, their configuration is associated with a VPC network. You cannot share GCP firewall rules among VPC networks, including networks connected by VPC peering or VPN tunnels. Google PSO recommends building GCP firewall rules that target (and source) service accounts instead of target tags and source tags. The reasoning is on account of:

* **A network tag is an arbitrary attribute.** One or more network tags can be associated with an instance by any IAM member who has permission to edit it. IAM members with the Compute Engine Instance Admin role to a project have this permission. IAM members who can edit an instance can change its network tags, which could change the set of applicable GCP firewall rules for that instance. If you are using network tags in your GCP firewall rules and want to make sure only the networking/security team can specify these tags, do not use the Compute Engine Instance Admin IAM role but instead create and use a custom role with the same permissions except the instances.setTags permission.
* **A service account represents an identity associated with an instance.** Only one service account can be associated with an instance. You control access to the service account by controlling the grant of the Service Account User role for other IAM members. For an IAM member to start an instance using a service account, that member must have the Service Account User role to at least that service account as well as appropriate permissions to create instances (for example, having the Compute Engine Instance Admin role to the project).

To satisfy AMEX's most restrictive security requirements, Google is recommending that ALL firewall rules be created with 2 explicit “Deny all” policies for Ingress and Egress. Any traffic flows should then be enabled on an exception basis. The main reason for this is to ensure that Firewall logs are triggered each and every time a non approved traffic flow is observed. Google default deny policies do not trigger Firewall logs. This can lead to loss of visibility to either misconfigured application traffic flows or possible unauthorized network scans.

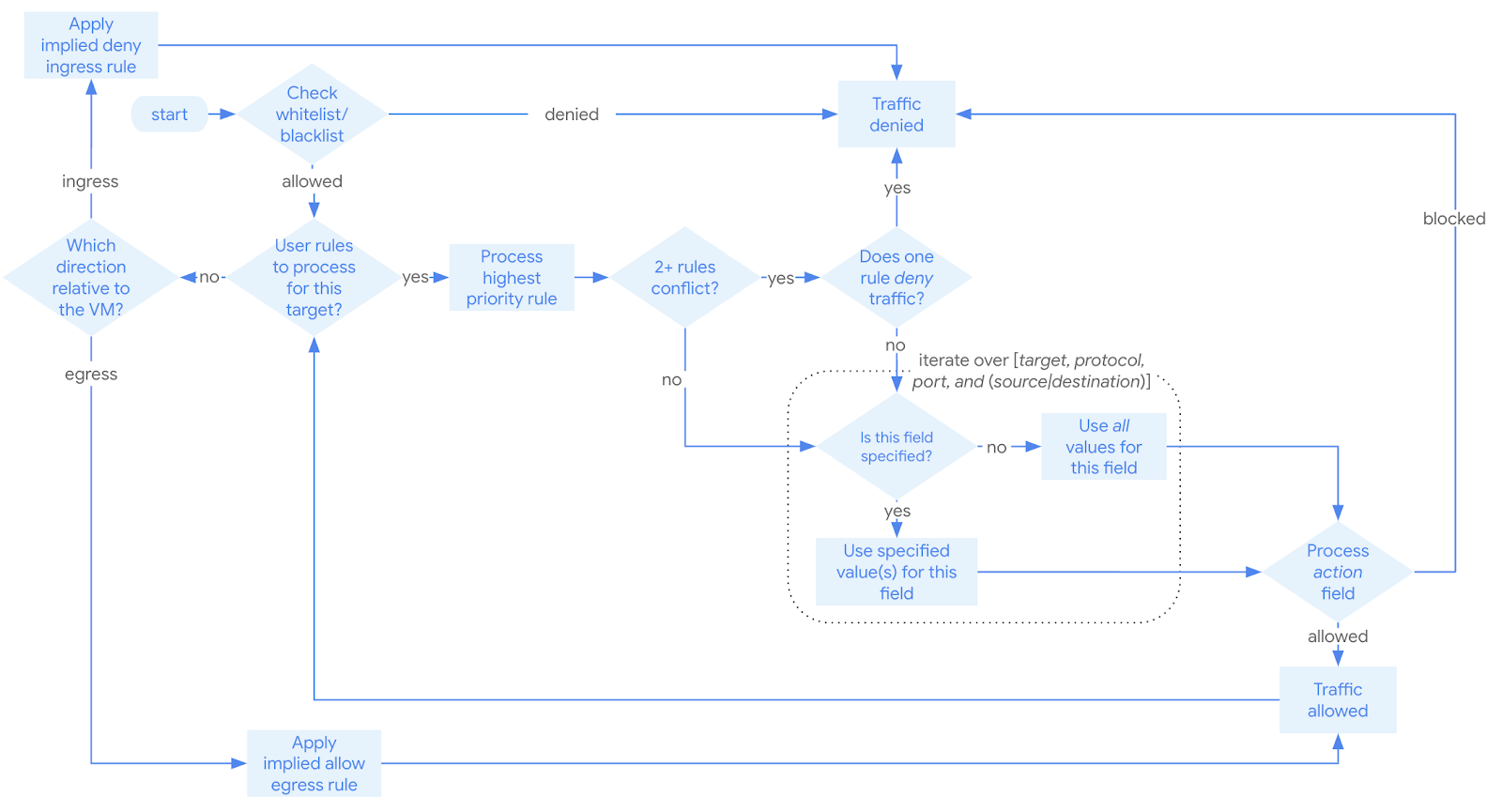
### 4.7.1 Evaluation Logic

Google Cloud always allows communication between a VM instance and its corresponding metadata server at 169.254.169.254. Google Cloud always blocks protocols other than TCP, UDP, ICMP, and IPIP, and egress traffic on TCP port 25. No firewall rules can be used to allow this traffic. Firewall rules only support IPv4 traffic.

Note : IPV6 for GCE instances is only supported in selective GCP regions. It is not supported in GCP Regions Amex leverages. When it does get supported, AMEX will have the capability to create v6 Firewall rules. Also, [by default v6 traffic Ingress](https://cloud.google.com/compute/docs/ip-addresses/configure-ipv6-address) into GCP is blocked.

Priority can be set on firewall rules. The firewall rule priority is an integer from 0 to 65535, inclusive. Lower integers indicate higher priorities. The relative priority of a firewall rule determines whether it is applicable when evaluated against others.

The diagram below illustrates how sets of Firewall Rules are processed. The diagram does not include implied rules or default rules for a default network, as the processing logic only cares about processing the rules and is not concerned with how the rules were inserted into the stack of rules to process.



The network topology defined for AMEX GCP is the Shared VPC per environment. Options for an initial set of security focused firewall rules AMEX may configure are:

**Production**

| **Action** | **Direction** | **Port** | **Target** | **Filter** | **Logging** | **Reason** |
| --- | --- | --- | --- | --- | --- | --- |
| Deny | Ingress | ALL | ALL | 0.0.0.0/0 | On | Recreate default behavior to enable logging |
| Deny | Egress | ALL | ALL | 0.0.0.0/0 | On | Block all egress and enable logging |
| Allow | Egress | ALL | ALL | Patching, monitoring, heartbeat, private APIs VIP, approved on prem CIDR, VPC Subnet CIDRs | On | Allow VMs within VPC to egress to requisite services |

Application teams could request specific Service Account firewall rules for production which allows appropriate ingress and egress. For example, if an application makes use of the Google Cloud Global Internal Load Balancer, then [firewall rules](https://cloud.google.com/load-balancing/docs/https#firewall_rules) must be created to allow connection to/from the Load Balancer.

**Non-Production**

| **Action** | **Direction** | **Port** | **Target** | **Filter** | **Logging** | **Reason** |
| --- | --- | --- | --- | --- | --- | --- |
| Deny | Ingress | ALL | ALL | 0.0.0.0/0 | On | Recreate default behavior to enable logging |
| Deny | Egress | ALL | ALL | 0.0.0.0/0 | On | Block all egress and enable logging |
| Allow | Egress | ALL | ALL | Patching, monitoring, heartbeat, private APIs VIP, approved on prem CIDR, VPC Subnet CIDRs | On | Allow VMs within VPC to egress to requisite services |
| Allow | Ingress | 22, 3389 | ALL | Amex Corporate Range | On | Allow SSH and RDP |
| Allow | Ingress | 80, 443, | ALL | Amex Corporate Range | On | Allow ingress for HTTP/S from AMEX ranges |

Application teams could request specific Service Account firewall rules for non-production which allows appropriate ingress and egress. Google’s default security posture has a default deny firewall rule to disallow all incoming traffic. Our recommendation is that this be overridden with an explicit deny rule to enable the triggering of FW logs when a deny rule is enforced. Failure to do this will result in the deny rules still being enforced, but not firewall logging of the event occurring.

All remote access connectivity will terminate directly into Amex’s on-premises sites. No remote access connectivity will terminate directly into the GCP environment. Additionally, Amex makes use of bastion hosts for production ssh and RDP access, to prevent lateral movement attacks from on premises to GCP. It is recommended that to simplify security audits and policy, Amex utilizes this same architectural framework in GCP.

Google PSO also recommends turning on VPC flow logs (on GCP subnets) and FW logs (on each FW rule created) to be turned on. This is so that any deny/allow rules are logged and any malicious actors/speakers traffic flows could be determined through log analysis.

#### *Firewall Layers*

Along with options for source, destination and ports, firewall rules also have the ability to define a priority. Rules are evaluated from **lowest numerical** (1) to **highest numerical** (65535) value and stop processing with the first direct match to the rule. Rules with a lower numerical value are considered a higher priority.

This affords the ability to layer **allow** and **deny** rules to reduce the number of rules and reduce the impact of an improperly tagged instance from elevating its access to other VMs.

The table below outlines the layers that have been **proposed as a possible framework:**

| 0-9,999 | 10,000 | 10,001-29,999 | 30,000 | 30,001-65,530 |
| --- | --- | --- | --- | --- |
| Global  InterProject InterRegional  (Cross Projects, Cross Regions)  Source Ip/Subnet based | Project lockdown rules | IntraProject InterRegional  (Single Project, Cross Regions)  Source Tag Based | Regional Lockdown Rules | IntraProject IntraRegional  (Single Project, Single Region)  Rules  Source Tag Based |

##### **Priority 0-9,999**

Reserved for Global rules:

1. Rules that allow for traffic from the Public Internet into GCP(exception)
2. Rules that allow for communication across different projects and different regions

Rules within this priority range should be “ALLOW” rules and should use an IP/Subnet for its source. A Service account can be used as a source here as well, but network tags should not be used as a source within this priority range.

**Note** : Cloud NAT will not be used today in AMEX’s design since internet communication from VM/Dataproc/GKE workloads is not allowed. All internet traffic must go through AMEX DC through the GCP PAN-VMSeries.

##### **Priority 10,000**

Reserved for Project Lockdown Rules

Rules with a priority of 10,000 should be used to lock down access to a project. This allows for the use of source tags in rules with a priority of 10,001 or greater.

An application/solution should have its own dedicated subnet space wherever possible. Applications/solutions that share subnets with other applications/solutions should use service accounts for firewall rules.

The source range for the lockdown rules should be all CIDR ranges except for the subnets that the project/solution is deployed in.

The Target should be one of the following:

1. A project/solution specific tag
2. All of the Service Accounts used within the project/solution
3. All of the Tags used within the project/solution

##### **Priority 10,001-29,999**

Reserved for IntraProject InterRegional Rules:

1. Rules that allow for communication within a specific project across regions

Rules within this priority range should be “ALLOW” rules and should use Service Accounts or Tags for its source. Service accounts should be preferred and Tags should only be used when absolutely necessary.

##### **Priority 30,000**

Reserved for Regional Lockdown Rules

Rules with a priority of 30,000 should be used to lock down access between regions. This allows for the ability to consolidate multiple rules that would normally be needed to allow for traffic only within a region into a single rule.

The source range for the Regional Lockdown rules should be all CIDR ranges except for the range for a particular region.

The Target should be one of the following:

1. A regional specific tag
2. All of the Service Accounts used within a specific region
3. All of the Tags used within a specific region

##### **Priority 30,001-65,529**

Reserved for IntraProject IntraRegional Rules:

1. Rules that allow for communication within a specific project and within a specific region

Rules within this priority range should be “ALLOW” rules and should use Service Accounts or Tags for its source. Service accounts should be preferred and Tags should only be used when absolutely necessary.

##### **Priority 65,530**

Reserved for default rules

Rules with a priority of 65,530 should be considered baseline rules that apply to everything but can be overridden if required. This allows for the ability to create a rule that applies to all services that needs to be explicitly overridden. These should be used to override Google default Ingress and Egress rules to deny all traffic as a default.

## 4.8 VPC Service Controls

VPC Service Controls improves the ability to mitigate the risk of data exfiltration from Google Cloud services such as Cloud Storage and BigQuery. With VPC Service Controls, perimeters are created that protect the resources and data of services that are explicitly specified.

VPC Service Controls provides an additional layer of defense for Google Cloud services that is independent of Identity and Access Management (IAM). While IAM enables granular identity-based access control, VPC Service Controls enables broader context-based perimeter security, including controlling data egress across the perimeter. AMEX will be using both VPC Service Controls and IAM for defense in depth.

### 4.8.1 Service Perimeter

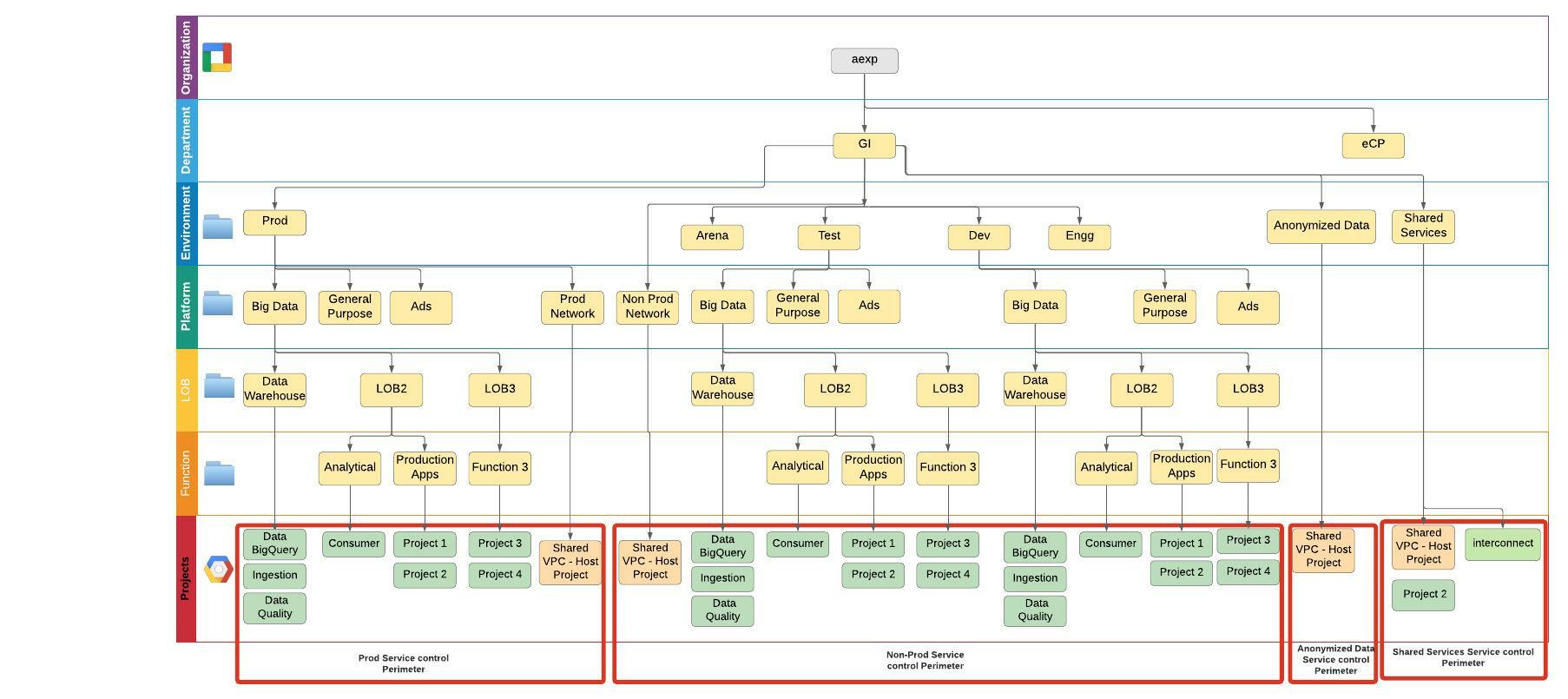
#### *4.8.1.1 Overview*

VPC Service Controls provide a mechanism to secure API-based services, like Cloud Storage and BigQuery by using a strict perimeter. [Here](https://cloud.google.com/vpc-service-controls/docs/restricted-vip-services) is a list of APIs supported by VPC-SC and [these](https://cloud.google.com/vpc-service-controls/docs/supported-products#unsupported_services) are the services that are not supported.

By default, a perimeter disallows all access to a service it is configured to protect from outside the perimeter, even if the user has permission to use that service.

The AMEX foundation will contain 4 primary perimeters under the following folders :

* Production folder
* Non Production Folder
* Anonymized Data Folder
* Shared Services Folder



#### *4.8.1.2 Prod Perimeter - Configuration*

The AMEX VPC Service Control perimeter will have the following [configuration](https://cloud.google.com/vpc-service-controls/docs/create-service-perimeters#creating_a_service_perimeter):

* **Name**:
  + prod\_perimeter
* **Title**:
  + prod\_perimeter
* **Restricted Services**:
  + All supported APIs which may be restricted will be restricted.
* [**VPC Accessible Services**](https://cloud.google.com/vpc-service-controls/docs/vpc-accessible-services)**:**
  + RESTRICTED-SERVICES
* **Projects**:
  + All projects under the <Prod-name> folder will be a part of the perimeter.
* **Ingress/Egress Rules**:
  + Please [see section 4.7.3](#_9kzxpba4ry02) for a list of ingress and egress rules applied to this perimeter
* **Access Levels:**
  + No access levels will be applied to this perimeter directly at this time. Access levels will be used as part of ingress/egress rules.
  + Please [see section 4.7.2](#_hhcitodqrqw8) for information on Access Levels

#### *4.8.1.3 Non Prod Perimeter - Configuration*

The AMEX VPC Service Control perimeter will have the following [configuration](https://cloud.google.com/vpc-service-controls/docs/create-service-perimeters#creating_a_service_perimeter):

* **Name**:
  + nonprod\_perimeter
* **Title**:
  + nonprod\_perimeter
* **Restricted Services**:
  + All supported APIs which may be restricted will be restricted.
* [**VPC Accessible Services**](https://cloud.google.com/vpc-service-controls/docs/vpc-accessible-services)**:**
  + RESTRICTED-SERVICES
* **Projects**:
  + All projects under the “Dev, Test. Engg, Arena” folders will be a part of the perimeter.
* **Ingress/Egress Rules**:
  + Please [see section 4.7.3](#_9kzxpba4ry02) for a list of ingress and egress rules applied to this perimeter
* **Access Levels:**
  + No access levels will be applied to this perimeter directly at this time. Access levels will be used as part of ingress/egress rules.
  + Please [see section 4.7.2](#_hhcitodqrqw8) for information on Access Levels

#### *4.8.1.4 Shared Services Perimeter - Configuration*

The AMEX VPC Service Control perimeter will have the following [configuration](https://cloud.google.com/vpc-service-controls/docs/create-service-perimeters#creating_a_service_perimeter):

* **Name**:
  + sharedservices\_perimeter
* **Title**:
  + sharedservices\_perimeter
* **Restricted Services**:
  + All supported APIs which may be restricted will be restricted.
* [**VPC Accessible Services**](https://cloud.google.com/vpc-service-controls/docs/vpc-accessible-services)**:**
  + RESTRICTED-SERVICES
* **Projects**:
  + All projects under the “Shared Services” folders will be a part of the perimeter.
* **Ingress/Egress Rules**:
  + Please [see section 4.7.3](#_9kzxpba4ry02) for a list of ingress and egress rules applied to this perimeter
* **Access Levels:**
  + No access levels will be applied to this perimeter directly at this time. Access levels will be used as part of ingress/egress rules.
  + Please [see section 4.7.2](#_hhcitodqrqw8) for information on Access Levels

#### *4.8.1.5 Anonymized Data Perimeter - Configuration*

The AMEX VPC Service Control perimeter will have the following [configuration](https://cloud.google.com/vpc-service-controls/docs/create-service-perimeters#creating_a_service_perimeter):

* **Name**:
  + anonymized\_data\_perimeter
* **Title**:
  + anonymized\_data\_perimeter
* **Restricted Services**:
  + All supported APIs which may be restricted will be restricted.
* [**VPC Accessible Services**](https://cloud.google.com/vpc-service-controls/docs/vpc-accessible-services)**:**
  + RESTRICTED-SERVICES
* **Projects**:
  + All projects under the “Anonymized Data” folders will be a part of the perimeter.
* **Ingress/Egress Rules**:
  + Please [see section 4.7.3](#_9kzxpba4ry02) for a list of ingress and egress rules applied to this perimeter
* **Access Levels:**
  + No access levels will be applied to this perimeter directly at this time. Access levels will be used as part of ingress/egress rules.
  + Please [see section 4.7.2](#_hhcitodqrqw8) for information on Access Levels

### 4.8.2 Access Levels

Access is granted into a service perimeter through one of two mechanisms: Access Levels and Ingress/Egress rules.

Access Levels may be applied to the entire perimeter or used within fine grained ingress/egress rules.

AMEX defines the following Access Levels to provide necessary access to the VPC Service Control perimeter from an approved context:

* **AMEX Corporate IPs & Location**

| **Amex corporate IPs and Location** | | |
| --- | --- | --- |
| **Name** | **corp\_ip\_and\_us\_location** | |
| **Description:** | Allows access to the VPC Service Control perimeter from callers to GCP APIs if they originate from the AMEX corporate IP addresses (public) and can be detected as originating from the US. This allows users to view their GCP project in the GCP Cloud Console. Without this access level users would be denied any information in the console.  Please note that Cloud IAM is still enforced so users will only be able to take the approved IAM actions | |
| **Configuration:** | **IP Addresses:** | Corp IP’s |
| **Regions:** | US |
| **Applied:** |  | Via Ingress Rule |

* **Partner Service Accounts**

| **Access Level : partner\_service\_account\_and\_ip** | | |
| --- | --- | --- |
| **Name** | **partner\_ip\_and\_us\_location** | |
| **Description:** | Allows Partner service accounts such as Snowflake (list others here), which originates outside of the GCP Service Perimeter, the ability to execute HTTP methods against certain AMEX Protected APIs(GCS, BQ..<list here> ) | |
| **Configuration:** | **IP Address:** | Snowflake\_IP |
|  | **Regions:** | US |
| **Applied:** |  | Via Ingress Rule |

### 4.8.3 Ingress & Egress Rules

Access is granted into a service perimeter through one of two mechanisms: Access Levels and Ingress & Egress rules.

The Access Level for a VPC Service Control perimeter affects the entire perimeter. The [Ingress & Egress rules](https://cloud.google.com/vpc-service-controls/docs/ingress-egress-rules#definition_of_ingress_and_egress) add additional context allowing access to and from the resources and clients protected by service perimeters.

* **Ingress:** Refers to any access by an API client from outside the service perimeter to resources within a service perimeter.
* **Egress:** Refers to any access that involves an API client or resources within the service perimeter and resources outside a service perimeter.

#### *4.8.3.1 Prod Perimeter*

AMEX will configure the following ingress rules and apply them to the Prod perimeter identified in section 4.7.1. There are no Egress rule definition.

* **User Console Access - All Methods All Resources - Ingress**

| **Rule Details** | | |
| --- | --- | --- |
| **Description** | Allows user identities access to GCP APIs protected by the VPC Service Control Perimeter, which meet the [Access Level Restrictions](#_hhcitodqrqw8) (see section 4.7.2)  Note: this ingress rule allows access to all services and all methods. AMEX uses IAM to prevent unwanted access to unapproved services or unapproved actions. The purpose of this ingress rule is to only allow users access to the perimeter from corporate IP addresses. | |
| **Ingress From:** | **Identity Type** | ANY\_USER\_ACCOUNT |
| **Sources** | Access Level = corp\_ip\_and\_us\_location |
| **Ingress To:** | **Operations / Service Name** | \"\*\" - All Services are permitted |
| **Operations / Method Selectors** | \"\*\" - All Methods are permitted |
| **Resources** | \"\*\" - All Projects are permitted |

* **Partner(Snowflake) - Ingress**

| **Rule Details** | | |
| --- | --- | --- |
| **Description** | Allows Snowflake ingress into the perimeter to access the < BigQuery API >. | |
| **Ingress From:** | **Identity Type** | Selective\_SERVICE\_ACCOUNT |
| **Sources** | Access Level = partner\_ip\_and\_us\_location  And Partner(project Number) |
| **Ingress To:** | **Operations / Service Name** | \"bigquery.googleapis.com\" - BigQuery Service is permitted |
| **Operations / Method Selectors** | \"\*\" - All Methods are permitted(or selective methods) |
| **Resources** | Selective Project Numbers are permitted |

* **Shared Services Service Accounts**

| **Shared Services(Terraform Enterprise,AD) - All Services All Resources - Ingress** | | |
| --- | --- | --- |
| **Description** | Allows the AMEX Service Accounts used by AMEX IaC CICD Pipeline (Terraform Enterprise Workspaces) from the Shared Services VPC for any service to any project in this perimeter. | |
| **Ingress From:** | **Identity Type** | *List of GCP Service Accounts that are used by TFE/Shared Services* |
| **Sources** | Resource - AMEX Shared Services project (number) |
| **Ingress To:** | **Operations / Service Name** | \"\*\" - All Services are permitted |
| **Operations / Method Selectors** | \"\*\" - All Methods are permitted (all methods). |
| **Resources** | \"\*\" - All Projects are permitted |

* **Anonymized Data Pipeline**

| **Anonymized Data Ingestion Pipeline - Ingress** | | |
| --- | --- | --- |
| **Description** | Allows the AMEX Service Accounts used by the Anonymization Data Pipeline from the amex-data-anonymized-shared VPC for selective service and method to selective project in this perimeter. | |
| **Ingress From:** | **Identity Type** | *List of GCP Service Accounts that are used by Anonymization Data Pipeline* |
| **Sources** | Resource - AMEX anonymized data project (number) |
| **Ingress To:** | **Operations / Service Name** | Selective services(bigquery, gcs, pub-sub) |
| **Operations / Method Selectors** | Selective methods such as READ |
| **Resources** | Selective prod-dataset projects |

#### *4.8.3.2 Non-Prod Perimeter*

AMEX will configure the following ingress rules and apply them to the Non-Prod perimeter identified in section 4.7.1. There are no Egress rule definition.

* **User Console Access - All Methods All Resources - Ingress**

| **Rule Details** | | |
| --- | --- | --- |
| **Description** | Allows user identities access to GCP APIs protected by the VPC Service Control Perimeter, which meet the [Access Level Restrictions](#_hhcitodqrqw8) (see section 4.7.2)  Note: this ingress rule allows access to all services and all methods. AMEX uses IAM to prevent unwanted access to unapproved services or unapproved actions. The purpose of this ingress rule is to only allow users access to the perimeter from corporate IP addresses. | |
| **Ingress From:** | **Identity Type** | ANY\_USER\_ACCOUNT |
| **Sources** | Access Level = corp\_ip\_and\_us\_location |
| **Ingress To:** | **Operations / Service Name** | \"\*\" - All Services are permitted |
| **Operations / Method Selectors** | \"\*\" - All Methods are permitted |
| **Resources** | \"\*\" - All Projects are permitted |

* **Shared Services Service Accounts**

| **Shared Services(Terraform Enterprise,AD) - All Services All Resources - Ingress** | | |
| --- | --- | --- |
| **Description** | Allows the AMEX Service Accounts used by AMEX IaC CICD Pipeline (Terraform Enterprise Workspaces) from the Shared Services VPC for any service to any project in this perimeter. | |
| **Ingress From:** | **Identity Type** | *List of GCP Service Accounts that are used by TFE/Shared Services* |
| **Sources** | Resource - AMEX Shared Services project (number) |
| **Ingress To:** | **Operations / Service Name** | \"\*\" - All Services are permitted |
| **Operations / Method Selectors** | \"\*\" - All Methods are permitted (all methods). |
| **Resources** | \"\*\" - All Projects are permitted |

* **Anonymized Data Pipeline**

| **Anonymized Data Ingestion Pipeline - Ingress** | | |
| --- | --- | --- |
| **Description** | Allows the AMEX Service Accounts used by the Anonymization Data Pipeline from the amex-data-anonymized-shared VPC for selective service and methods to selective project in this perimeter. | |
| **Ingress From:** | **Identity Type** | *List of GCP Service Accounts that are used by Anonymization Data Pipeline* |
| **Sources** | Resource - AMEX anonymized data project (number) |
| **Ingress To:** | **Operations / Service Name** | Selective services(bigquery, gcs, pub-sub) |
| **Operations / Method Selectors** | Selective methods such as READ and Write |
| **Resources** | Selective nonprod-dataset projects |

#### *4.8.3.3 Shared Services Perimeter*

AMEX will configure the following ingress rules and apply them to the Shared Services perimeter identified in section 4.7.1.

* **User Console Access - All Methods All Resources - Ingress**

| **Rule Details** | | |
| --- | --- | --- |
| **Description** | Allows user identities access to GCP APIs protected by the VPC Service Control Perimeter, which meet the [Access Level Restrictions](#_hhcitodqrqw8) (see section 4.7.2)  Note: this ingress rule allows access to all services and all methods. AMEX uses IAM to prevent unwanted access to unapproved services or unapproved actions. The purpose of this ingress rule is to only allow users access to the perimeter from corporate IP addresses. | |
| **Ingress From:** | **Identity Type** | ANY\_USER\_ACCOUNT |
| **Sources** | Access Level = corp\_ip\_and\_us\_location |
| **Ingress To:** | **Operations / Service Name** | \"\*\" - All Services are permitted |
| **Operations / Method Selectors** | \"\*\" - All Methods are permitted |
| **Resources** | \"\*\" - All Projects are permitted |

* **Shared Services Service Accounts - All Methods All Resources - Egress**

| **AMEX TFE/Shared-service Service Account - All Perimeters & All Services Egress** | | |
| --- | --- | --- |
| **Description:** | Allows all AMEX services accounts being used by the AMEX IaC CICd pipeline (TFE workspace associated GCP service accounts) to egress from the TFE perimeter to all AMEX projects for all services. | |
| **Egress From:** | Identities | [List of AMEX IaC Pipeline Service Accounts]  This list will be maintained by AMEX. |
| **Egress To:** | Operations | Service Name: \"\*\" All Services  Method Selectors: \"\*\" All Methods |
|  | Resources | [List of AMEX projects (by number)]  This list will be maintained by AMEX. |

#### 

#### *4.8.3.4 Anonymized Data Perimeter*

AMEX will configure the following ingress rules and apply them to the Anonymized Data perimeter identified in section 4.7.1.

* **User Console Access - All Methods All Resources - Ingress**

| **Rule Details** | | |
| --- | --- | --- |
| **Description** | Allows user identities access to GCP APIs protected by the VPC Service Control Perimeter, which meet the [Access Level Restrictions](#_hhcitodqrqw8) (see section 4.7.2)  Note: this ingress rule allows access to all services and all methods. AMEX uses IAM to prevent unwanted access to unapproved services or unapproved actions. The purpose of this ingress rule is to only allow users access to the perimeter from corporate IP addresses. | |
| **Ingress From:** | **Identity Type** | ANY\_USER\_ACCOUNT |
| **Sources** | Access Level = corp\_ip\_and\_us\_location |
| **Ingress To:** | **Operations / Service Name** | \"\*\" - All Services are permitted |
| **Operations / Method Selectors** | \"\*\" - All Methods are permitted |
| **Resources** | \"\*\" - All Projects are permitted |

* **Data Ingestion Pipeline Service Accounts - Certain Methods Certain Resources To Prod - Egress**

| **Data Ingestion Pipeline Service Accounts - Certain Methods Certain Resources - Egress** | | |
| --- | --- | --- |
| **Description:** | Allows certain AMEX services accounts being used by the AMEX Data Ingestion Pipeline for Data Anonymization of Prod Datasets to egress from the Data Anonymization perimeter to certain AMEX projects for certain services. | |
| **Egress From:** | Identities | [List of AMEX Data Ingestion Pipeline Service Accounts]  This list will be maintained by AMEX. |
| **Egress To:** | Operations | Service Name: Certain Services(pub-sub, gcs, bigquery)  Method Selectors: Only Read Operation |
|  | Resources | [List of AMEX Prod Dataset projects (by number)]  This list will be maintained by AMEX. |

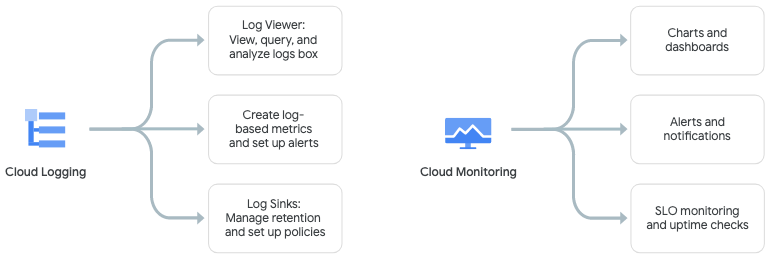
* **Data Ingestion Pipeline Service Accounts - Certain Methods Certain Resources To Non-Prod - Egress**

| **Data Ingestion Pipeline Service Accounts - Certain Methods Certain Resources to Non-Prod - Egress** | | |
| --- | --- | --- |
| **Description:** | Allows certain AMEX services accounts being used by the AMEX Data Ingestion Pipeline for Data generation of Non-Prod Datasets to egress from the Data Anonymization perimeter to certain AMEX projects for certain services. | |
| **Egress From:** | Identities | [List of AMEX Data Ingestion Pipeline Service Accounts]  This list will be maintained by AMEX. |
| **Egress To:** | Operations | Service Name: Certain Services(pub-sub, gcs, bigquery)  Method Selectors: Only Read and Write Operation |
|  | Resources | [List of AMEX Non-Prod Dataset projects (by number)]  This list will be maintained by AMEX. |

# 

# 5. Logging and Monitoring

Cloud Logging and Cloud Monitoring provide your IT Ops/SRE/DevOps teams with out-of-the box observability needed to monitor your infrastructure and applications. Cloud Logging automatically ingests Google Cloud audit and platform logs so that you can get started right away. Cloud Monitoring provides a view of all Google Cloud metrics at zero cost and integrates with a variety of providers for non Google Cloud monitoring.



## 

## 5.1 Current architecture

Logging and monitoring of workloads running on-premise or in the cloud is a critical component of a healthy IT infrastructure. Understanding a user’s experience, how an application is performing, or how updates have impacted usability is essential to realizing the potential of countless hours of development work. Beyond this, ensuring logging of access or changes to applications and the underlying cloud infrastructure is often a legal requirement in regulated industries, such as finance. While proper IAM controls and best practices attempt to mitigate the risk of a security incident, it’s also necessary to have insight into actions taken when the controls are not sufficient. Logging and Monitoring provides the foundation layer of an effective detective control solution.

Google Cloud Platform provides options across all of the different aspects of cloud logging and monitoring, including application performance metrics, administrative activity logs, data access logs, and alerting, amongst others. Setting up the environment to best leverage these capabilities will ensure that the right teams, whether they be developers or security, get the information they need in a timely and seamless manner.

Currently AMEX is using on-prem solutions for Logging and Monitoring.

## 5.2 Logging requirements

* Usage of GCP Cloud Logging as default for storage, search, analysis and alerts on logging.
* Enable audit and other compliance related logs for all GCP projects. @TODO: List all compliance related logs for AMEX.
* Create a Cloud Storage bucket with low cost on-demand storage for storing log files.
* BigQuery to be used for log analytics for logs stored in Cloud Storage buckets.

## 5.3 Logging findings and recommendations

* Applications will be logically grouped into projects and segregated into lab, dev, test, and prod projects.
* Logging and metrics measurement happens within individual projects. It’s proposed to create separate projects to consolidate metrics across all non-prod (lab, dev, test), and prod environments. For logging depending on the log type, export it to either GCS or BigQuery. Logs that need to be retained for compliance purposes can be exported to GCS and ones that are needed for analysis can be exported to BigQuery. Another consideration is the volume of logs; it’s recommended to have low volume logging to go through Cloud Logging and the higher volume can go straight to BigQuery or GCS.
* In the future, Production logs may be replicated (sinked) and made available to the on premises SIEM. Non Production logs can be replicated (sinked) and made available to the on premises SIEM based on specific business requirements.
* BigQuery will be leveraged for further analysis and correlation between monitored alerts.
* Premium Cloud Logging stores logs and metrics for 30 days (audit logs for 400 days) by default; therefore logs beyond 30 days are to be exported to GCS to retain it. \_Default settings can be changed as business needs require.
* Centralized logging is a desirable pattern on GCP.
* While it is possible to send all logs to on-premise systems, it may be preferable to leverage GCP’s capabilities to filter the logs before sending them. This limits the volume of logs being sent, and allows teams to focus on the relevant information.
* Need to facilitate access by teams to logs applicable to their applications.
* Access Transparency Logs are enabled for the organization.

## 5.4 Categories of Logs

Logging receives, indexes, and stores log entries from many sources, including Google Cloud, VM instances running the Logging agent, and other cloud services providers.

The following sections broadly categorize the kinds of logs supported by Cloud Logging:

### 5.4.1 Google Cloud platform logs

Google Cloud platform logs are service-specific logs that can help you debug and troubleshoot issues, as well as better understand the Google Cloud services you're using.

The Google Cloud platform logs visible to you in Cloud Logging vary, depending on which Google Cloud monitored resources you're using in your Cloud project, folder, or organization.

To learn more about the available Google Cloud platform logs, go to [Using platform logs](https://cloud.google.com/logging/docs/api/platform-logs). Note that some Google Cloud platform logs are sent by an agent.

#### 5.4.3.1 VPC Flow Logs

VPC Flow Logs records a sample of network flows sent from and received by [VM instances](https://cloud.google.com/compute/docs/instances), including instances used as [Google Kubernetes Engine nodes](https://cloud.google.com/kubernetes-engine/docs). These logs can be used for network monitoring, forensics, real-time security analysis, and expense optimization. For details, see [Using VPC Flow Logs](https://cloud.google.com/vpc/docs/using-flow-logs).

#### 5.4.3.1 Firewall Rule Logs

Firewall Rules Logging lets you audit, verify, and analyze the effects of your firewall rules. For example, you can determine if a firewall rule designed to deny traffic is functioning as intended. Firewall Rules Logging is also useful if you need to determine how many connections are affected by a given firewall rule. For details, see [Firewall Rule Logs](https://cloud.google.com/vpc/docs/firewall-rules-logging).

Google recommends enabling VPC Flow logs, Firewall Rule Logs and other service-specific platform logs for the services used by AMEX on GCP.

### 5.4.2 User written logs

User-written logs are written to Cloud Logging by the user in one of the common ways that users write their own logs: using the [logging agent](https://cloud.google.com/logging/docs/view/available-logs#agent-logs), the [Cloud Logging API](https://cloud.google.com/logging/docs/reference/api-overview), or the [Cloud Logging client libraries](https://cloud.google.com/logging/docs/reference/libraries). User-written logs contain information related to their custom applications and services.

The [Logging agent](https://cloud.google.com/logging/docs/agent) is a process that collects logs from user applications and writes them to the Cloud Logging API. For more information, see [Default logging agent logs](https://cloud.google.com/logging/docs/agent/default-logs).

AMEX will be using the default logging agent. @TODO Validate

### 5.4.3 Security logging

Cloud Logging provides two kinds of security-related logs, Cloud Audit Logs and Access Transparency logs.

#### 5.4.3.1 Audit logs

Google Cloud services write audit logs that record administrative activities and accesses within your Google Cloud resources. Audit logs help you answer "who did what, where, and when?" within your Google Cloud resources with the same level of transparency as in on-premises environments. Enabling audit logs helps your security, auditing, and compliance entities monitor Google Cloud data and systems for possible vulnerabilities or external data misuse.

Cloud Audit Logs provides the following audit logs for each Cloud project, folder, and organization:

* [Admin Activity audit logs](https://cloud.google.com/logging/docs/audit#admin-activity)
* [Data Access audit logs](https://cloud.google.com/logging/docs/audit#data-access)
* [System Event audit logs](https://cloud.google.com/logging/docs/audit#system-event)
* [Policy Denied audit logs](https://cloud.google.com/logging/docs/audit#policy_denied)

Cloud Logging and Monitoring can be used in conjunction to create alerts on audit log events and provides a robust solution for awareness and visibility across the organization. Audit logs will be stored in the centralized Cloud Storage buckets for AMEX. Each environment folder will have a project to centralize and store logs for all of the projects in the environments. @TODO Validate

**Admin Activity audit logs**

Admin Activity audit logs contain log entries for API calls or other administrative actions that modify the configuration or metadata of resources. For example, these logs record when users create VM instances or change Identity and Access Management permissions.

Admin Activity audit logs are always written; you can't configure, exclude, or disable them. Even if you disable the Cloud Logging API, Admin Activity audit logs are still generated.

**Data Access audit logs**Data Access audit logs contain API calls that read the configuration or metadata of resources, as well as user-driven API calls that create, modify, or read user-provided resource data.

Publicly available resources that have the Identity and Access Management policies [allAuthenticatedUsers](https://cloud.google.com/iam/docs/overview#allauthenticatedusers) or [allUsers](https://cloud.google.com/iam/docs/overview#allusers) don't generate audit logs. Resources that can be accessed without logging into a Google Cloud, Google Workspace, Cloud Identity, or Drive Enterprise account don't generate audit logs. This helps protect end-user identities and information.

Data Access audit logs-- except for BigQuery Data Access audit logs-- are disabled by default because audit logs can be quite large and can be costly. If you want Data Access audit logs to be written for Google Cloud services other than BigQuery, you must explicitly enable them. Enabling the logs might result in your Cloud project being charged for the additional logs usage. For instructions on enabling and configuring Data Access audit logs, see [Configure Data Access logs](https://cloud.google.com/logging/docs/audit/configure-data-access).

**System Event audit logs**

System Event audit logs contain log entries for Google Cloud actions that modify the configuration of resources. System Event audit logs are generated by Google systems; they aren't driven by direct user action.

System Event audit logs are always written; you can't configure, exclude, or disable them.

**Policy Denied audit logs**

Policy Denied audit logs are recorded when a Google Cloud service denies access to a user or [service account](https://cloud.google.com/iam/docs/service-accounts) because of a security policy violation. The security policies are determined by VPC Service Controls, which provides the Policy Denied audit logs to Cloud Logging.

Policy Denied audit logs are generated by default and your Cloud project is charged for the logs storage. You can't disable Policy Denied audit logs, but you can use [exclusion filters](https://cloud.google.com/logging/docs/routing/overview#exclusions) to prevent Policy Denied audit logs from being ingested and stored in Cloud Logging.

**Audit Logs**

| **Google PSO Best Practice Recommendation** | |
| --- | --- |
| **Log Category** | **Recommendation** |
| **Admin Activity audit logs** | Use Admin Activity audit logs to view API calls or administrative actions that modify the configuration or metadata of resources. These logs should be written to the GCP Cloud storage bucket. @TODO Validate |
| **Data Access audit logs** | Use Data Access audit logs to view API calls that read the configuration of metadata of resources, as well as user-driven API calls that create, modify, or read user-provided resource data. These logs should be written to the GCP Cloud storage bucket. @TODO Validate |
| **System Event audit logs** | Use System Event audit logs to view log entries for Google Cloud administrative actions that modify the configuration of resources. These logs should be written to the GCP Cloud storage bucket. @TODO Validate |
| **Policy Denied audit logs** | Use Policy Denied audit logs to view log entries when a Google Cloud service denies access to a user or [service account](https://cloud.google.com/iam/docs/service-accounts) because of a security policy violation. These logs should be written to the GCP Cloud storage bucket. @TODO Validate |

#### 5.4.3.2 Access Transparency logs

Security, transparency, and data protection are at the core of how Google designs and builds its products. All customers of Google Cloud own their data and have complete control on how it is used. [Google Cloud Trust Principles](https://cloud.google.com/privacy) summarize Google's commitment to protecting the privacy of customer content that is stored in Google Cloud.

Access Transparency is a part of Google's long-term commitment to transparency and user trust. Access Transparency logs record the actions that Google personnel take when accessing customer content.

Access Transparency logs give you different information than [Cloud Audit Logs](https://cloud.google.com/logging/docs/audit). Cloud Audit Logs record the actions that members of your Google Cloud organization have taken in your Google Cloud resources, whereas Access Transparency logs record the actions taken by Google personnel.

Google personnel are strictly restricted in what is visible to them. All access to customer content requires a valid justification. Access Transparency log entries include details such as the affected resource and action, the time of the action, the reason for the action, and information about the accessor.

You might need Access Transparency logs for the following reasons:

* Verifying that Google personnel are accessing your content only for valid business reasons, such as fixing an outage or attending to your support requests.
* Verifying that Google personnel haven't made an error while carrying out your instructions.
* Verifying and tracking compliance with legal or regulatory obligations.
* Collecting and analyzing tracked access events through an automated security information and event management (SIEM) tool.

For more information, including how to enable Access Transparency logs, see [Access Transparency](https://cloud.google.com/cloud-provider-access-management/access-transparency/docs/overview).

Google recommends enabling Access Transparency logs.

### 5.4.4 Multi-cloud and hybrid-cloud logging

Cloud Logging can ingest logs arriving from other cloud services providers, including Microsoft Azure and Amazon Web Services (AWS). These logs are called "multi-cloud" logs.

Cloud Logging also suppo rts logs from your on-premises infrastructure and apps. If extending Cloud Logging to include your on-premises resources, the logs that are received by Cloud Logging are known as "hybrid-cloud" logs.

For details on how to ingest logs from your on-premises or other cloud sources, see [Logging on-premises resources](https://cloud.google.com/solutions/logging-on-premises-resources-with-blue-medora).

## 5.5 Using and managing Logs

The following sections provide basic information about using and managing logs in Cloud Logging.

### 

### 5.5.1 Log entry structure

Every log entry is characterized by the following information:

* A log name. This includes the identifier of the Cloud project, folder, or organization that contains the log entry and the identifier of the log type, **LOG\_ID**.
* The resource from which the log entry originated. This consists of a resource type from the [Monitored resource list](https://cloud.google.com/logging/docs/api/v2/resource-list) and corresponding label values that identify a specific instance.
* A timestamp.
* A payload, which can be represented as one of textPayload, jsonPayload, or (for some Google Cloud services) protoPayload.

### 

### 5.5.2 Viewing logs

There are several ways to view your log entries:

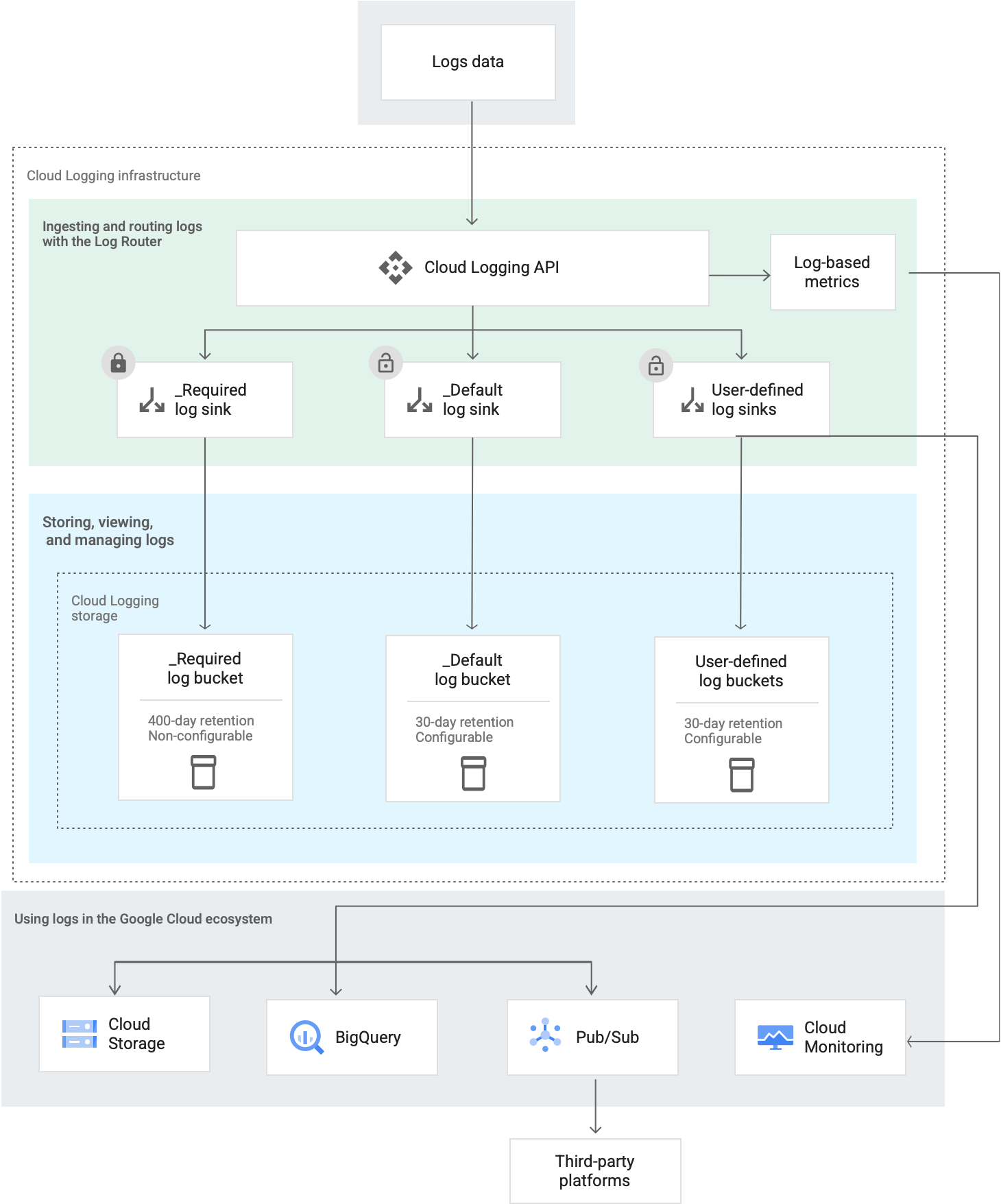
* To read log entries using the Google Cloud Console, see [Using the Logs Explorer](https://cloud.google.com/logging/docs/view/logs-viewer-interface).
* To read log entries through the Logging API, see the [entries.list](https://cloud.google.com/logging/docs/reference/v2/rest/v2/entries/list) method.
* To read log entries using the gcloud command-line tool, see [Reading log entries](https://cloud.google.com/logging/docs/api/gcloud-logging#reading_log_entries).

### 

### 5.5.3 Routing logs

You can control how your log entries are routed and stored. The Log Router checks each log entry against existing rules to determine which log entries to discard, which log entries to store in Cloud Logging, and which log entries to include in exports to other destinations.

At a high level, this is how Cloud Logging routes and stores log entries:



#### 5.5.3.1 Log Router

Cloud Logging receives log entries through the [Cloud Logging API](https://cloud.google.com/logging/docs/reference/api-overview) where they pass through the Log Router. The [sinks](https://cloud.google.com/logging/docs/routing/overview#sinks) in the Log Router check each log entry against the existing [inclusion filter](https://cloud.google.com/logging/docs/routing/overview#inclusion-filters) and [exclusion filters](https://cloud.google.com/logging/docs/routing/overview#exclusions) that determine which destinations, including Cloud Logging buckets, that the log entry should be sent to. You can use combinations of sinks to route logs to multiple destinations.

#### 5.5.3.2 Sinks

Using sinks, you can route some or all of your logs to [supported destinations](https://cloud.google.com/logging/docs/routing/overview#destinations). Some of the reasons that you might want to control how your logs are routed include the following:

* To store logs that are unlikely to be read but that must be retained for compliance purposes.
* To organize your logs in buckets in a format that is useful to you.
* To use big-data analysis tools on your logs.
* To stream your logs to other applications, other repositories, or third parties.

You usually [create sinks](https://cloud.google.com/logging/docs/export/configure_export_v2) at the Cloud project level, but if you want to combine and route logs from the resources contained by a Google Cloud organization or folder, you can create [aggregated sinks](https://cloud.google.com/logging/docs/export/aggregated_sinks).

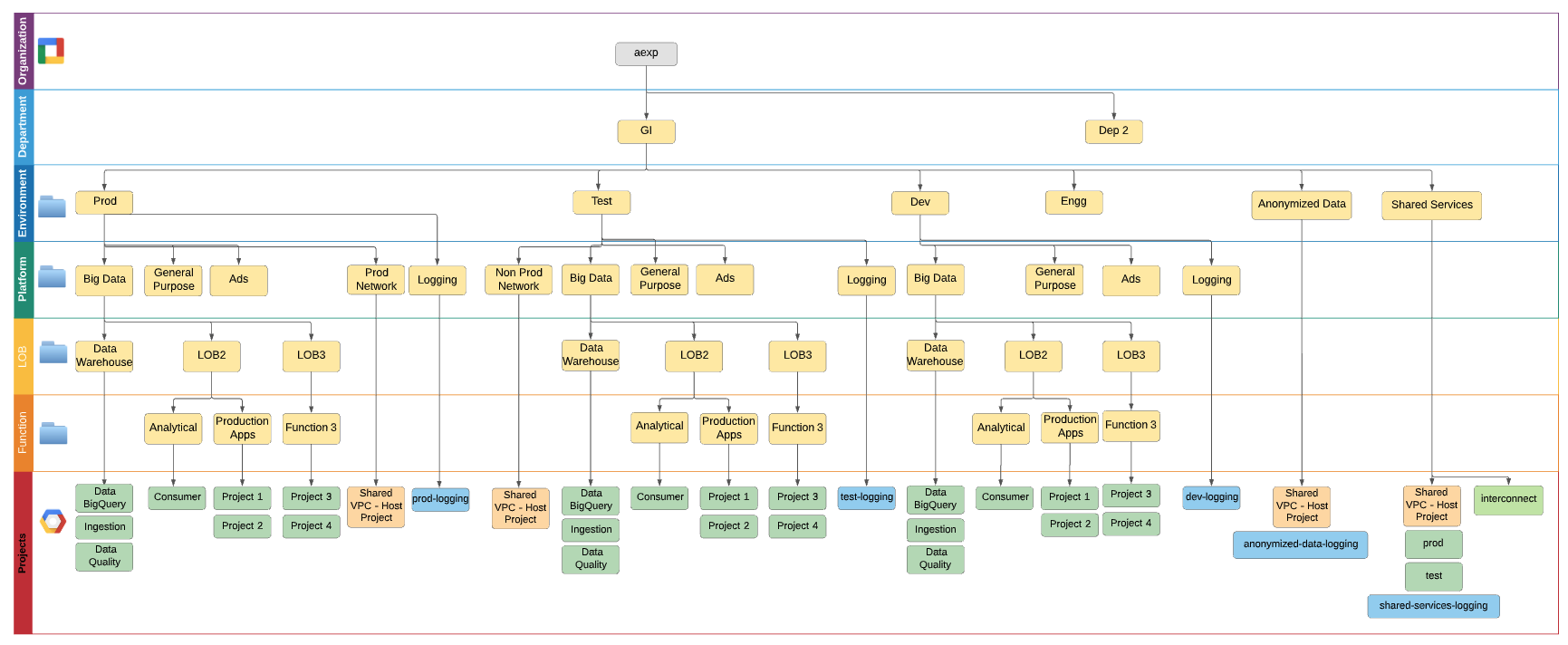
You can't route log entries that Logging received before your sink was created because routing happens as logs pass through the Logging API, and new routing rules only apply to logs written after those rules have been created. If you need to route log entries retroactively, see [Copy logs](https://cloud.google.com/logging/docs/routing/copy-logs).

In order to ensure that all relevant logs are captured, and to ensure that sinks cannot be modified by end users, AMEX should make use of [aggregated log sinks](https://cloud.google.com/logging/docs/export/aggregated_sinks) to implement logging throughout the organization. Folder level aggregate log sinks would likely be a good candidate to centralize logs on the basis of the environments.

Google Recommended Best Practices:

* Use folder level aggregated log sinks to centralize logs on the basis of the environments for auditing, retention, and non-repudiation purposes.
* BigQuery is a common pattern for log analysis as it allows powerful analytics using SQL queries.
* Cloud Storage for longer retention periods, and vault / non-repudiation
  + Object lifecycle management for archiving objects based on retention.
  + Bucket Lock for vaulting.
* Pub/Sub for streaming to third-party tools, or triggering automated actions.

AMEX will leverage default Cloud Logging and Cloud Storage Buckets for logs. In a future state, sending log data to BigQuery is an option being considered. In order to query the required logs and perform analytic tasks, BigQuery can be leveraged. Logs can be sent directly to BQ using a log sink, or the logs can be read from Google Cloud Storage directly. A common pattern is to ingest the logs into BigQuery directly and after some defined period, export the data out of BQ to GCS, for archival purposes, [using a scheduled export job](https://cloud.google.com/bigquery/docs/exporting-data#api) or ingest data in both active and archival storage simultaneously with different expiration periods. Another potential pattern includes the use of two log sinks, one to store the archival data in GCS, and the other to store active data for querying in BQ. Dataflow can also be utilized for enrichment of both batch and streaming data and sent to BQ for downstream analytics. Following diagram explains the current logging structure:



#### 5.5.3.3 Regionalization

Default Log Buckets (different from user defined Cloud Storage Buckets) are regional resources. In addition to these regions, you also have the option to set the location to global, which means that you don't need to specify where your logs are physically stored.

Google recommends setting the location to global. @TODO Validate if there is a concern on geographic location for logs.

### 

### 5.5.4 Retention

Cloud Logging retains logs according to retention rules applying to the log bucket type where the logs are held.

You can configure Cloud Logging to retain logs between 1 day and 3650 days. Custom retention rules apply to all the logs in a bucket, regardless of the log type or whether that log has been copied from another location.

AMEX has a requirement of keeping system and application logs for 5 years @TODO Validate number of years. Logs will be exported every 30 days to Google Cloud Storage for instances where longer retention is needed. [Lifecycle Management](https://cloud.google.com/storage/docs/xml-api/put-bucket-lifecycle) for GCS will need to be configured using terraform to expire logs older than five years.

## 5.6 Log Compliance

It is highly important to make sure logs are centrally located, secure, and can only be accessed by the right resources.

Log compliance is also a very important topic for AMEX. Here are some ways to achieve that:

* **Separation of duties**:
  + Use **aggregated exports** to a **different project** that is **owned by a different team**
  + **Cloud Storage** is recommended for **log archival**
* **Least privilege**
  + Grant the **minimum required permissions** on this project, so users **cannot remove the logs**.
  + Enforce **Bucket Policy Only** (effectively **disabling object-level permissions**)
* **Non-repudiation**
  + In case of **Cloud Storage**, configure [**Bucket Lock**](https://cloud.google.com/storage/docs/bucket-lock) to ensure retention of objects within the bucket. This **prevents deleting** the bucket or underlying objects **by anyone** until all objects **have reached their retention**.
  + Additionally, enable **object versioning** on the bucket

### 5.6.1 Inadvertent PII data within Logs

Google generated logs take care not to reveal PII data within the content of their logs. However, situations have been known to occur when customer application logs or query details include inadvertent PII data. As such, access to logs should be managed and restricted when dealing with certain log types.

## 5.7 Monitoring requirements

* Email alerts for all direct login access in Google console using admin account
* Email alerts for new Admin user additions
* Incident reporting through ITSM API(Amex to provide ITSM API for creating new incident tickets)
* Dynatrace and Splunk to be leveraged in future for JVM monitoring and Availability reports

## 

## 5.8 Monitoring findings and recommendations

It has been determined that the current monitoring architecture should utilize AMEX’s existing on-premise infrastructure. This infrastructure is used throughout the organization. There is potential interest in making use of Google Cloud Monitoring and [MQL](https://cloud.google.com/monitoring/mql) in the future.

### 5.8.1 Alerting

Alertsin the Cloud Monitoring Console allow you to determine when your applications are not operating normally. When external events trigger [**conditions**](https://cloud.google.com/monitoring/alerts/#conditions) in one of your alerting policies, Stackdriver Monitoring displays an [**incident**](https://cloud.google.com/monitoring/alerts/#incidents) and sends [**notifications**](https://cloud.google.com/monitoring/alerts/#notifications) to people or third-party notification services. Monitoring Admins can acknowledge receipt of the notification. A list of GCP agent metrics for alerting can be found [here](https://cloud.google.com/monitoring/api/metrics#gcp).

AMEX will leverage Cloud Monitoring to enable alerting in the GCP environment.

### 5.8.2 Log-based alerting

When you want to be notified any time a specific message occurs in a log, use log-based alerts. Log-based alerts are well suited for catching security-related events in logs, like the following:

* You want to be notified if an event appears in an audit log; for example, a human user accesses the security key of a service account.
* Your application writes deployment messages to logs, and you want to be notified when a deployment change is logged.

Log-based alerts are well suited for events that you expect to be both rare and important. You don't want to know about a trend or pattern; you want to know that something occurred.

For information about creating log-based alerts, see [Using log-based alerts](https://cloud.google.com/logging/docs/alerting/log-based-alerts).

[User-defined log-based metrics](https://cloud.google.com/logging/docs/logs-based-metrics#user-metrics) are computed from log entries in both included and excluded logs. If you create an alerting policy based on a user-defined log-based metric, then the policy monitors data from all logs.

[System-defined log-based metrics](https://cloud.google.com/logging/docs/logs-based-metrics#system_logs-based_metrics) are computed only from included logs. If you create an alerting policy based on a system-defined log-based metric, then the policy monitors data only from included logs.

Google recommends:

* Use Log-based alerting for all security-related events.
* Use User-defined log based metrics to create alerting policies for any change in IAM (ex - permission updated for a custom role), Firewall rules etc.

### 

### 5.8.3 Dashboards and Charts

Monitoring Cloud environments is a fundamental Site Reliability Engineering practice. Google offers the ability to build custom dashboards that can measure how well your applications and services are operating. Google has made available service specific dashboards that can be leveraged and further customized on [github](https://github.com/GoogleCloudPlatform/monitoring-dashboard-samples/tree/master/dashboards). Dashboards can be built per project, per group of projects or at the organization level. For further details, please refer to Google’s [documentation](https://cloud.google.com/monitoring/dashboards).

# 6. Security

IT Security is the practice of protecting critical systems and sensitive information from a variety of risks such as unauthorized access, unavailability of critical systems, exfiltration of data, etc. Operational measures are designed within an Enterprise to combat threats against networked systems and applications, whether those threats originate from inside or outside of an organization. The practice focuses around the tenet of ensuring confidentiality, integrity and availability of information for an organization to achieve its business objectives.

Many Cybersecurity frameworks satisfy IT Security requirements by looking at an Enterprise’s business objectives, mapping these to legal, regulatory, and internal policies. Then, identifying threats and vulnerabilities within a technical environment to define operating methodologies within the environment required to protect that environment within the organization’s risk appetite. Key to these frameworks are viewing controls around 5 core functions: Identify, Protect, Detect, Respond and Recover. **Identify** focuses on identification of the processes and assets needing protection. **Protect** focuses on the safeguards available to prevent threats from materializing and vulnerabilities from being exploited. **Detect** focuses on techniques available to identify actual incidents. **Respond** focuses on the techniques available that can contain impacts of incidents. Finally, **Recover** focuses on the techniques that can restore capabilities. In that context, the security section of the Technical Design Document focuses on identifying various GCP, 3rd party and Amex security solutions and practices that **together** provide an effective layered security framework necessary to secure Amex’s GCP environment.

The below summary is a subset of services identifying how the collection of GCP and 3rd party services along with Amex’s practices work together to protect the GCP environment.

| Service | Identify | Protect | Detect | Respond | Recover |
| --- | --- | --- | --- | --- | --- |
| GCP IAM |  | X | X |  |  |
| GCP Encryption in Transit |  | X |  |  |  |
| GCP Encryption at Rest |  | X |  |  |  |
| GCP Organization Policies |  | X |  |  |  |
| GCP Network Constraints |  | X | X |  |  |
| GCP Operations Suite |  |  | X | X |  |
| GCP Access Transparency |  |  | X |  |  |
| GCP Data Loss Prevention |  | X | X |  |  |
| 3rd party OS agents |  | X | X | X |  |
| GCP VPC SC |  | X |  |  |  |
| Hashicorp Terraform | X |  | X |  | X |
| Hashicorp Terraform Sentinel |  | X |  |  |  |
| Hashicorp Vault |  | X |  |  |  |
| GCP Security Command Center | X |  | X | X |  |
| GCP Asset Inventory | X |  |  |  |  |
| ServiceNow ITSM | X |  |  | X |  |
| Palo Alto Prisma Cloud | X |  | X | X |  |
| SIEM |  |  | X |  |  |
| Amex Operations Team |  | X | X | X | X |

## 6.1 Current Architecture

Amex has a number of IT Security Policies and Standards developed to protect their on premises environment. As a general principle, unless specifically called out as being a Cloud Anti-Pattern, it is recommended that enterprises leverage universal methods to protect their overall environment across all their environments. As such, it is recommended that Amex extend existing IT Security Policies and Standards to Public Cloud environments, modifying them as necessary to accommodate Cloud specific differentiators.

For example, Amex currently leverages Prisma Cloud in helping secure their on premise workloads. Prisma Cloud Workload Protection tools provide a consistent set of multi-cloud based security solutions that can provide a consistent and standardized set of scanning, monitoring, response and recovery tools available to help secure Amex’s IT assets. These tools can effectively complement GCP’s internal solutions to provide an independent and holistic perspective of an enterprise’s security posture.

### 6.1.1 Amex Approved Services

Many regulated enterprises go through an assessment process to approve services before they can be used in production environments. Amex will assess in scope GCP services and then approve them by their central security team to accommodate expected deployment use cases.

The process to approve new cloud native service/application patterns typically involves multiple steps and input for cross-team approvals, which may generate lag in the adoption of new technologies, services, or even providers.

The process to approve services usually involves:

Step 1: Identify threats and vulnerabilities within scope.

Step 2: Identify configurations constraints and mitigation controls from security plan (this Technical Design Document).

Step 3: Individually assess the risks to cloud native services (like Google BigQuery) against threats and vulnerabilities consistent with constraints and mitigation controls in place.

Step 4: Certify a specified application architecture pattern for cloud deployment that consists of one or more cloud native services with respect to its residual risk.

## 

## 6.2 Requirements

From a risk management perspective, a principle requirement for American Express is to establish the most restrictive configuration mode following the Google Cloud Platform (information system) best practices that enables the desired operational business outcomes. This means that the GCP environment should be locked down in the most secure manner possible, following Financial Services best practices, while still enabling their teams to perform their job functions. The following are specific high level requirements on the environment:

### 6.2.1 Encryption in transit

All data being transmitted within the GCP environment must be encrypted end to end. Specifically, leverage application layer encryption such as Transport Layer Security (TLS) when possible for all applications and services. VM to VM communication should also be encrypted as traffic resides within the VLAN subnet.

### 6.2.2 Encryption at rest

All data stored within all resources and services must be encrypted at rest in all environments. Within production environments, the ability to support Customer Managed Keys should be available, including the ability to use Hardware based encryption solutions (HSM).

### 6.2.3 Organization security

Enforce a set of organization policies which apply to all the projects throughout the organization. Support the ability to create exceptions that are managed by a dedicated security team.

### 6.2.4 Data security

Data managed within projects need to have a predefined data ingress and egress security checkpoint controls (e.g: VPC-SC, networking isolation, etc.).

### 6.2.5 Security events

Leverage cloud native security solutions while still leveraging standardized set of 3rd party security services. Specifically, export (feed) current centralized on-premise SIEM systems with security events.

## 6.3 Design options

Google Cloud Platform (information system) best practices recommend the enablement of a number of native security solutions to help identify, protect, detect, respond and finally recover from security events.

### 6.3.1 Identity and Access Management

Cloud IAM is the central information system security control used by GCP to ensure that users have appropriate access to authorized services or resources in a manner that is consistent with an enterprise’s access policies. Please refer to Section 1. Identity and Access Management for more details.

### 6.3.2 Organization Policies

Organization policies apply constraints at the organization, folder, and project level with the ability to include, deny, specify, and control permission. While Cloud Identity and Access Management focuses on the “who”, organization policies focus on the “what”, defining a pattern of desired behaviors and constraints for resources within the organization.

When determining constraints, keep in mind that organization policies are by default, inherited from the parent (whether an organization node or a folder). Also, keep in mind that the organization policies can be merged and extended from the parent policy, or completely overwritten.

Additional details pertaining to the benefits and use cases of organization policies can be found on the [Google Cloud Platform Organization Policies documentation site](https://cloud.google.com/resource-manager/docs/organization-policy/overview).

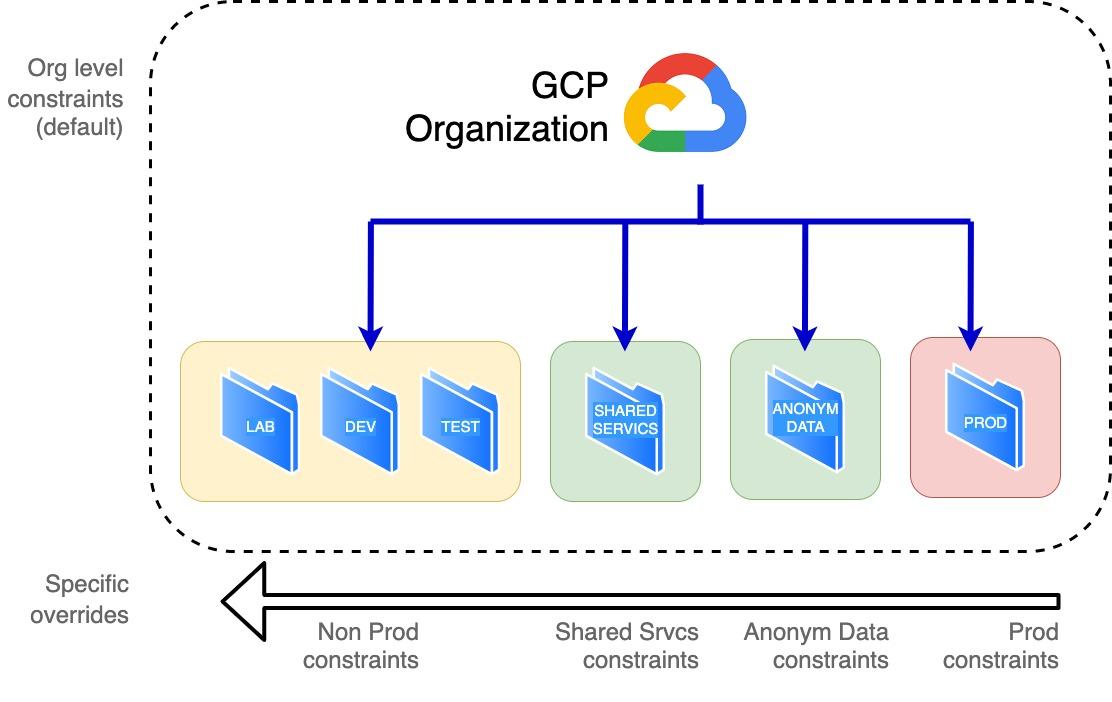
Keep in mind that new constraints may be available through the time, check all current available constraints at [Resource Manager Organization Policy Constraints](https://cloud.google.com/resource-manager/docs/organization-policy/org-policy-constraints)

#### 6.3.2.1 Resources hierarchy

As previously described, Organization Policies can be applied at various levels across the organization. It is a best practice to apply default policy constraints at the organization node (inherited across all underlying resources) and customize policies in the specific levels as required.

**There are 4 major levels of constraints customization at the Environment level:**

* Non-production folders
* Shared services folders
* Anonymized Data folders
* Non-production folders



Generally speaking, Amex will be enforcing constraints at the Organizational level to enforce a uniform policy across the organization. These Organization Policies can be overwritten at a deeper level (e.g: sub-folder or projects) if it is required to enforce or relax constraints for a certain project/group of projects.

#### 6.3.2.2 Constraints

A constraint is a particular type of restriction against a Google Cloud service or a list of Google Cloud services. Think of the constraint as a blueprint that defines what behaviors are controlled. This blueprint is then applied to a resource hierarchy node as an organization policy, which implements the rules defined in the constraint. The Google Cloud service mapped to that constraint and associated with that resource hierarchy node will then enforce the restrictions configured within the organization policy.

A constraint has a type, either **list** or **boolean**. List constraints evaluate the constraint with a list of allowed or denied values that you provide, such as a allowlist of IP addresses that can connect to a virtual machine. Boolean constraints are either enforced or not enforced for a given resource, and govern a specific behavior, such as whether external service accounts can be created.

The below list shows the available recommended constraints:

| **Constraint** | **Recomm** | **Effective** | **Description** |
| --- | --- | --- | --- |
| gcp.resourceLocations | Org | Allow <list> | Resource Location Restriction |
| serviceuser.services | - | Allow all | Restrict allowed Google Cloud APIs and services  **Note:** this policy does not allow all the available services |
| appengine.disableCodeDownload | Org | True | GAE - Disable Source Code Download |
| cloudfunctions.allowedIngressSettings | - | Allow all | GCF - Allowed ingress settings |
| cloudfunctions.allowedVpcConnectorEgressSettings | - | Allow all | GCF - Allowed VPC Connector egress settings |
| cloudfunctions.requireVPCConnector | Org | True | GCF - Require VPC Connector |
| sql.restrictPublicIp | Org | True | Cloud SQL - Restrict Public IP access |
| compute.disableGuestAttributesAccess | - | False | GCE - Disable Guest Attributes of Compute Engine metadata |
| compute.disableInternetNetworkEndpointGroup | Org | True | GCE - Disable Internet Network Endpoint Groups |
| compute.disableNestedVirtualization | Org | True | GCE - Disable VM nested virtualization |
| compute.disableSerialPortAccess | Prod | True | GCE - Disable VM serial port access |
| compute.disableSerialPortLogging | - | False | GCE - Disable VM serial port logging to Stackdriver |
| compute.requireOsLogin | - | False | GCE - Enables OS Login to VMs on all newly created Projects. This constraint prevents metadata updates that disable OS Login at the project or instance level |
| compute.requireShieldedVm | Prod | True | GCE - Requires that all new Compute Engine VM instances use Shielded disk images with Secure Boot, vTPM, and Integrity Monitoring options enabled |
| compute.restrictAuthenticatedGoogleConnection | - | Allow All | GCE - VMs that are allowed to use Authenticated Google Connection |
| compute.restrictCloudNATUsage | - | Allow All | GCE - Restrict Cloud NAT usage |
| compute.restrictDedicatedInterconnectUsage | - | Allow All | GCE - Restrict Dedicated Interconnect usage in subnetworks |
| compute.restrictDirectGoogleAccess | - | Allow All | GCE - Restrict Compute Engine VMs that are allowed to use Direct Google Access |
| compute.restrictLoadBalancerCreationForTypes | Org | Allow in:INTERNAL | GCE - Restrict Load Balancer Creation Based on Load Balancer Types |
| compute.restrictNonConfidentialComputing | - | Allow All | GCE - Restrict Non-Confidential Computing |
| compute.restrictPartnerInterconnectUsage | - | Allow All | GCE - Restrict Partner Interconnect usage |
| compute.restrictProtocolForwardingCreationForTypes | Org | Allow INTERNAL | GCE - Restrict Protocol Forwarding Based on type of IP Address (internal/external) |
| compute.restrictSharedVpcHostProjects | - | Allow All | GCE - Restrict Shared VPC Host Projects |
| compute.restrictSharedVpcSubnetworks | - | Allow All | GCE - Restrict Shared VPC Subnetworks |
| compute.restrictVpcPeering | - | Allow All | GCE - Restrict VPC peering usage |
| compute.restrictVpnPeerIPs | - | Allow All | GCE - Restrict VPN Peer IPs |
| compute.skipDefaultNetworkCreation | Org | True | GCE - Skip default network creation |
| compute.storageResourceUseRestrictions | - | Allow All | GCE - Compute Storage resource use restrictions (Compute Engine disks, images, and snapshots) |
| compute.trustedImageProjects | Org | Allow project\_id | GCE - Projects that can be used for image storage and disk instantiation for Compute Engine |
| compute.vmCanIpForward | Org | Deny All | GCE - Restrict VM IP Forwarding |
| compute.vmExternalIpAccess | Org | Deny All | GCE - Restrict external IPs for VM instances |
| compute.restrictXpnProjectLienRemoval | - | False | RM - Restricts users that can remove a Shared VPC project lien without organization-level permission |
| iam.allowServiceAccountCredentialLifetimeExtension | - | Deny All | IAM - Allow extending lifetime of OAuth 2.0 access tokens to up to 12 hours |
| iam.allowedPolicyMemberDomains | Org | Allow aexp.com | IAM - Domain users that can be added to Cloud IAM policies |
| iam.disableServiceAccountCreation | - | Allow All | IAM - Disable service account creation |
| iam.disableServiceAccountKeyCreation | Org | Deny All | IAM - Disable service account key creation |
| iam.disableServiceAccountKeyUpload | Org\* | Deny All | IAM - Disable uploading public key to Service Accounts |
| iam.disableWorkloadIdentityClusterCreation | - | False | IAM - Disable Workload Identity Cluster Creation |
| iam.automaticIamGrantsForDefaultServiceAccounts | Org | False | IAM - Disable Automatic IAM Grants for Default Service Accounts |
| gcp.detailedAuditLoggingMode | - | True | GCS - Detailed Audit Logging Mode (both, request and response are included in Cloud Audit Logs) |
| storage.retentionPolicySeconds | - | Allow all | GCS - Retention policy duration in seconds |
| storage.uniformBucketLevelAccess | Org | True | GCS - Enforce uniform bucket-level access |
| storage.publicAccessPrevention | Org | True | GCS - Restrict for all buckets and objects public access |

\*Service Account Keys may be required for external trusted sources connectivity to GCP, for that, it’s required to disable the policy manually, create the key and re-enable.

#### 6.3.2.3 Violations

A violation is when a Google Cloud service acts or is in a state that is counter to the organization policy restriction configuration within the scope of its resource hierarchy. Google Cloud services will enforce constraints to prevent violations, but the application of new organization policies is usually not retroactive. If an organization policy constraint is retroactively enforced, it will be labeled as such on the Organization Policy Constraints page.

If a new organization policy sets a restriction on an action or states that a service is already in, the policy is considered to be in violation, but the service will not stop its original behavior. You will need to address this violation manually. This prevents the risk of a new organization policy completely shutting down your business continuity.

### 

### 6.3.3 VPC service controls

VPC Service Controls improves the ability to mitigate the risk of data exfiltration from Google Cloud services such as Cloud Storage and BigQuery. With VPC Service Controls, you create perimeters that protect the resources and data of services that you explicitly specify.

VPC Service Controls provides an additional layer of security defense for Google Cloud services that is independent of Identity and Access Management (IAM). While IAM enables granular identity-based access control, VPC Service Controls enables broader context-based perimeter security, including controlling data egress across the perimeter.

Note: Any account with access to the perimeter can still access the restricted services. It is absolutely required to use both VPC Service Controls and IAM for defense in depth.

#### 6.3.3.1 Perimeter bridge

While a project can be assigned to only one service perimeter, you may want your project to be able to communicate with projects in another perimeter. You can enable communication to services and share data across service perimeters by creating a perimeter bridge.

A perimeter bridge allows projects in different security perimeters to communicate. Perimeter bridges are bidirectional, allowing projects from each service perimeter equal access within the scope of the bridge. However, the access levels and service restrictions of the project are controlled solely by the service perimeter that the project belongs to. A project can have multiple bridges connecting it to other projects.

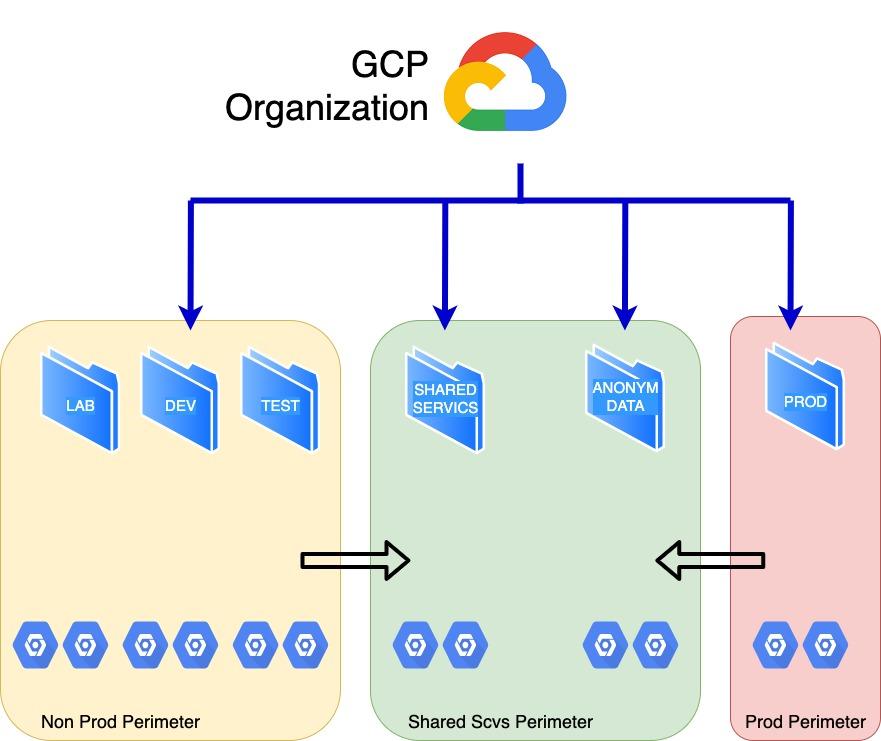
A project from one service perimeter cannot indirectly gain access to projects in other perimeters. For example, assume we have three projects: A, B, and C. Each project belongs to a different service perimeter. A and B share a perimeter bridge. B and C also share a bridge. While data can move between A and B, as well as between B and C, nothing can pass between A and C because the two projects are not directly connected by a perimeter bridge.

A project must belong to a service perimeter before it can be connected to another project using a perimeter bridge.

#### 6.3.3.2 Perimeter definition

AMEX has decided to define 3 perimeters which includes all the projects from **Shared Services,** all projects within **Non Production (E0, E1, E2),** and all projects within **Production (E3)** in the organization.

The non production environments (E0/lab, E1/dev, E2/test) are considered less trusted and should not contain Amex customer or Amex sensitive data. It will be allowed access to de-identified or tokenized data from the Shared Services perimeter that may have been sourced from Production. As such, the Shared Services Perimeter acts as a secure bridge to restrict specific production data sets that a production project explicitly wanted to share with Shared Services. There is no way for non production environments to have any visibility to production data in such a setup.



**Characteristics:**

* Unique, simple perimeter management
* Uniform Access Level and API restriction for all Non Prod and Production projects
* Common allowed access to only data explicitly shared

**Limitations:**

* Perimeter required to be updated when projects are created/destroyed (folder perimeters not supported OOTB, but achievable by automation)
* Uncontrolled SC lateral access within project-to-project (inside the perimeter)
* Not possible to segregate Access Level permissions per project (e.g.: allow partner access to a certain project)

#### 6.3.3.3 Perimeter shared resources

As mentioned, some organizational resources are shared across all environments. This means that any non prod resource in the organization can access or be accessed by the shared resource.

The below list indicate the major shared resources and components:

* Shared Networking (e.g: shared VPC, VPC peering)
* Shared Images
* DNS Peering
* Enablement Team (i.e: CI/CD pipeline)
* Log sinks
* Security Command Center

#### 6.3.3.4 Access context manager (access level)

Access Context Manager allows Google Cloud organization administrators to define fine-grained, attribute based access control for projects and resources in Google Cloud.

Administrators first define an access policy, which is an organization-wide container for access levels and service perimeters. Access levels describe the necessary requirements for requests to be honored. Examples include:

* Device type and operating system
* IP address
* User identity

Service perimeters define sandboxes of resources which can freely exchange data within the perimeter, but are not allowed to export data outside of it. The Access Context Manager isn't responsible for policy enforcement. Its purpose is to describe the desired rules. Policy is configured and enforced across various points, such as VPC Service Controls.

#### 6.3.3.5 Services Restrictiction

It’s recommended to restrict all services which are intended to be used within the perimeter, in order to prevent lateral services from exfiltrating data from the perimeter.

**Note:** Because VPC Service Controls allow the restriction of up to **100 services** within a perimeter, it is recommended to restrict allowed services by using policy enforcement like OPA (open policy agent policies) in order to prevent the usage of services that cannot function within a service perimeter.

The below list of services are being considered to be part of the defined perimeter restrictions:

**Beta:** This product integration is ready for broader testing and use, but is not fully supported for production environments.

| **Service** | **Status** |
| --- | --- |
| artifactregistry.googleapis.com | GA |
| bigquery.googleapis.com | GA |
| binaryauthorization.googleapis.com | GA |
| privateca.googleapis.com | GA |
| bigtable.googleapis.com | GA |
| cloudbuild.googleapis.com | **Beta** |
| dlp.googleapis.com | GA |
| cloudfunctions.googleapis.com | GA |
| cloudkms.googleapis.com | GA |
| logging.googleapis.com | GA |
| monitoring.googleapis.com | GA |
| spanner.googleapis.com | GA |
| sqladmin.googleapis.com | GA |
| storage.googleapis.com | GA |
| compute.googleapis.com | GA |
| containeranalysis.googleapis.com | GA |
| containerregistry.googleapis.com | GA |
| dataflow.googleapis.com | GA |
| dataproc.googleapis.com | GA |
| container.googleapis.com  gkeconnect.googleapis.com  gkehub.googleapis.com | GA |
| oslogin.googleapis.com | GA |
| pubsub.googleapis.com | GA |
| cloudresourcemanager.googleapis.com | **Beta** |
| secretmanager.googleapis.com | GA |
| servicecontrol.googleapis.com | GA |

More information at: [VPC SC Supported Products](https://cloud.google.com/vpc-service-controls/docs/supported-products)

#### 6.3.3.6 Custom perimeter definition

There may be applications with critical data classification which would require custom, very specific access control protection to its data, for example:

* Regulatory requirements
* Share data with external consumers
* Allow 3rd party integrations

More granular, specific perimeter configuration (and restrictions) cannot be achieved within the shared perimeters as these restrictions are applied at perimeter level impacting all its members. A separated, custom perimeter is required to be created and configured with the required restrictions.

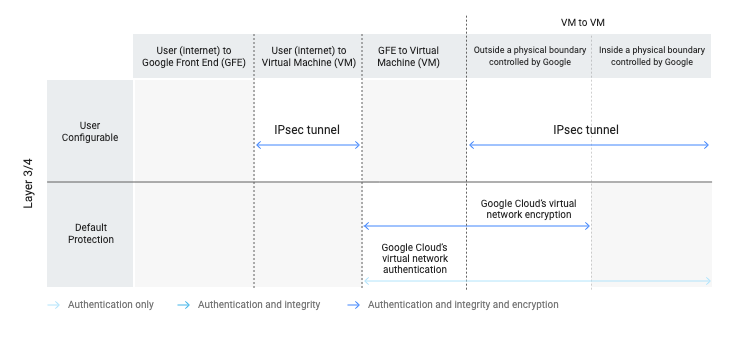
This custom perimeter still needs access to the Shared services perimeters in order to access the organization's shared resources.

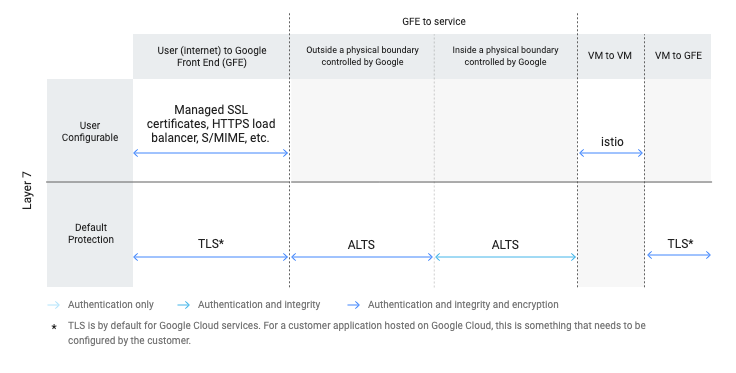
### 

### 

### 6.3.4 Encryption in transit

Google uses various methods of encryption, both default, and user-configurable, for data in transit. The type of encryption used depends on the OSI layer, the type of service, and the physical component of the infrastructure. The following figures below illustrate the optional and default protections Google Cloud has in place for layers 3, 4, and 7.





Google Cloud’s virtual network infrastructure enables encryption when traffic goes outside its physical boundaries. Encryption is performed at the network layer and applies to private IP traffic within the same Virtual Private Cloud (VPC) or across peer-ed VPC networks. Google assumes that any network crossing a physical boundary not controlled by or on behalf of it can be compromised by an active adversary, who can snoop, inject, or alter traffic on the wire. Google ensures the integrity and privacy of communications using encryption when data moves outside physical boundaries they don’t control.

For Google Cloud services, RPCs are protected using ALTS (Application Layer Transport Security) by default. For customer applications hosted on Google Cloud, if traffic is routed via the Google Front End, for example, if they are using the Google Cloud Load Balancer, traffic to the VM is protected using Google Cloud’s virtual network encryption. Google uses the Advanced Encryption Standard (AES) in Galois/Counter Mode (GCM) with a 128-bit key (AES-128-GCM) to implement encryption at the network layer. Each pair of communicating hosts establishes a session key via a control channel protected by ALTS for authenticated and encrypted communications. The session key is used to encrypt all VM-to-VM communication between those hosts, and session keys are rotated periodically.

For additional information into how Google protects data in transit see the [encryption in transit whitepaper](https://cloud.google.com/security/encryption-in-transit/).

### 

### 

### 6.3.4 Encryption at rest and key management

Google Cloud works off the fundamental premise that Google Cloud customers own their data and should control how it is used.

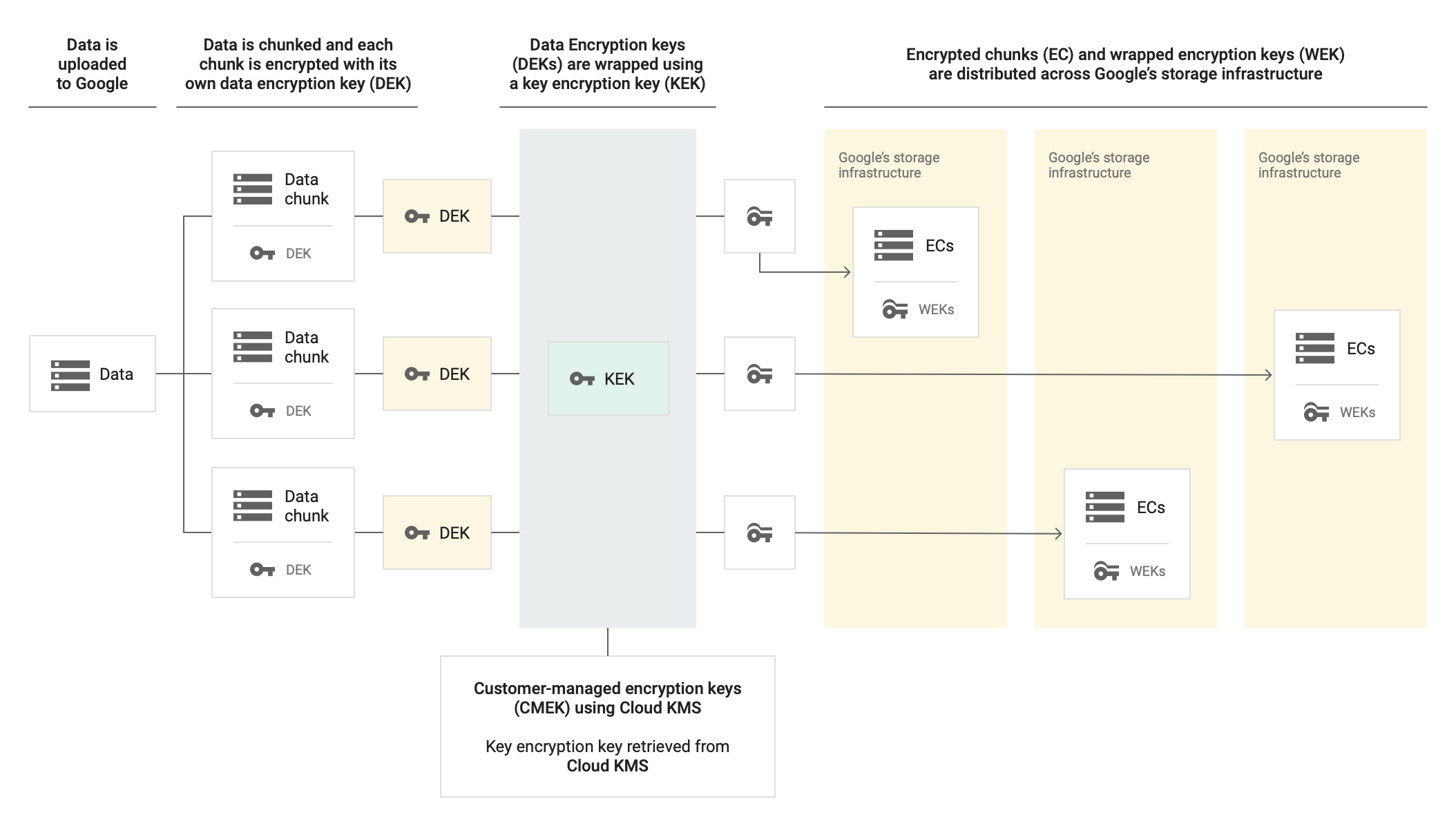
When you store data with Google Cloud, **your data is encrypted at rest by default**. When you use our Cloud Key Management Service (Cloud KMS) platform, you can gain greater control over how your data is encrypted at rest and how your encryption keys are managed.

The Cloud KMS platform lets Google Cloud customers manage cryptographic keys in a central cloud service for either direct use or use by other cloud resources and applications.

Please note that within GCP, Cloud KMS acts as the front line to all key management solutions (default, [CMEK](https://cloud.google.com/kms/docs/cmek), [Cloud HSM](https://cloud.google.com/kms/docs/hsm), [Cloud EKM](https://cloud.google.com/kms/docs/ekm)), allowing for higher level and more demanding encryption controls within one’s environment.

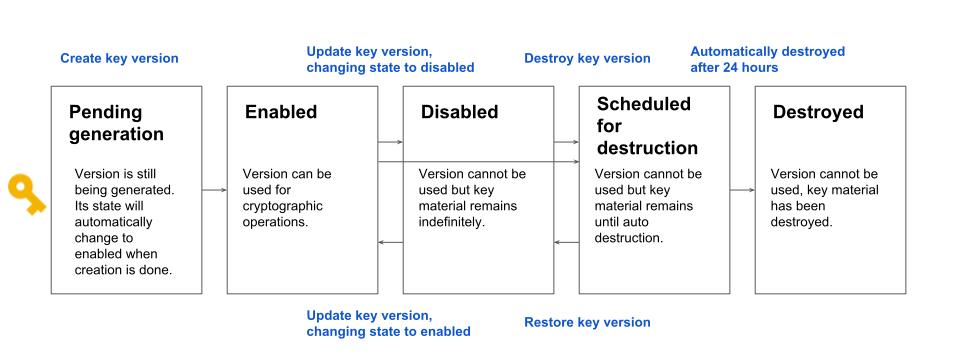


For non production environments, it will be assumed that default encryption will satisfy all business crypto requirements. However, for production (E4) environments, Cloud HSM or Cloud EKM can be leveraged to provide FIPS 140-2 Level 3 security, ensuring additional operational control over your keys as needed. Cloud HSM is operationally simple to support and used by many GCP clients in the Financial Services industry.



**Tip:** For deep understanding on how KMS service works, visit the [Key Management deep dive](https://cloud.google.com/security/key-management-deep-dive) whitepaper.

#### Key States



### 

#### 6.3.4.1 KMS Project & roles segregation

Cloud KMS keys will be hosted within the same project where the sensitive data is stored.

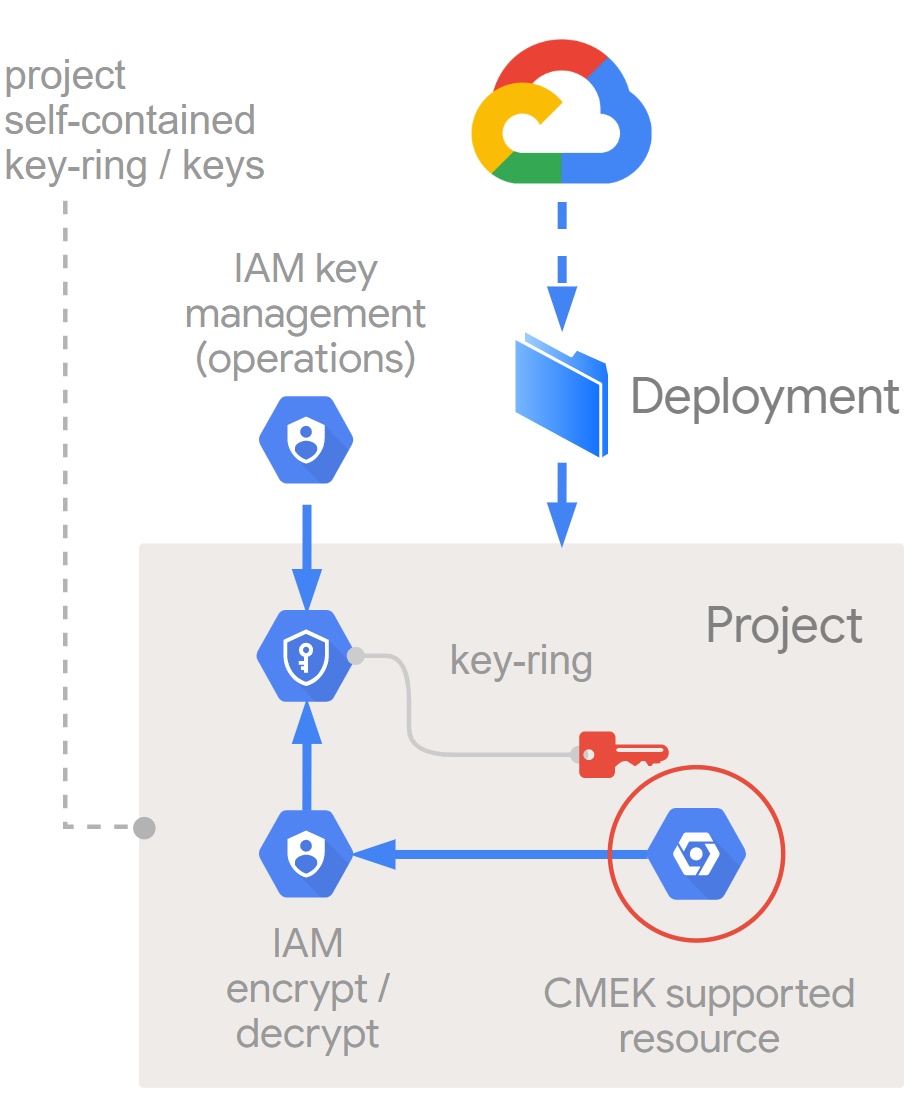
It is of great importance to segregate the appropriate access to these keys.

Note: Any user with owner access on that project is then also able to manage (and perform cryptographic operations with) keys in Cloud KMS in that project.

To run Cloud KMS, the provisioning account (e.g: Terraform Service Account) creates a keyring within the region where data will be hosted, then it creates and attaches the required keys to the storage resources.

Additionally, activities such as key version creation, key import, rotation, and deletion require management access to the keys (functionality required by the operators). These permissions should be segregated from encrypt or decrypt permissions (use key material) to avoid access to the actual encrypted data.

For data generators and consumers, it is required to provide **encrypt** and/or **decrypt** access to the keys applied to the stored data. Data users should not be granted key management permissions, to maintain the correct key access segregation. This allows the platform operators to immediately disable access to data by deleting the related encryption keys (and prevents consumers from being able to recover the key material within delete schedule period)



**Note:** KMS provides Kubernetes secrets encryption, please check secrets section

#### 6.3.4.2 Key rotation

For symmetric encryption, periodically and automatically rotating keys is a recommended security practice. Some industry standards, such as Payment Card Industry Data Security Standard (PCI DSS), require the regular rotation of keys. Automatic key rotation at a defined period, such as every 90 days, increases security with minimal administrative complexity.

**Note**: Cloud Key Management Service does not support automatic rotation of asymmetric keys.

#### 6.3.4.3 Key Deletion

When you submit a request to destroy a key version, destruction occurs after 24 hours unless you cancel the destruction request by restoring the key version.

It is possible to manage access to the key using Identity and Access Management (IAM), for example remove encrypt/decrypt permissions. IAM operations are consistent within seconds.

For destroyed key-rings, VPC service control can remove access to other projects in a single configuration (be careful: in a multi-key-ring project it may lead to remote access to other projects which still require access to active key-rings).

#### 6.3.4.4 Software vs HSM operations

The protection level indicates how cryptographic operations are performed. After you create a key, you cannot change the protection level.

The implementation of KMS key type will mostly depend on the protection level, availability, performance and budget required for the application’s data:

|  | **Software** | **HSM** |
| --- | --- | --- |
| **Cryptographic operations** | Software | In certified HSMs |
| **Material generation** | Random number generator (RNG) built by Google | HSM generated and stored |
| **Protection** | FIPS 140-2 Level 1 | FIPS 140-2 Level 3 |
| **Max block size** | 64 KiB | 16,384 bytes |
| **Availability** | several global locations and across multi-regions | multi or dual regions limitation |
| **Latency** | - | Increased compared to Software |
| **QPS - Symm** | up to 60,000 (per project)\* | up to 30,000 (per project, per region)\* |
| **QPS - Asymm** | up to 60,000 (per project)\* | up to 3,000 (per project, per region)\* |
| **Key upload** | Yes | Services must match HSM keys location exactly |
| **Operation effort** | Console/Gcloud CLI/API | Same as Software |
| **Pricing\*\*** | $0.06 per key +  $0.03 per 10,000 operations | $1.00 per key +  $0.03 per 10,000 operations |

\*It is possible to extend the quota by request

\*\*Average monthly USD at November 2020 for AES-256 and RSA 2048 key types

#### 6.3.4.5 CMEK Integration

| **Service** | **Protected with CMEK** |
| --- | --- |
| **AI Platform Training** | Data on VM disks |
| **AI Platform Notebooks** | Data on VM disks |
| **Artifact Registry** | Data in repositories |
| **BigQuery** | Data in BigQuery |
| **Compute Engine** | Data on VM disks |
| **Google Kubernetes Engine** | Data on VM disks, application-layer Secrets |
| **Dataflow** | Pipeline state data |
| **Dataproc** | Data on VM disks |
| **Cloud Logging** | Logging data |
| **Pub/Sub** | Data associated with topics |
| **Cloud SQL** | Data written to databases |
| **Cloud Storage** | Data in storage buckets |

More details at: [Using KMS in other products](https://cloud.google.com/kms/docs/using-other-products#cmek_compliant)

### 

### 6.3.5 Google Secret Manager

A secret is a project-global object that contains a collection of metadata and secret versions. The metadata can include replication locations, labels, and permissions. The secret versions store the actual secret data, such as an API key or credential.

Google Secret Manager allows you to store, manage, and access secrets as binary blobs or text strings. With the appropriate permissions, you can view the contents of the secret.

Google Secret Manager works well for storing configuration information such as database passwords, API keys, or TLS certificates needed by an application at runtime.

Example when to use Google Secret Manager:

* Non GKE workloads
* Serverless secrets management (e.g: for Cloud Functions)
* IAM level secret access management
* Secret versioning
* Secret access audit

KEY DECISION: AMEX has decided to use Hashicorp Vault as it’s default Secret Manager. Please review the section on [Hashicorp Vault](#_8et2lxc8zh3u) for more information.

## 

### 

### 6.3.6 Google Access Transparency

Access Transparency provides Amex the ability to review logs of actions taken by Google staff when accessing Amex content. Google Access Transparency & Access Approval have the following features:

#### 6.3.6.1 Access approval

Explicitly approve access to your data or configurations on Google Cloud. Access Approval requests, when combined with Access Transparency logs, can be used to audit an end-to-end chain from support ticket to access request to approval, to eventual access.

#### 6.3.6.2 Access justifications

View the reason for each access, including references to specific support tickets where relevant.

#### 6.3.6.3 Resource and method identification

Identify the exact resources accessed by administrators and the methods run.

#### 6.3.6.4 Cloud Logging integration

Integrate seamlessly into your existing Cloud Logging configuration.

#### 6.3.6.5 Accessor location

View the country in which the administrator performing the action was based.

#### 6.3.6.6 Data protection controls

Take advantage of Google’s data-protection controls designed to limit support and engineering’s ability to access your data unless necessary.

#### 6.3.6.7 Near real-time publication

Retrieve logs in near real-time.

### 

### 6.3.7 Cloud Data Loss Prevention (DLP)

Google’s Cloud Data Loss Prevention (DLP) product can be utilized to ensure sensitive data is identified and properly protected. The tool can scan existing datasets and utilize various techniques to obscure sensitive data, including redaction, tokenization, and format-preserving encryption. This ensures that the data is obfuscated before it is stored.

Additionally, the product can look for combinations of pseudo-identifiers that might allow for the identification of an individual even if standard identifiers aren’t included in the dataset. This can be particularly important for consumer privacy protection and compliance policies related to legislation such as GDPR and CCPA.

The Key features for GCP DLP are:

**Data discovery and classification**

With over 120 built-in info types, Cloud DLP gives you the power to scan, discover, classify, and report on data from virtually anywhere. Cloud DLP has native support for scanning and classifying sensitive data in Cloud Storage, BigQuery, and Datastore and a streaming content API to enable support for additional data sources, custom workloads, and applications.

**Automatically mask your data to safely unlock more of the cloud**

Cloud DLP provides tools to classify, mask, tokenize, and transform sensitive elements to help you better manage the data that you collect, store, or use for business or analytics. With support for structured and unstructured data, Cloud DLP can help you preserve the utility of your data for joining, analytics, and AI while protecting the raw sensitive identifiers.

**Measure re-identification risk in structured data**

Enhance your understanding of data privacy risks. Quasi-identifiers are partially identifying elements or combinations of data that may link to a single person or a very small group. Cloud DLP allows you to measure statistical properties such as k-anonymity and l-diversity, expanding your ability to understand and protect data privacy.

When storing PII or other sensitive information, use multiple projects or GCS buckets, assign permissions based on data sensitivity accordingly. At the project level, the best practice is to create two different projects, each with its own data sensitivity. Use a DLP solution to replicate data from a sensitive dataset in Project 1 into the non-sensitive dataset in Project 2. IAM restrictions should be in place to prevent non-authorized users from accessing sensitive data in Project 1. But, a different IAM group for Project 2 should allow users needing access to the redacted set of data. At the GCS bucket level, the best practice is to create two or more different buckets, each with its own data sensitivity. Use DLP to replicate data from sensitive buckets into nonsensitive buckets and use IAM to restrict users from accessing sensitive buckets.

## 6.4 Security Design

Security Design needs to be driven based on the Threat Assessment and Vulnerabilities associated with them. From a holistic perspective, external threats are viewed as the most serious and significant threat to the environment. As such, for Phase 0, no Internet access will be provided to the GCP environment directly. What this means is that no Public IPs will be allowed for any virtual machines or service. Any and all Internet access needs within GCP will be satisfied by either leveraging local (Google Cloud) air-gapped clones of needed repositories or by directing Internet routes to Amex’s On Premises environment. In future Phases, Internet access may be enabled. However, all such access must be processed through a Network Security Appliance before being allowed access to Internal resources. The Network Security Appliance should conform to existing Amex standards. In all phases, care must be taken to ensure that data is never exfiltrated out of the Amex perimeter.

### 6.4.1 Network Security

Fundamental to a secure Cloud perimeter is constraining the Network architecture to only acceptable data traffic flow patterns. Since no direct Internet access will be permitted, Cloud NAT will be explicitly prohibited as a service on GCP. Furthermore, no resource or service will be granted a Public IP address accessible from the Internet. All communications must flow via approved application data flow paths. As such, GCP Firewall rules must be provisioned by default to explicitly block all ingress and egress traffic. Only explicitly required communication flows should be enabled on a case by case basis.

#### 6.4.1.1 Traffic Director

##### **Policy-driven security**

Google [Traffic Director](https://cloud.google.com/traffic-director/) supports authentication and authorization policies. You create these policies so Traffic Director can configure your data plane and enable security capabilities. You don't need to make changes to your application to create or update these policies.

##### **Securing traffic (authentication)**

The authentication policy enables services to (1) assert and validate their identities, and (2) encrypt communication sessions using TLS. In a service mesh, this type of security is handled by the data plane so applications don't need to make special provisions to be secure.

##### **Encryption & supported authentication modes**

When a service calls on another service, the first step in establishing secure communications is to have each service prove its identity to the other service. The degree to which a service needs to prove its identity is based on the TLS mode that you configure.

Traffic Director supports the following modes:

* OPEN: services do not need to prove identity. Communications are unencrypted.
* TLS: the service that is being called must prove its identity to the caller. In other words, the server must prove its identity to the client. Communications are encrypted using TLS.
* MTLS: the service that is being called must prove its identity to the caller and the caller must prove its identity to the called service. In other words, the server must prove its identity to the client and the client must prove its identity to the server. Communications are encrypted using TLS.
* OPEN\_OR\_MTLS (sometimes referred to as "permissive" mode): the service that is being called accepts both OPEN and MTLS. Depending on how the calling service sends its request, communications may or may not be encrypted. TODO: Validate

##### **6.4.1.2 Certificates & certificate authorities**

[Certificate Authority Service](https://cloud.google.com/certificate-authority-service) is a trusted certificate authority (CA) providing the foundation for trust in a distributed system like a service mesh.

* A service that wants to prove its identity to another service presents its certificate to the other service. This certificate is cryptographically signed and issued by a CA which both services trust.
* The service that receives this certificate can verify that the certificate originated from a CA which it trusts.

To learn more about Google CAS, please refer to the [Google blog](https://cloud.google.com/blog/products/identity-security/google-cloud-certificate-authority-service-is-now-ga).

### 

#### 6.4.1.3 Private Google Access

Private Google Access enables on-premises hosts to reach [Google APIs and services](https://developers.google.com/apis-explorer/) using a restricted IP address range rather than a routable, public IP address Private Interconnect connection. The route must have a destination matching one of the VIP ranges and a next-hop being the default internet gateway. Traffic sent to the VIP range stays within Google's network instead of traversing the public Internet because Google does not publish routes to them externally ([Source](https://cloud.google.com/vpc/docs/private-access-options#pga-onprem)). Private Google Access is to be the only path for Amex access into Google APIs and services and no paths via Internet for any APIs should be considered without a security assessment.

### 6.4.2 Security Monitoring

Security monitoring is the fundamental detective control of looking for unexpected events or vulnerabilities as they present themselves within an environment. Not all unexpected events are security incidents. But, unexpected events indicate changes to environments that are outside of norms and should be investigated further.

A significant challenge to security monitoring today is the fact that many diverse tools and solutions exist that often work independently of one another. There is no single source dashboard that allows a security team to see all the signals from all the different solutions in a single place.

Google Cloud Security Command Center (SCC) is a single pane of glass that empowers GCP Security Operation teams to monitor for unexpected events, threats, vulnerabilities, misconfigurations along with 3rd party signal indicators within a single view.

#### 6.4.2.1 Cloud Security Command Center (SCC)

Security Command Center (SCC) is the native Google Cloud product that helps manage and improve cloud security and risk posture. Cloud Security Command Center, with its native security and risk management capabilities, is used by enterprises across the world to protect their environment by gaining visibility into cloud assets, discovering misconfigurations and vulnerabilities in resources, detecting threats targeting Google Cloud assets, and maintaining compliance based on industry standards and benchmarks.

The key features of SCC include:

(1) Asset discovery and inventory: Discover and view your assets in near-real-time across App Engine, BigQuery, Cloud SQL, Cloud Storage, Compute Engine, Cloud Identity, and Access Management, Google Kubernetes Engine, and more. Review historical discovery scans to identify new, modified, or deleted assets.

(2) Threat prevention: Understand the security state of your Google Cloud assets. Uncover common web application vulnerabilities such as cross-site scripting or outdated libraries in your web applications running on App Engine, GKE, and Compute Engine. Quickly resolve misconfigurations by clicking directly on the impacted resource and following the prescribed steps on how to fix it.

(3) Threat detection: Detect threats using logs running in Google Cloud at scale. Detects some of the most common container attacks, including suspicious binary, suspicious library, and reverse shell.

The services offered by SCC include:

1. Security Health Analytics: Offers insights into security vulnerabilities and threats by providing you with managed vulnerability assessment scanning that not only provides visibility and control over your Google Cloud Platform resources but detects vulnerabilities and misconfigurations for your cloud assets and helps reduce your exposure to threats. Some vulnerabilities surfaced by Security Health Analytics include detecting publicly exposed buckets (IAM and Legacy ACLs) and VMs, misconfigured firewalls such as open and overly permissive firewall rules, insecure Cloud IAM configurations, disabled logging, and monitoring, etc. Security Health Analytics also provides monitoring and reporting for industry best practices and compliance monitoring across your Google Cloud assets such as PCI DSS, CIS, NIST, etc.
2. Cloud Anomaly Detection and Event Threat Detection: Focuses intelligence on Cloud Logging and easy integration with other SIEM tools. Monitors Cloud Logging streams such as Cloud Audit streams, VPC flow logs, Cloud DNS, Syslogs to detect threats such as Malware, Cryptomining, Unauthorized Access, Anomalous Grants, Brute-force SSH, and outgoing DDoS attacks.
3. Web Security Scanner: Provides managed scans that help you take preventive actions to reduce your exposure to threats by detecting vulnerabilities in your Google cloud apps such as Cross-Site Scripting, Flash Injection, Mixed Content (HTTP in HTTPS), Outdated-Insecure Libraries, clear text passwords, etc.

SCC offers easy integration with third-party security solutions such as Cloudflare, McAfee, Tenable.io, Palo Alto Networks, etc. Information on [features](https://cloud.google.com/security-command-center) and how to set up and use SCC can be found [here](https://cloud.google.com/security-command-center/docs/quickstart-security-command-center). SCC also supports [exporting](https://cloud.google.com/security-command-center/docs/how-to-export-data) in order to incorporate findings into other security tools.

SCC has two tiers: Standard and Premium. SCC standard tier is free and provides a good asset inventory capability for the security team. The standard tier has the following features:

1. Security Health Analytics: provides managed vulnerability assessment scanning for Google Cloud that can automatically detect the highest severity vulnerabilities and misconfigurations for your Google Cloud assets.
2. Web Security Scanner: supports custom scans of deployed applications with public URLs and IPs that aren't behind a firewall. Scans are manually configured, managed, and executed for all projects.
3. Support for granting users Identity and Access Management (IAM) roles at the organization level.
4. Access to integrated Google Cloud services, including the following:
   1. Cloud Data Loss Prevention discovers, classifies, and protects sensitive data
   2. Google Cloud Armor protects Google Cloud deployments against threats
   3. Anomaly Detection identifies security anomalies for your projects and virtual machine (VM) instances, like potentially leaked credentials and coin mining

SCC premium tier adds the following, in addition to the standard tier features:

1. Event Threat Detection monitors Cloud Logging stream to detect the following threats: Malware, Cryptomining, Brute force SSH, Outgoing DoS, IAM anomalous grant, Data exfiltration, etc.
2. Security Health Analytics provides monitoring for many more industry best practices, and compliance monitoring across your Google Cloud assets, including monitoring and reporting for the following standards: CIS 1.1/1.0, PCI DSS v3.2.1, NIST 800-53, ISO 27001
3. Web Security Scanner provides managed scans that are automatically configured. These scans identify the following security vulnerabilities: Cross-site scripting (XSS), Flash injection, Mixed-content, Cleartext passwords, Usage of insecure JavaScript libraries
4. Support for granting users IAM roles at the organization, folder, and project levels.
5. Continuous Exports, which automatically manage the export of new findings to Pub/Sub.

More details for the standard tier and premium tier, including features and prices, can be found in the following document: <https://cloud.google.com/security-command-center/pricing>.

SCC discovers and inventories Google Cloud assets and resources. It’s a great tool for GCP resource management and other service management. Throughout this Technical Design Document, SCC will be referenced in various sections to address related security issues and best practices.

SCC premium is to be deployed for the initial phase of GCP deployment including tight integration with Prisma Cloud. Specifically, Event Threat Detection is to be enabled to ensure that effective correlation analytics are done within the GCP environment to identify lateral movement threats within the environment.

## 

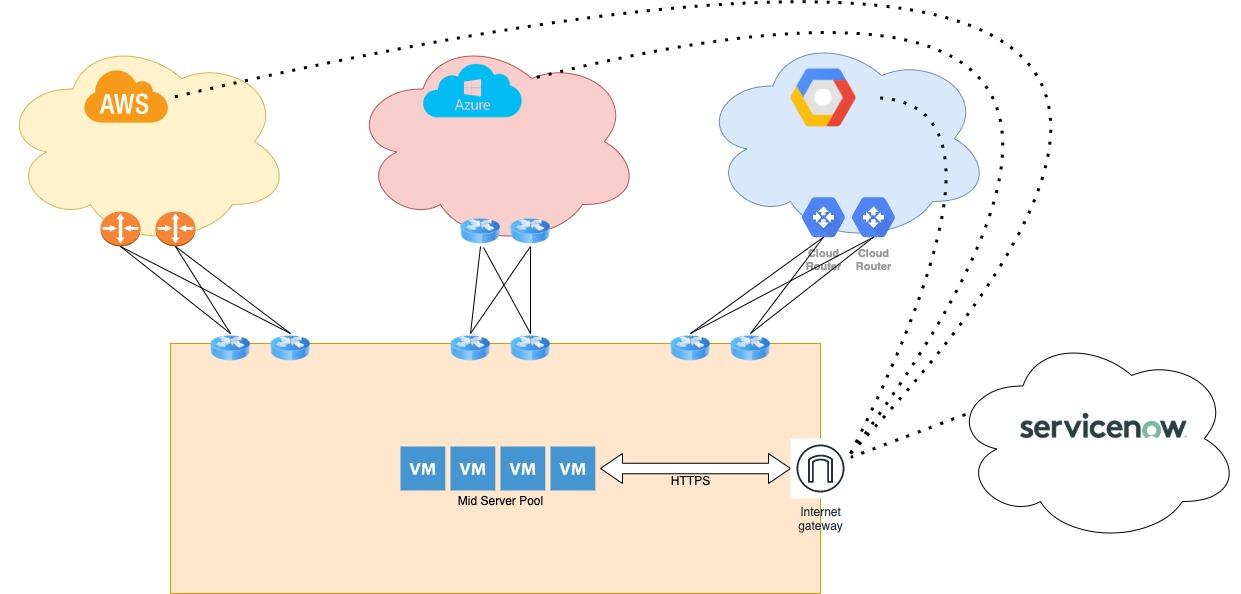
## 

## 6.4.3 Third Party Integrations

A number of 3rd party solutions have been identified. This section covers 3rd party security and governance solutions that have been identified and are in scope.

#### 6.4.3.1 ServiceNow

## 



ServiceNow is a widely used ITSM (IT Service Management) platform deployed within large enterprise environments and is the tool currently used by Amex. There are many architectures that ServiceNow supports. In the case of hybrid multi-cloud (extensive on-premise, and more than 1 Public Cloud), the most common deployment architecture observed is found in the figure above.

ServiceNow establishes management oversight, including change management, of IT assets with the use of Mid Servers. These servers run within a local network and facilitate the discovery of assets (on-premise and Public Cloud) and enable communication and sharing of data within the ServiceNow platform.

NOTE: Full GCP support by ServiceNow was not available before the ‘Paris’ release. Moreover, we are assuming that the IT Operations Management ITOM Module is also included in the deployment model to facilitate the real-time discovery of assets rather than the default once-a-day polling approach.

From an operational perspective, it is simpler to deploy all the ServiceNow Mid Servers in a centrally managed environment. This can be within an ‘on premise’ environment or within one of the Public Clouds. These Windows-based servers require LAN-based access to individual servers if enabling certain ServiceNow application monitoring capabilities. Moreover, these servers will need access to the Internet to access servicenow.com and also need the ability to access all the Public Cloud APIs via HTTPS (either via private Cloud API access or over the Internet).

For GCP specifically, the Mid Server will require a Service Account with organization and project wide “Viewer” rights. A custom “Org Viewer” role should be created to ensure least privileged access containing:

* Iam.roles.get
* Iam.roles.list
* resourcemanager.organizations.get
* resourcemanager.organizations.getIamPolicy
* resourcemanager.projects.get
* resourcemanager.projects.getIamPolicy
* Resourcemanager.projects.list

The Service Account should then have a Service Account key created for the Mid Server. This key has access to very confidential information and should be secured per AMEX’s best practices. ServiceNow will use this Service Account to poll the GCP environment regularly and populate its internal CMDB (Cloud Management DataBase) with current resource metadata information.

#### 6.4.3.2 Prisma Cloud

Palo Alto Networks Prisma Cloud (formerly Twistlock, Redlock) is a cloud security posture management (CSPM) and cloud workload protection platform (CWPP). This provides multi-cloud visibility and threat detection. Please consult the [Prisma Cloud Admin Guide (latest as of August 13th, 2021)](https://docs.paloaltonetworks.com/prisma/prisma-cloud/prisma-cloud-admin) for more details.

Amex currently uses Prisma Cloud (Twistlock) on premises to scan ingress traffic for their GKE clusters. Amex has a deep relationship with Palo Alto Networks and is very familiar with using their products to help secure their environments. Many large enterprises use Prisma Cloud together with the Public Cloud native security solutions to manage and secure their cloud environments. Specifically, Amex should integrate their Prisma Cloud with Google Security Command Center as [outlined by Prism Cloud](https://docs.paloaltonetworks.com/prisma/prisma-cloud/prisma-cloud-admin/configure-external-integrations-on-prisma-cloud/integrate-prisma-cloud-with-google-cloud-security-command-center.html).

##### 6.4.3.2.1 Prisma Cloud Service Exception

Prisma Cloud has the ability to [automatically remediate Alerts](https://docs.paloaltonetworks.com/prisma/prisma-cloud/prisma-cloud-admin/manage-prisma-cloud-alerts/configure-prisma-cloud-to-automatically-remediate-alerts.html) as part of its capabilities. Although this is a very operationally viable solution, it introduces significant risk by providing 3rd parties significant access to one’s environment with highly privileged Service Accounts. From a security perspective, it is strongly recommended that any automations that make changes to your environment reside within the enterprise’s control. Thus, as an Enterprise Policy, no Service Account keys with the ability to make organizational changes should ever be granted outside of the Public Cloud perimeter. It is perfectly fine for external systems to trigger internal automations. Just that the automations must be under the code control of the enterprise and not a 3rd party.

#### 6.4.3.3 Hashicorp Vault

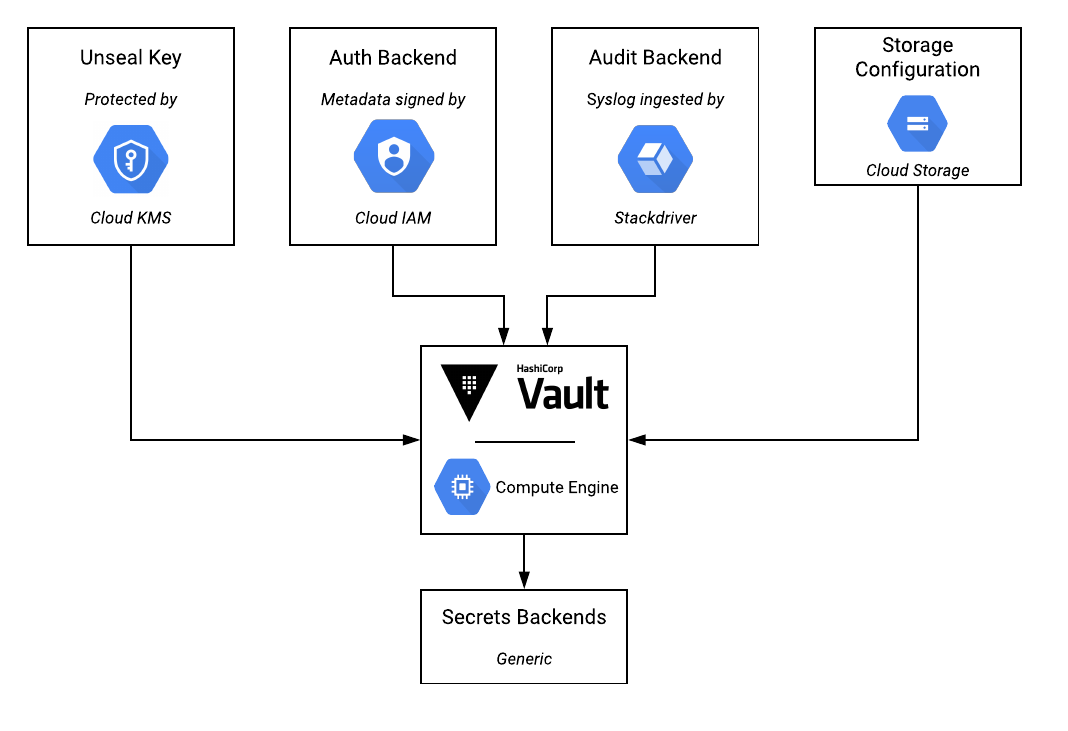
The Google Key Management Service provides the ability to encrypt secrets, but it does not provide an ability to store secrets. As such, an overall approach for the storage and management of application secrets and credentials (eg. service account keys, database credentials, etc) needs to be determined, in alignment with the overall Amex IT Security strategy.

For an overview of how Service Account authentication keys will be managed, refer to the Key management section, above.

Whilst there are ways to create a [secret management solution](https://cloud.google.com/kms/docs/secret-management) with native GCP services, for longer term scalability and ease of management, Amex selected a 3rd party tool for the management of secrets.

[Hashicorp Vault](https://www.vaultproject.io/intro/index.html) is an open source tool for secret management (with a paid Enterprise offering). It provides various capabilities for secret storage and management, and is integrated with various authentication solutions and storage backends, including native GCP services. Amex will use Hashicorp Vault as the central secret management solution for applications on GCP, ([replication](https://www.vaultproject.io/docs/internals/replication.html) can be configured for high availability, in the Vault Enterprise edition).

The following diagram shows an example of how Hashicorp Vault can hypothetically be integrated with GCP services:

[](https://www.lucidchart.com/documents/edit/97855511-bb15-4055-b655-5f5926545873/0?callback=close&name=docs&callback_type=back&v=1287&s=534)

*Figure 6.2 - Hashicorp Vault and GCP Integration*

In this scenario, Vault can be integrated with Google Cloud KMS to provide an “unseal key” that enables encryption of all data stored within Vault. Google Cloud IAM can be used as an “[auth backend](https://www.vaultproject.io/docs/auth/gcp.html)”, to allow applications and services to authenticate to Vault (eg. to access secrets) using native Google Cloud IAM. Google Operations Suite Logging can be used to capture logs from Vault, and Google Cloud Storage can be used as a storage backend.

**Dynamic access management**

Hashicorp Vault should be deployed at Amex to provide the ability to manage application secrets and service account access permissions (eg. definition of scopes and automatic cleanup of service account keys based on leases).

Vault supports various authentication workflows, including a [GCP Auth Method](https://www.vaultproject.io/docs/auth/gcp.html). In terms of accessing secrets post-authentication, an illustrative example is that Vault contains a [Google Cloud Secrets Engine](https://www.vaultproject.io/docs/secrets/gcp/index.html) that can perform the creation of on-demand, “just-in-time” access credentials. Having authenticated with Vault, applications can request Service Account keys, as well as OAuth2 access tokens that provide limited access to specific resources. Vault provides the ability to pre-define the permissions of these service account keys and OAuth2 access tokens. The leasing/expiry of service account keys is configurable, but OAuth2 token lifetimes are currently not configurable (explained [here](https://www.vaultproject.io/docs/secrets/gcp/index.html#access-tokens-vs-service-account-keys)).

## 

## 

## 6.5 Application Security

This section covers Cloud specific best practices around application security on GCP. It is assumed that applications will be written following software development best practices. Key GCP differentiators are described in this section.

### 6.5.1 Application-layer Secret Encryption (Kubernetes)

There are some data related workloads that currently run on OpenShift, and will likely shift to GKE at some point.

By default, GKE encrypts customer content stored at rest, including secrets. GKE handles and manages this default encryption for you without any additional action on your part.

Application-layer Secrets Encryption provides an additional layer of security for sensitive data, such as Secrets, stored in etcd. Using this functionality, you can use a key managed with Cloud KMS to encrypt data at the application layer. This protects against attackers who gain access to an offline copy of etcd.

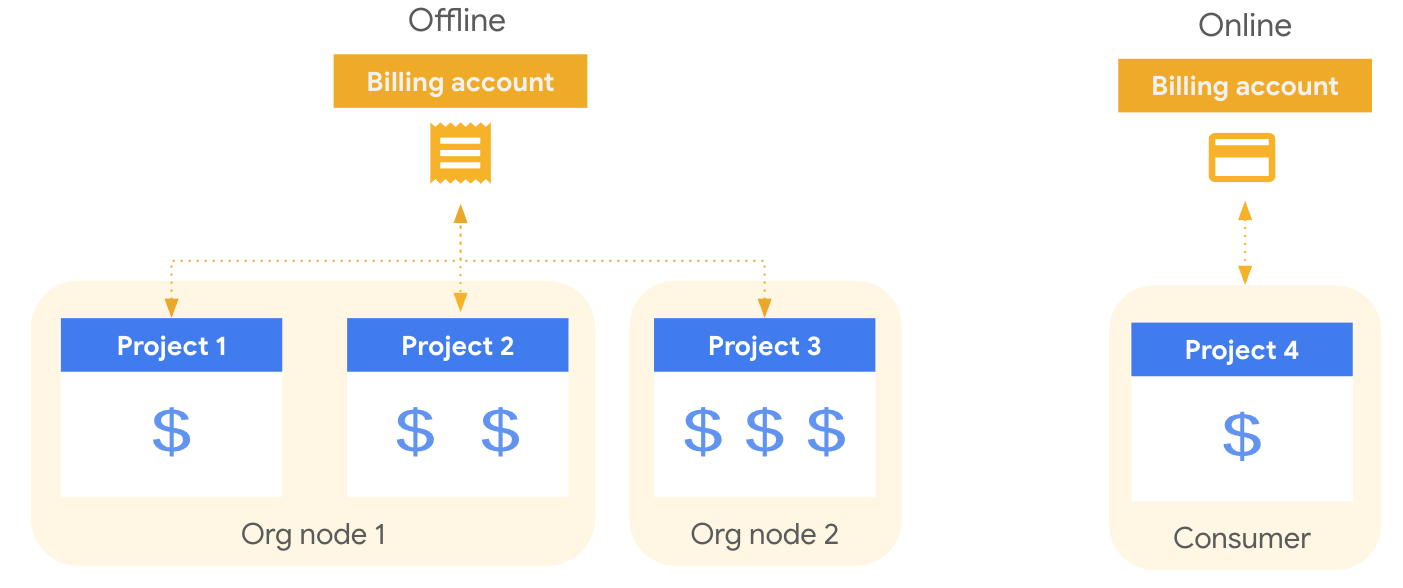
To use Application-layer Secrets Encryption, you must first create a Cloud KMS key and give the GKE service account access to the key. The key must be in the same location as the cluster to decrease latency and to prevent cases where resources depend on services spread across multiple failure domains. Then, you can enable the feature on a new or existing cluster by specifying the key you would like to use.

More information at: [encrypting Kubernetes secrets](https://cloud.google.com/kubernetes-engine/docs/how-to/encrypting-secrets)

# 7. Billing and Quota

Google Compute Platform charges are handled through billing accounts. Billing accounts can be handled three ways for Enterprise customers: credit card charges, bank account debits, or “offline” billing paid via purchase order. At least one billing account is necessary to get started in GCP.

Each project that is created can be associated with a single billing account; a single billing account can be associated with multiple projects.

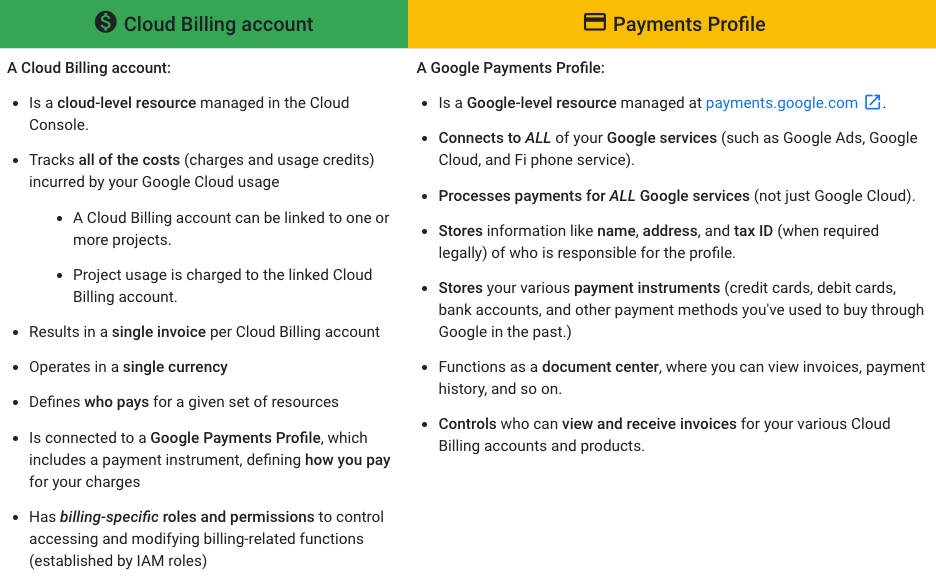


## 7.1 Cloud Billing Account & Payments Profile

### 

### 7.1.1 Overview

A **Cloud Billing account** is set up in Google Cloud and is used to define who pays for a given set of Google Cloud resources. [Access control to a Cloud Billing account](https://cloud.google.com/billing/docs/how-to/billing-access) is established by IAM roles. A Cloud Billing account is connected to a [Google payments profile](https://support.google.com/paymentscenter/topic/9017382?ref_topic=9037778). Your Google payments profile includes a payment instrument to which costs are charged.



### 

### 7.1.2 Billing contacts

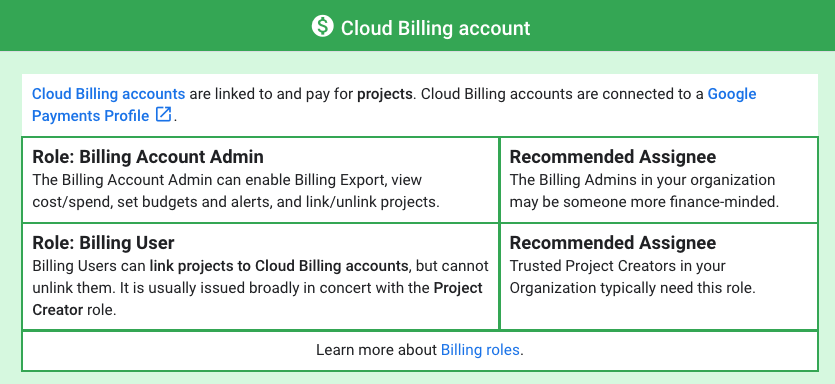
A Cloud Billing account includes one or more contacts that are defined on the [Google Payments profile](https://support.google.com/paymentscenter/answer/9028746) that is connected to the Cloud Billing account. These contacts are people who are designated to receive billing information specific to the payment instrument on file (for example, when a credit card needs to be updated). To access and manage this list of contacts, you can use the [Payments console](https://support.google.com/paymentscenter/answer/7162853) or you can use the [Cloud Console](https://cloud.google.com/billing/docs/how-to/modify-contacts).

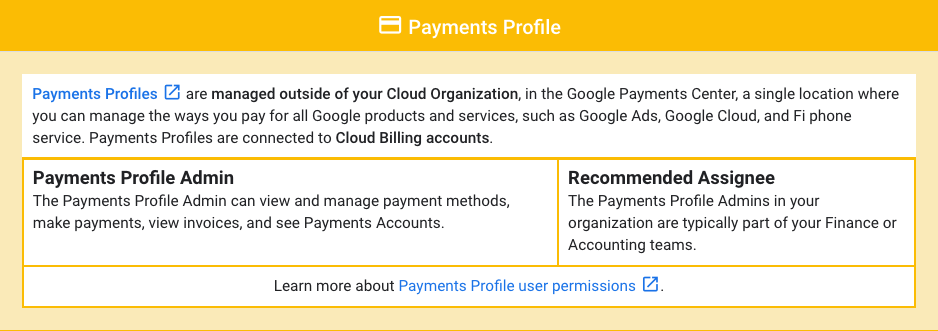
### 7.1.3 Access control

Cloud Billing lets you control which users have administrative and cost viewing permissions for specific resources by setting [Identity and Access Management (IAM)](https://cloud.google.com/iam/docs/overview) policies on the resources.

To grant or limit access to Cloud Billing, you can set an IAM policy at the [organization level](https://cloud.google.com/resource-manager/docs/access-control-org), the Cloud Billing account level, and/or the project level. Google Cloud resources **inherit** the IAM policies of their parent node, which means you can set a policy at the organization level to apply it to all the Cloud Billing accounts, projects, and resources in the organization.

**Key Roles:**





For details on Billing predefined roles refer to [this](https://cloud.google.com/billing/docs/how-to/billing-access#overview-of-cloud-billing-roles-in-cloud-iam) page.

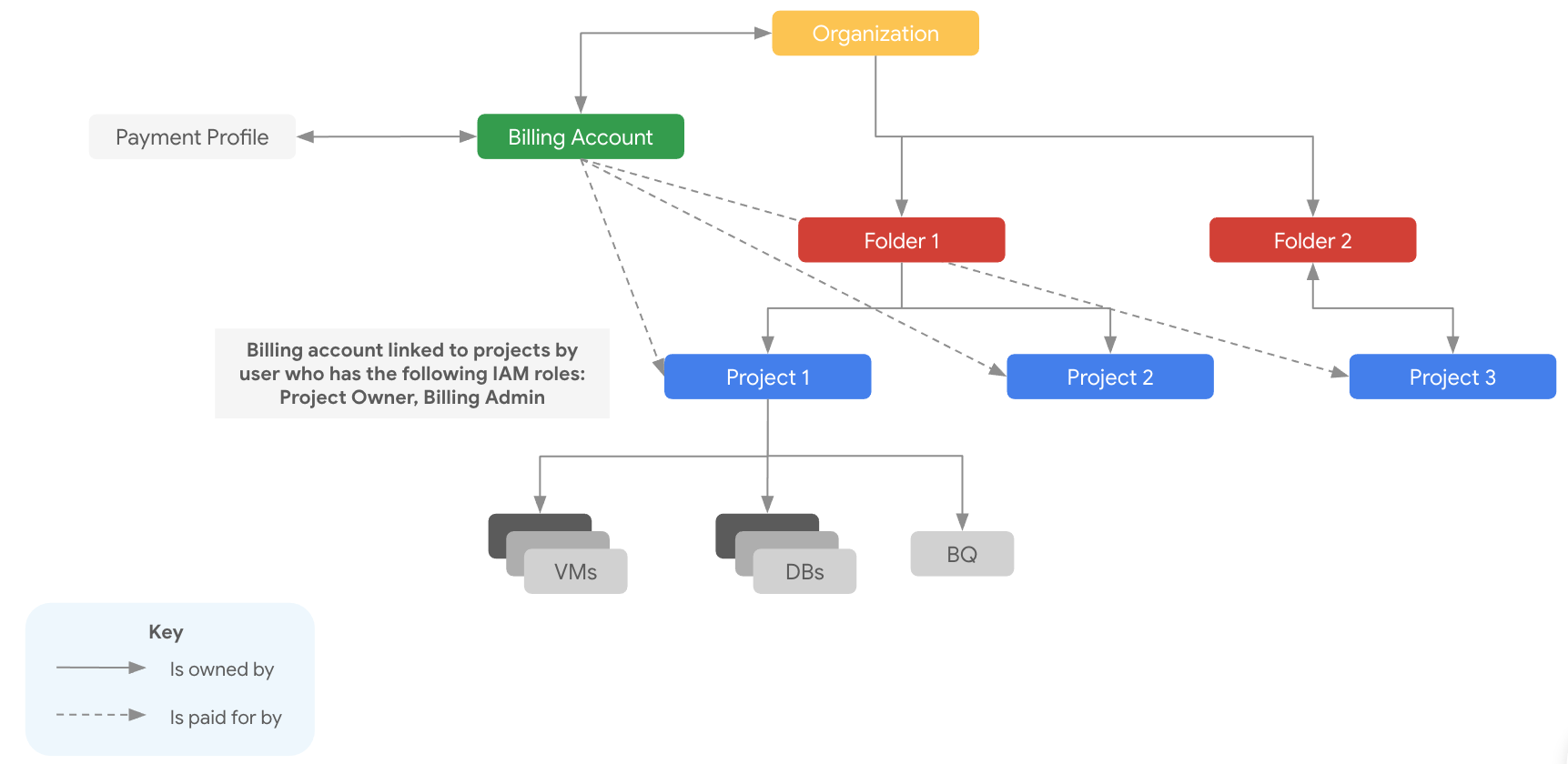
## 7.2 Cloud Billing Resource Hierarchy

Two types of relationships govern the interactions between organizations, Cloud Billing accounts, and projects: ownership and payment linkage.

* **Ownership** refers to IAM permission inheritance.
* **Payment linkages** define which Cloud Billing account pays for a given project.

**Note:** *Ownership* of a Cloud Billing account is limited to a single organization. *Payment linkage* of a project linked to a Cloud Billing account is not limited by organization ownership. It is possible for a Cloud Billing account to pay for projects that belong to an organization that is different from the organization that owns the Cloud Billing account. For more information on managing the Cloud Billing account for your projects, see [Enable, disable, or change billing for a project](https://cloud.google.com/billing/docs/how-to/modify-project).

The following diagram shows the relationship of ownership and payment linkages for a sample organization.



In the diagram, the organization has ownership over Projects 1, 2, and 3, meaning that it is the IAM permissions parent of the three projects.

The Cloud Billing account is linked to Projects 1, 2, and 3, meaning that it pays for costs incurred by the three projects.

**Note:** Although you link Cloud Billing accounts to projects, Cloud Billing accounts are not parents of projects in an IAM sense, and therefore projects don't inherit permissions from the Cloud Billing account they are linked to.

The Cloud Billing account is also linked to a [Google payments profile](https://cloud.google.com/billing/docs/concepts#billing_account), which stores information like name, address, and payment methods.

In this example, any users who are granted IAM billing roles in the organization also have those roles on the Cloud Billing account or the projects.

For more information on granting IAM billing roles, see [Overview of Cloud Billing access control](https://cloud.google.com/billing/docs/how-to/billing-access).

## 7.3 Billing Accounts

AMEX has a single billing account. Each project will be associated with this billing account. Typically, the project creator would associate the billing account to the new projects. This is ideally accomplished by automation, leveraging a GCP Service Account with proper permissions.

For review of costs and administration of the billing account, users can be granted the Billing Account Viewer of Billing Account Administrator role. As of the writing of this section of the TDD the following users have been granted the Billing Account Administrator role:

@TODO Who are the AMEX billing administrators?

| **Name** | **Email** |
| --- | --- |
| AMEX Billing Admin |  |
| AMEX Billing Admin |  |
| AMEX Billing Admin |  |
| AMEX Billing Admin |  |

It is highly recommended that groups be created and assigned the respective Billing roles, rather than keeping the named individuals listed.

Review how billing accounts will be managed?

* What billing account strategy will be used?
  + Single account / multiple accounts? - Single billing account
* How will the billing accounts be charged?
  + P.O., CC, Bank transfer

## 7.4 Cloud Billing Budget

### 

### 7.4.1 Overview

Avoid surprises on your bill by creating Cloud Billing budgets to monitor all of your Google Cloud charges in one place. A budget enables you to track your actual Google Cloud spend against your planned spend. After you've set a budget amount, you set budget alert threshold rules that are used to trigger email notifications. Budget alert emails help you stay informed about how your spend is tracking against your budget. You can also use budgets to automate cost control responses.

With Cloud Billing budgets:

* You can specify the time period for the budget, configuring budgets for monthly, quarterly, yearly, or custom time ranges.
* You can define the scope of the budget. For example, you can scope the budget to apply to the spend in an entire Cloud Billing account, or narrow the scope to one or more projects, and/or one or more services, and/or other budget filters applicable to your Cloud Billing account.
* You can set the budget amount to a total that you specify, or base the budget amount on the previous calendar period's spend.
* You can set threshold rules to trigger email alert notifications. When your costs (actual costs or forecasted costs) exceed a percentage of your budget (based on the rules you set), alert emails are sent to the recipients you specify.
* You can specify the recipients of email alerts in these ways:
  + Using the role-based option (default), you can send email alerts to billing admins and users on the Cloud Billing account.
  + Using Cloud Monitoring, you can specify other people in your organization (for example, project managers) to receive budget alert emails.
* You can also use Pub/Sub for programmatic notifications (for example, to forward your budget messages to other mediums or to automate cost management tasks).

### 7.4.2 Budget alerts

The following sections need to be reviewed with the AMEX team to determine [billing alerting](https://support.google.com/cloud/answer/6293540?hl=en) requirements.

Configuring billing alerts is recommended to draw attention to unexpectedly high-spending projects early. The following is a list of the alerts to be configured for the billing account[s].

For each budget account, create billing alerts:

* Budget name: @TODO
* Billing account name: @TODO
* Budget amount: @TODO

The following associated alerts are to be created.

| **Percentage of budget** | **Amount** |
| --- | --- |
| 50% | TBD |
| 75% | TBD |
| 90% | TBD |
| 100% | TBD |
| 110% | TBD |
| 125% | TBD |
| 150% | TBD |

It is highly recommended to include thresholds over and above 100% of the target. These thresholds do not incur any resource stoppage if met, meaning deployed resources will continue to accrue charges regardless of the percentage of spend consumed.

## 7.5 Billing Labels

As mentioned above in [section 2.2.7](#_i0ele21aylhx) you can attach a label to each resource, then filter the resources based on their labels. Information about labels is forwarded to the billing system, so you can [break down your billed charges](https://cloud.google.com/billing/docs/how-to/bq-examples) by label.

A label is a **key:value pair**

* Can be **attached** to a **project / resource**
* Proper role(s) required to create labels
  + Outside of billing permissions
* Labels based on **team** or **cost center** 
  + May distinguish projects owned by teams
  + Can be used in cost accounting or budgeting
  + For example, team:marketing, team:research

**Note:** When filtering your billing breakdown by label keys, you are not able to select labels applied to a project. You can select other user-created labels that you set up and applied to Google Cloud services.

**Considerations:**

* You should focus on **label consistency** by **applying** them **programmatically** if that's possible. Additionally, if you are **multi-cloud**, then you should try to **ensure** your labeling is **consistent across both**.
* You have the ability to apply **many labels** but Google recommends using **not more than 5 per resource**.
* You should pick what these labels are by asking yourself what aggregations of cost you intend to report on.

## 7.6 Billing Reporting

Get at-a-glance and user-configurable views of your cost history, current cost trends, and forecasted costs with intuitive reports available in the Google Cloud Console. Several different reports are available for your billing data analysis needs.

### 

### 7.6.1 Budget report

Use the [billing report](https://cloud.google.com/billing/docs/how-to/reports) to view and analyze your Google Cloud usage costs using many selectable settings and filters.

Configuring various views of the Cloud Billing report can help you answer questions like these:

* How is my current month's Google Cloud spending trending?
* What Google Cloud project cost the most last month?
* What Google Cloud service (for example, Compute Engine or Cloud Storage) cost me the most?
* What are my forecasted future costs based on historical trends?
* How much am I spending by region?
* What was the cost of resources with label **X**?

Your customized report views are saveable and shareable.

Use the [cost table report](https://cloud.google.com/billing/docs/how-to/cost-table) to access and analyze the details of your invoices and statements.

Based on conversations with AMEX Google recommends creating consolidated billing reports which would span across projects and based per application. In order to make relevant reports, AMEX will need to provide different aggregations to allow users to interact at the correct level of granularity. Console reports and custom dashboards are one way to achieve this. This will help application teams view their respective billing dashboards and manage cost.

### 7.6.2 Cost table report

Because your generated invoice and statement PDFs only contain simplified, summarized views of your costs, the cost table report is available to provide invoice or statement cost details, such as the following:

* Includes project-level cost details from your invoices and statements, including your [tax costs broken out by project](https://cloud.google.com/billing/docs/how-to/cost-table#taxes).
* Includes additional details you might need, such as [service IDs](https://cloud.google.com/billing/docs/how-to/cost-table#service-id), [SKU IDs](https://cloud.google.com/billing/docs/how-to/cost-table#sku-id), and [project numbers](https://cloud.google.com/billing/docs/how-to/cost-table#project-number).
* The report view is customizable and downloadable to CSV.

### 7.6.3 Cost breakdown report

Use the [cost breakdown report](https://cloud.google.com/billing/docs/how-to/cost-breakdown) for an at-a-glance [*waterfall*](https://wikipedia.org/wiki/Waterfall_chart) overview of your monthly costs and savings.

This report shows the following summarized view of monthly charges and credits:

* The combined costs of your monthly Google Cloud usage at the on-demand rate, calculated using non-discounted list prices.
* Savings realized on your invoice due to negotiated pricing (if applicable to your Cloud Billing account).
* Savings earned on your invoice with usage-based credits, broken down by credit type (for example, committed use discounts, sustained use discounts, free tier usage).
* Your invoice-level charges such as tax and adjustments (if any) applied for that invoice month.

### 7.6.4 Prices per SKU pricing report

Use the [pricing table report](https://cloud.google.com/billing/docs/how-to/pricing-table) to access SKU prices for Google's cloud services, including Google Cloud, Google Maps Platform, and Google Workspace, as of the date the report is viewed.

This report shows the following pricing information:

* Displays SKU prices specific to the selected Cloud Billing account.
* If your Cloud Billing account has negotiated contract pricing, each SKU displays the list price, your contract price, and your effective discount.
* If a SKU is subject to [tiered pricing](https://cloud.google.com/billing/docs/how-to/pricing-table#tiered-pricing), each pricing tier for a SKU is listed as a separate row.
* All the prices are shown in the currency of the selected billing account.
* The report view is customizable and downloadable to CSV for offline analysis.

### 7.6.5 Committed use discounts analysis reports

Use the [CUD analysis reports](https://cloud.google.com/billing/docs/how-to/cud-analysis) to visualize and understand the effectiveness and financial impact of the [Committed use discounts](https://cloud.google.com/docs/cuds) (CUDs) you have purchased.

The CUD analysis reports help answer questions such as the following:

* How much are my committed use discounts saving me on my bill?
* Am I fully utilizing my existing commitments?
* How much of my eligible usage is covered by commitments?
* Is there an opportunity to save more by increasing my commitments?
* Can I improve my commitment utilization by enabling discount sharing?

### 7.6.6 Exported billing data

You can create your own custom billing reports based on your [exported billing data](https://cloud.google.com/billing/docs/how-to/export-data-bigquery).

* Enable [Cloud Billing export to BigQuery](https://cloud.google.com/billing/docs/how-to/export-data-bigquery) to export your detailed Google Cloud billing data (such as usage, cost estimates, and pricing data) automatically throughout the day to a [BigQuery dataset](https://cloud.google.com/bigquery/docs/datasets-intro) that you specify.
* Access your exported Cloud Billing data for detailed analysis.
* Use tools like [Google Data Studio to visualize your data](https://cloud.google.com/billing/docs/how-to/visualize-data).

## 7.7 Quota

Google Cloud enforces [quotas](https://cloud.google.com/compute/docs/resource-quotas) on resource usage for a variety of reasons. For example, quotas protect the community of GCP users by preventing unforeseen spikes in usage.

Not all projects have the same quotas. As your use of GCP expands over time, your quotas may increase via a quota increase request process. If you expect an upcoming increase in usage that exceeds your existing quotas, you can proactively request quota increases from the [Quotas](https://console.cloud.google.com/compute/quotas) page in the Cloud Platform Console. To view all quota details and request increases, log in to the Cloud Console and select IAM & Admin -> Quota.

Capacity planning and quotas should be reviewed as projects incur natural growth or any time a new application is being added into an existing project. Reasonable quota requests are usually processed within three business days. Large quota requests may take additional time to procure resources.

At the current time, there are no immediate needs for large-scale quota increases @TODO Validate. If this changes in the near future, please consult your Google PSO Deployment Engineer during the Cloud Plan or your GCP account team to discuss quota strategies. For monitoring quotas across projects in the organization or folder, use [Quota Monitoring Solution](https://cloud.google.com/blog/products/gcp/google-cloud-introduces-quota-monitoring-solution).

# 8. Compute and Storage

## 8.1 Google Cloud Storage

Cloud Storage allows world-wide storage and retrieval of any amount of data at any time. You can use Cloud Storage for a range of scenarios including serving website content, storing data for archival and disaster recovery, or distributing large data objects to users via direct download.

### 

### 8.1.1 Buckets

Buckets are the basic containers that hold your data. Everything that you store in Cloud Storage must be contained in a bucket. You can use buckets to organize your data and control access to your data. The name and location of the bucket cannot be changed after creation, though you can [delete and re-create the bucket](https://cloud.google.com/storage/docs/moving-buckets) to achieve a similar result. While there is no limit to the number of buckets you can have in a project or [location](https://cloud.google.com/storage/docs/locations), there are [limits to the rate you can create or delete buckets](https://cloud.google.com/storage/quotas). Google recommends applying [bucket labels](https://cloud.google.com/storage/docs/key-terms#bucket-labels) to organize your resources. ​​@TODO What are different types of data that will be stored in buckets?

### 8.1.2 Objects

Objects are the individual pieces of data that you store in Cloud Storage. There is no limit on the number of objects that you can create in a bucket. Objects are immutable, which means that an uploaded object cannot change throughout its storage lifetime. However, it is possible to replace objects that are stored in Cloud Storage, and doing so happens atomically: until the new upload completes, the old version of the object is served to readers, and after the upload completes the new version of the object is served to readers.

### 8.1.3 Storage Classes

Save costs without sacrificing performance by storing data across different storage classes. You can start with a class that matches your current use, then reconfigure for cost savings.

* [Standard Storage](https://cloud.google.com/storage/docs/storage-classes#standard): Good for “hot” data that’s accessed frequently, including websites, streaming videos, and mobile apps.
* [Nearline Storage](https://cloud.google.com/storage/docs/storage-classes#nearline): Low cost. Good for data that can be stored for at least 30 days, including data backup and long-tail multimedia content.
* [Coldline Storage:](https://cloud.google.com/storage/docs/storage-classes#coldline) Very low cost. Good for data that can be stored for at least 90 days, including disaster recovery.
* [Archive Storage:](https://cloud.google.com/storage/docs/storage-classes#archive) Lowest cost. Good for data that can be stored for at least 365 days, including regulatory archives.

### 8.1.4 Data Retention Lifecycle

There are two features that help managing objects lifecycle:

#### 8.1.4.1 Object Versioning

To support the retrieval of objects that are deleted or replaced, Cloud Storage offers the Object Versioning feature. Once versioning is enabled:

* Version is created upon object **deletion / overwriting.**
* **Good** for protecting data in case of **accidental** object **deletion / overwriting.**
* Objects **cannot be recovered** when the **bucket is deleted.**
* **Enabled** on **bucket level.**

AMEX will be leveraging versioning on all buckets. @TODO Is there any bucket where versioning will not be used?

#### 8.1.4.2 Object Lifecycle Management

Configure your data with [Object Lifecycle Management](https://cloud.google.com/storage/docs/lifecycle) (OLM) to automatically transition to lower-cost storage classes when it meets the criteria you specify, such as when it reaches a certain age or when you’ve stored a newer version of the data. You can assign a lifecycle management configuration to a bucket. The configuration contains a set of rules which apply to current and future objects in the bucket. When an object meets the criteria of one of the rules, Cloud Storage automatically performs a specified action on the object. Here are some example use cases:

* Downgrade the storage class of objects older than 365 days to Coldline Storage.
* Delete objects created before January 1, 2013.
* Keep only the 3 most recent versions of each object in a bucket with versioning enabled.

@TODO Details of Lifecycle management for buckets created by AMEX Foundation workstream

### 8.1.5 Geo-Redundancy

Geo-redundancy ensures maximum availability of your data, even in the event of large-scale disruptions, such as natural disasters. For dual-regions, geo-redundancy is achieved using two specific regions. For multi-regions, geo-redundancy is achieved using any combination of data centers within the specified multi-region, which may include data centers that are not explicitly listed as available regions.

* You specify a [location](https://cloud.google.com/storage/docs/locations#available-locations) for storing your [object data](https://cloud.google.com/storage/docs/key-terms#objects) when you create a bucket.
* You can select from the following location types:
  + A *region* is a specific geographic place, such as São Paulo.
  + A *dual-region* is a specific pair of regions, such as Tokyo and Osaka.
  + A *multi-region* is a large geographic area, such as the United States, that contains two or more geographic places.
* Objects stored in a multi-region or dual-region are [*geo-redundant*](https://cloud.google.com/storage/docs/key-terms#geo-redundant). You can choose how quickly you want your data to be geo-redundant:
  + *Default replication* asynchronously replicates newly written Cloud Storage objects to a separate region. While most objects are replicated within minutes, some larger objects can take longer.
  + [*Turbo replication*](https://cloud.google.com/storage/docs/turbo-replication) asynchronously replicates newly written Cloud Storage objects to a separate region within a target of 15 minutes. Applicable only for dual-region buckets.

@TODO AMEX is going to use dual-region or multi-region?

### 8.1.6 Access Control

When you create a bucket, you should decide whether you want to apply permissions using *uniform* or *fine-grained* access.

* **Uniform (recommended)**: [Uniform bucket-level access](https://cloud.google.com/storage/docs/uniform-bucket-level-access) allows you to use [Identity and Access Management (IAM)](https://cloud.google.com/storage/docs/access-control/iam) alone to manage permissions. IAM applies permissions to all the objects contained inside the bucket or groups of objects with common name prefixes. IAM also allows you to use features that are not available when working with ACLs, such as [IAM Conditions](https://cloud.google.com/iam/docs/conditions-overview) and Cloud Audit Logs.
* **Fine-grained**: The fine-grained option enables you to use IAM and [Access Control Lists (ACLs)](https://cloud.google.com/storage/docs/access-control/lists) together to manage permissions. ACLs are a legacy access control system for Cloud Storage designed for interoperability with Amazon S3. You can specify access and apply permissions at both the bucket level and per individual object.

#### 8.1.6.1 Access Control Overview

| **Permission method** | **Scope** | **Access control** | **Use case** |
| --- | --- | --- | --- |
| ***IAM***  ***permission*** | Project, Bucket | * Grant access to project’s bucket and objects * User must be in IAM | GCP user access |
| ***Access control lists***  ***(Object ACL)*** | Object | * Grant read or write access to users for objects * Can permit users from outside | User access per object |
| ***Signed URLs*** | Object | * Grant time-limited read or write access to an object * Access to anyone you share URL with | Time limited, temporary access |
| ***Signed policy***  ***document*** | Bucket | * Policy control contents that can be uploaded | Session-based uploads,  e.g., web form |

**Key points:**

* There are four types of access control:
  + **Signed policy document** is used to control the contents that can be uploaded.
  + **Signed URLs** grant temporary access to **objects.**
  + **ACLs** are used to manage access on **objects**, using READ/WRITE permissions.
  + **IAM** are used to manage access on levels of **project** and **buckets**, using IAM roles such as Storage Viewer.
* All of these options can be configured together, and are **not mutually inclusive.**

#### 8.1.6.2 Recommendations:

* Bucket level policy:
  + Managing permissions on buckets is **much more manageable**
  + Permissions are managed through a **single system, and not two**
* Object level policy:
  + Consider a bucket with thousands of objects, which is a very common case. Managing access becomes **unmanageable**
  + Leads to a risk of **unintended access**, possibly **outside of the org**
* Enforce **Bucket Policy Only** org policy
* Enable **Domain Restricted Sharing** org policy to disable access from non-whitelisted domains

## 8.2 BigQuery

BigQuery is a fully managed enterprise data warehouse that helps you manage and analyze your data with built-in features like machine learning, geospatial analysis, and business intelligence. BigQuery's serverless architecture lets you use SQL queries to answer your organization's biggest questions with zero infrastructure management. BigQuery's scalable, distributed analysis engine lets you query terabytes in seconds and petabytes in minutes.

BigQuery maximizes flexibility by separating the compute engine that analyzes your data from your storage choices.

### 

### 8.2.1 BigQuery Storage

BigQuery presents data in tables, rows, and columns and provides full support for database transaction semantics ([ACID](https://en.wikipedia.org/wiki/ACID)). BigQuery storage is automatically replicated across multiple locations to provide high availability.

* BigQuery's top-level container of tables and views.
* [Load data into BigQuery](https://cloud.google.com/bigquery/docs/loading-data) using:
  + [Stream data](https://cloud.google.com/bigquery/streaming-data-into-bigquery) with the [Storage Write API](https://cloud.google.com/bigquery/docs/write-api) ([preview](https://cloud.google.com/products#product-launch-stages)).
  + [Batch-load data](https://cloud.google.com/bigquery/docs/batch-loading-data) from local files or Cloud Storage using formats that include: [Avro](https://cloud.google.com/bigquery/docs/loading-data-cloud-storage-avro), [Parquet](https://cloud.google.com/bigquery/docs/loading-data-cloud-storage-parquet), [ORC](https://cloud.google.com/bigquery/docs/loading-data-cloud-storage-orc), [CSV](https://cloud.google.com/bigquery/docs/loading-data-cloud-storage-csv), [JSON](https://cloud.google.com/bigquery/docs/loading-data-cloud-storage-json), [Datastore](https://cloud.google.com/bigquery/docs/loading-data-cloud-datastore), and [Firestore](https://cloud.google.com/bigquery/docs/loading-data-cloud-firestore) formats.
* [BigQuery Data Transfer Service](https://cloud.google.com/bigquery-transfer/docs/introduction) automates data ingestion.
* Similar to dataset-level and table-level, you can also set up expiration at partition-level.

### 8.2.2 BigQuery Analytics

Descriptive and prescriptive analysis uses include business intelligence, ad hoc analysis, geospatial analytics, and machine learning. You can query data stored in BigQuery or run queries on data where it lives using external tables or federated queries including Cloud Storage, Bigtable, Spanner, or Google Sheets stored in Google Drive.

* ANSI-standard SQL queries ([SQL:2011 support](https://www.iso.org/standard/53681.html)) including support for joins, nested and repeated fields, analytic and aggregation functions, scripting, and a variety of spatial functions with geospatial analytics - Geographic Information Systems.
* [Create views](https://cloud.google.com/bigquery/docs/views-intro) to share your analysis.
* Business intelligence tool support including [BI Engine](https://cloud.google.com/bi-engine/docs/introduction) with [Data Studio](https://cloud.google.com/bigquery/docs/visualize-data-studio), [Looker](https://cloud.google.com/bigquery/docs/looker), [Google Sheets](https://cloud.google.com/bigquery/docs/connected-sheets), and 3rd party tools like Tableau and Power BI.
* [BigQuery ML](https://cloud.google.com/bigquery-ml/docs/introduction) provides machine learning modeling and predictive analytics.
* [Query data outside of BigQuery](https://cloud.google.com/bigquery/external-data-sources) with [external tables](https://cloud.google.com/bigquery/docs/external-tables) and [federated queries](https://cloud.google.com/bigquery/docs/federated-queries-intro).

### 

### 8.2.3 BigQuery Analytics

BigQuery provides centralized management of data and compute resources that are secured using [Identity and Access Management (IAM)](https://cloud.google.com/iam/docs), the access model used throughout Google Cloud. [Google Cloud security best practices](https://cloud.google.com/security/best-practices) provide a solid yet flexible approach that can include traditional perimeter security or more complex and granular [defense-in-depth approach](https://cloud.google.com/security/overview/whitepaper#technology_with_security_at_its_core).

* [Intro to data security and governance](https://cloud.google.com/bigquery/docs/data-governance) helps you understand data governance, and what controls you might need to secure BigQuery resources.
* [Jobs](https://cloud.google.com/bigquery/docs/jobs-overview) are actions that BigQuery runs on your behalf to load, export, query, or copy data.
* [Reservations](https://cloud.google.com/bigquery/docs/reservations-intro) let you switch between on-demand pricing and flat-rate pricing.

## 

## 8.3 Dataproc

Dataproc is a managed Spark and Hadoop service that lets you take advantage of open source data tools for batch processing, querying, streaming, and machine learning. Dataproc automation helps you create clusters quickly, manage them easily, and save money by turning clusters off when you don't need them. With less time and money spent on administration, you can focus on your jobs and your data.

### 

### 8.3.1 Key Features

* **Low cost** - Dataproc is [priced](https://cloud.google.com/dataproc/docs/resources/pricing) at only 1 cent per virtual CPU in your cluster per hour, on top of the other Cloud Platform resources you use. In addition to this low price, Dataproc clusters can include [preemptible instances](https://cloud.google.com/preemptible-vms) that have lower compute prices, reducing your costs even further.
* **Super fast** - Without using Dataproc, it can take from five to 30 minutes to create Spark and Hadoop clusters on-premises or through IaaS providers. By comparison, Dataproc clusters are quick to start, scale, and shutdown, with each of these operations taking 90 seconds or less, on average.
* **Integrated** - Dataproc has built-in integration with other Google Cloud Platform services, such as [BigQuery](https://cloud.google.com/bigquery), [Cloud Storage](https://cloud.google.com/storage), [Cloud Bigtable](https://cloud.google.com/bigtable), [Cloud Logging](https://cloud.google.com/logging), and [Cloud Monitoring](https://cloud.google.com/monitoring), so you have more than just a Spark or Hadoop cluster—you have a complete data platform. For example, you can use Dataproc to effortlessly ETL terabytes of raw log data directly into BigQuery for business reporting.
* **Managed** — Use Spark and Hadoop clusters without the assistance of an administrator or special software. You can easily interact with clusters and Spark or Hadoop jobs through the Google Cloud Console, the Cloud SDK, or the Dataproc REST API. When you're done with a cluster, you can simply turn it off, so you don’t spend money on an idle cluster. You won’t need to worry about losing data, because Dataproc is integrated with [Cloud Storage](https://cloud.google.com/storage), [BigQuery](https://cloud.google.com/bigquery), and [Cloud Bigtable](https://cloud.google.com/bigtable).
* **Simple and familiar** — You don’t need to learn new tools or APIs to use Dataproc, making it easy to move existing projects into Dataproc without redevelopment. Spark, Hadoop, Pig, and Hive are frequently updated, so you can be productive faster.

# 9. Incorporating a DevOps culture

The operational context of an organization largely mandates its organizational structure. Despite the desire to adhere to an Agile methodology, when a company like AMEX makes the decision to operate its own data center, there is a life cycle associated with all serving capacity that needs to be managed and maintained by an operations team. Further, the very nature of an Agile philosophy is sometimes at odds with an organization's need to maintain a stable and uninterrupted production environment.

In GCP, hardware is virtualized and controlled as if it were software. This allows AMEX to evaluate a fundamental shift in its organizational model, while still adhering to Agility.

AMEX desires to achieve these long-term objectives:

* Maintain an Agile development environment.
* Operate a stable and protected production environment.
* Evolve operational procedures without causing development disruption.
* Prevent development teams from spending time troubleshooting operations issues and vice versa.
* Minimize operational interruptions resulting from human error.

Establishing a DevOps culture can allow AMEX to achieve these goals.

## 9.1 Recommendations

1. **Form a Site Reliability Engineering team:** Site Reliability Engineers (SREs) are a mix of application SMEs and operations SMEs who are focused on operational reliability through procedural automation. Provide these SREs a mandate to automate every aspect of operations. The SRE team will continue to be responsible for the deployment of all infrastructure, but will do so via repeatable declarative configurations.
2. **Separate your environments:** Google recommends creating at least three environments:
   1. **Dev:** Environments where software engineers have full access to play, test, and do their work. While the SRE team will probably not need regular access to these development environments, the SRE team's automation capabilities can be used in these projects to ease the creation of desired infrastructure.
   2. **Stage (Test):** A single staging environment where the SRE team and software engineers collaborate on implementing and testing automation capabilities.
   3. **Prod:** A single production environment where the SRE team uses scripts and automation developed and successfully tested in the staging environment to affect change.
3. **Fully implement Continuous Integration:** To free the SRE team from having to deal with production issues stemming from minor development errors:
   1. All code paths need associated unit tests.
   2. All checked-in code must pass a peer review, which includes checking for unit tests.
   3. All checked-in code must pass an automated build process that implements the full battery of unit tests.

It is recommended to set up a code repository structure that is capable of containing an entire AMEX enterprise architecture and integrate with the CI/CD process.

1. **Fully implement Continuous Delivery:** Once a successful build has been created, automate its rollout to the staging environment. Despite the ability to automate the process, most organizations elect to have a manual review of the new build in a staging environment and have someone "press the button" to initiate the phases of the production deployment.

### 

## 9.2 CI/CD process for AMEX

The current process will remain as is with a transition to host the testing and deployment applications to GCP for the Data analytics applications.

The CI/CD pipeline detailed recommendations are as per the discussion with Data workstream.

## 9.3 Infrastructure As Code

Infrastructure as code (IaC) is an approach to provisioning and managing infrastructure through the use of code. There are many benefits to implementing an IaC strategy including:

* Code Management - commit, version, trace and collaborate just like source code
* Declarative - specify the desired state of infrastructure
* Auditable - assess desired state vs. current state of infrastructure
* Scale - build reusable infrastructure blocks across an organization

AMEX is using [Terraform](https://www.terraform.io/) to enable their IaC strategy.

[The Cloud Foundation Toolkit](https://cloud.google.com/foundation-toolkit) (CFT) provides a series of ready-made templates for Terraform which reflect Google Cloud best practices. These templates can be used off-the-shelf to quickly build a repeatable enterprise-ready foundation in Google Cloud.

Please see complete list of all terraform modules available [here](https://cloud.google.com/docs/terraform/blueprints/terraform-blueprints). We have used these modules to write code for the AMEX foundation. For data services also, modules are available in CFT.

AMEX intends to host both the Google provider and cloud foundations toolkit modules locally, therefore the expectation is that over time this code base will deviate from the later officially released versions.

Provided to AMEX is a document titled ‘HashiCorp Terraform Standards Guide”. The document serves as a guide for effective development with Terraform across multiple team members and workstreams.

## 9.4 IaC Testing

This section is meant to act as guidance and best practices around testing Terraform modules and configurations, as it follows different patterns and conventions from testing application code. While testing application code primarily involves testing the business logic of applications themselves, fully testing infrastructure code requires deploying real cloud resources to minimize the risk of production failures. There are a few considerations when running Terraform tests:

1. Running a Terraform test creates, modifies, and destroys real infrastructure, so your tests can potentially be very time-consuming and expensive.
2. **You cannot purely unit test an end-to-end architecture**. The best approach is to break up your architecture into modules and test those individually. The benefits of this approach include faster iterative development due to faster test runtime, reduced costs per test, and reduced chances of test failures from factors beyond your control.
3. **Avoid reusing state if possible**. There may be situations where you are testing with configurations that share data with other configurations, but ideally each test should be independent and should not reuse state across tests.

### 9.4.1 Testing Methods

There are multiple methods which can be used to test Terraform. In ascending order of cost, runtime, and depth, they include:

1. **Static Analysis:** Testing syntax/structure of your configuration without deploying any resources, using tools such as compilers, linters, and dry runs. Use terraform validate to do so.
2. **Module Integration Testing**: Individual modules can be tested in isolation to ensure they work correctly. Integration testing for modules involves deploying the module into a test environment and verifying that expected resources were created. There are several testing frameworks that make it easier to write tests, such as:
   1. [Google’s blueprint testing framework](https://pkg.go.dev/github.com/GoogleCloudPlatform/cloud-foundation-toolkit/infra/blueprint-test)
   2. [Terratest](https://terratest.gruntwork.io/)
   3. [kitchen-terraform](https://newcontext-oss.github.io/kitchen-terraform/)
   4. [InSpec](https://github.com/inspec/inspec-gcp)
3. **End-to-End Testing:** By extending the integration testing approach to an entire environment, you can confirm that multiple modules work well together. In this approach you should deploy all modules that make up the architecture are deployed in a fresh test environment, which is ideally as close to possible as your production environment. This is costly but provides the greatest confidence that changes will not break your live environments.

### 9.4.2 Testing best practices

Running tests on Terraform configurations can take a long time, since each test creates, modifies, and destroys real resources. Following these testing best practices can help you optimize your infrastructure testing methods and minimize test failures.

* **Start small.** Your tests should iteratively build on top of each other. Consider running smaller tests first and then working up to more complex tests, using a “fail fast” approach.
* **Randomize project IDs and resource names.** To avoid naming conflicts such as not using a globally unique project ID or having overlapping resource names within a project, namespace your resources. Terraform has a built-in [random provider](https://registry.terraform.io/providers/hashicorp/random/latest/docs) for this.
* **Use a separate environment for testing.** During testing, many resources will be spun up and deleted. Ensure that the environment is isolated from development or production projects to avoid accidental deletions during resource cleanup. The best approach is to have each test spin up a fresh project or folder. On top of this, consider creating service accounts specifically for each test execution to avoid misconfiguration impact.
* **Fully clean up resources after testing**. Testing IaC code means that you’ll be deploying actual resources, so consider implementing a cleanup step to avoid incurring charges. Use terraform destroy to destroy all remote objects managed by a particular configuration.
  + Some testing frameworks have a built-in cleanup step for you. For example, if using Terratest, add defer terraform.Destroy(t, terraformOptions) to your test to do so. If using kitchen-terraform, delete your workspace using terraform kitchen delete <workspace\_name>
  + After running terraform destroy, you should also run additional cleanup to remove any resources which Terraform failed to destroy. This can be accomplished by deleting any projects used for test execution or by using a tool like [cloud-nuke](https://github.com/gruntwork-io/cloud-nuke). **This should never be used in a production environment**.

### 

#### 9.4.2.1 Optimizing testing runtime

To optimize your test execution time, a few approaches can be used:

* **Run tests in parallel.** Some testing frameworks support running multiple Terraform tests simultaneously.
  + For example, if using Terratest, you can do this by adding t.Parallel() after the test function definition.
* **Test in stages.** Break your tests up into independent configurations which can be tested separately. This removes the need to go through all stages when running a test, and will speed up the iterative development cycle.
  + For example, in kitchen-terraform you should split tests into separate suites. Then, execute each suite independently when iterating.
  + Similarly, using Terratest, wrap each stage of your test with stage(t, <stage\_name>, <correspondingTestFunction>) and set environment variables that indicate which tests to run such as SKIP <stage\_name> = TRUE.
  + The [blueprint testing framework](https://pkg.go.dev/github.com/GoogleCloudPlatform/cloud-foundation-toolkit/infra/blueprint-test) has native support for staged execution.

# 10. High Availability and Disaster Recovery

## 10.1 How Google Cloud is designed for resilience

### 10.1.1 Google data centers

Traditional data centers rely on maximizing availability of individual components. In the cloud, scale allows operators like Google to spread services across many components using virtualization technologies and thus exceed traditional component reliability. This means you can shift your reliability architecture mindset away from the myriad details you once worried about on-premises. Rather than worry about the various failure modes of components -- such as cooling and power delivery -- you can plan around Google Cloud products and their stated reliability metrics. These metrics reflect the aggregate outage risk of the entire underlying infrastructure. This frees you to focus much more on application design, deployment, and operations rather than infrastructure management.

Google designs its infrastructure to meet aggressive availability targets based on our extensive experience building and running modern data centers. Google is a world leader in data center design. From power to cooling to networks, each data center technology has its own redundancies and mitigations, including [FMEA](https://wikipedia.org/wiki/Failure_mode_and_effects_analysis) plans. [Google's data centers](https://www.google.com/about/datacenters/) are built in a way that balances these many different risks and presents to customers a consistent expected level of availability for Google Cloud products. Google uses its experience to model the availability of the overall physical and logical system architecture to ensure that the data center design meets expectations. Google's engineers take great lengths operationally to help ensure those expectations are met. Actual measured availability normally exceeds our design targets by a comfortable margin.

By distilling all of these data center risks and mitigations into user-facing products, Google Cloud relieves you from those design and operational responsibilities. Instead, you can focus on the reliability designed into Google Cloud regions and zones.

### 10.1.2 Regions and zones

Google Cloud products are provided across a large number of [regions and zones](https://cloud.google.com/docs/geography-and-regions). Regions are physically independent geographic areas that contain three or more zones. Zones represent groups of physical computing resources within a region that have a high degree of independence from one another in terms of physical and logical infrastructure. They provide high-bandwidth, low-latency network connections to other zones in the same region.

Google Cloud products are divided into zonal resources, regional resources, or multi-regional resources.

**Zonal resources** are hosted within a single zone. A service interruption in that zone can affect all of the resources in that zone. For example, a Compute Engine instance runs in a single, specified zone; if a hardware failure interrupts service in that zone, that Compute Engine instance is unavailable for the duration of the interruption.

**Regional resources** are redundantly deployed across multiple zones within a region. This gives them higher reliability relative to zonal resources.

**Multi-regional resources** are distributed within and across regions. In general, multi-regional resources have higher reliability than regional resources. However, at this level products must optimize availability, performance, and resource efficiency. As a result, it is important to understand the tradeoffs made by each multi-regional product you decide to use.

The two Google cloud regions that have been identified for the foundational landing zone for AMEX are: us-east4 (three zones) and us-central1 (four zones). These two regions are roughly 1000 miles apart from each other.

## 10.2 Design for scale and high availability

This document in the [Google Cloud Architecture Framework](https://cloud.google.com/architecture/framework) provides design principles to architect your services so that they can tolerate failures and scale in response to customer demand. A reliable service continues to respond to customer requests when there's a high demand on the service or when there's a maintenance event. The following reliability design principles and best practices should be part of your system architecture and deployment plan.

### 10.2.1 Create redundancy for higher availability

Systems with high reliability needs must have no single points of failure, and their resources must be replicated across multiple failure domains. A failure domain is a pool of resources that can fail independently, such as a VM instance, zone, or region. When you replicate across failure domains, you get a higher aggregate level of availability than individual instances could achieve. For more information, see [Regions and zones](https://cloud.google.com/compute/docs/regions-zones).

As a specific example of redundancy that might be part of your system architecture, in order to isolate failures in DNS registration to individual zones, use [zonal DNS names](https://cloud.google.com/compute/docs/internal-dns#zonal-dns) for instances on the same network to access each other.

### 10.2.2 Design a multi-zone architecture with failover for high availability

Make your application resilient to zonal failures by architecting it to use pools of resources distributed across multiple zones, with data replication, load balancing and automated failover between zones. Run zonal replicas of every layer of the application stack, and eliminate all cross-zone dependencies in the architecture.

### 10.2.3 Replicate data across regions for disaster recovery

Replicate or archive data to a remote region to enable disaster recovery in the event of a regional outage or data loss. When replication is used, recovery is quicker because storage systems in the remote region already have data that is almost up to date, aside from the possible loss of a small amount of data due to replication delay. When you use periodic archiving instead of continuous replication, disaster recovery involves restoring data from backups or archives in a new region. This procedure usually results in longer service downtime than activating a continuously updated database replica and could involve more data loss due to the time gap between consecutive backup operations. Whichever approach is used, the entire application stack must be redeployed and started up in the new region, and the service will be unavailable while this is happening.

For a detailed discussion of disaster recovery concepts and techniques, see [Architecting disaster recovery for cloud infrastructure outages](https://cloud.google.com/architecture/disaster-recovery).

### 10.2.4 Design a multi-region architecture for resilience to regional outages

If your service needs to run continuously even in the rare case when an entire region fails, design it to use pools of compute resources distributed across different regions. Run regional replicas of every layer of the application stack.

Use data replication across regions and automatic failover when a region goes down. Some Google Cloud services have multi-regional variants, such as [BigQuery](https://cloud.google.com/bigquery). To be resilient against regional failures, use these multi-regional services in your design where possible. For more information on regions and service availability, see [Google Cloud locations](https://cloud.google.com/about/locations).

Make sure that there are no cross-region dependencies so that the breadth of impact of a region-level failure is limited to that region.

Eliminate regional single points of failure, such as a single-region primary database that might cause a global outage when it is unreachable. Note that multi-region architectures often cost more, so consider the business need versus the cost before you adopt this approach.

For further guidance on implementing redundancy across failure domains, see the survey paper [Deployment Archetypes for Cloud Applications (PDF)](https://storage.googleapis.com/pub-tools-public-publication-data/pdf/cd6b7106c4decf552edc20c125dcb587c4cdcba9.pdf).

## 10.3 Product reference

Listed in this section are product references to GCP products and services that are currently known to be of interest to AMEX.

### 10.3.1 Common themes

Many Google Cloud products offer regional or multi-regional configurations. Regional products are resilient to zone outages, and multi-region and global products are resilient to region outages. In general, this means that during an outage, your application experiences minimal disruption. Google achieves these outcomes through a few common architectural approaches, which mirror the architectural guidance above.

* **Redundant deployment**: The application backends and data storage are deployed across multiple zones within a region and multiple regions within a multi-region location.
* **Data replication**: Products use either synchronous or asynchronous replication across the redundant locations.
  + **Synchronous** replication means that when your application makes an API call to create or modify data stored by the product, it receives a successful response only once the product has written the data to multiple locations. Synchronous replication ensures that you do not lose access to any of your data during a Google Cloud infrastructure outage because all of your data is available in one of the available backend locations.  
    Although this technique provides maximum data protection, it can have tradeoffs in terms of latency and performance. Multi-region products using synchronous replication experience this tradeoff most significantly -- typically on the order of 10s or 100s of milliseconds of added latency.
  + **Asynchronous** replication means that when your application makes an API call to create or modify data stored by the product, it receives a successful response once the product has written the data to a single location. Subsequent to your write request, the product replicates your data to additional locations.  
    This technique provides lower latency and higher throughput at the API than synchronous replication, but at the expense of data protection. If the location in which you have written data suffers an outage before replication is complete, you lose access to that data until the location outage is resolved.
* **Handling outages with load balancing**: Google Cloud uses software load balancing to route requests to the appropriate application backends. Compared to other approaches like DNS load balancing, this approach reduces the system response time to an outage. When a Google Cloud location outage occurs, the load balancer quickly detects that the backend deployed in that location has become "unhealthy" and directs all requests to a backend in an alternate location. This enables the product to continue serving your application's requests during a location outage. When the location outage is resolved, the load balancer detects the availability of the product backends in that location, and resumes sending traffic there.

### 10.3.2 Compute Engine

Compute Engine is Google Cloud's infrastructure-as-a-service. It uses Google's worldwide infrastructure to offer virtual machines (and related services) to customers.

Compute Engine instances are zonal resources, so in the event of a zone outage instances are unavailable by default. Compute Engine does offer [managed instance groups](https://cloud.google.com/compute/docs/instance-groups#managed_instance_groups) (MIGs) which can automatically scale up additional VMs from pre-configured instance templates, both within a single zone and across multiple zones within a region. MIGs are ideal for applications that require resilience to zone loss and are stateless, but require [configuration and resource planning](https://cloud.google.com/compute/docs/instance-groups/distributing-instances-with-regional-instance-groups). Multiple regional MIGs can be used to achieve region outage resilience for stateless applications.

Applications that have stateful workloads can still use [stateful MIGs](https://cloud.google.com/compute/docs/instance-groups/stateful-migs) (beta) but extra care needs to be made in capacity planning since they do not scale horizontally. It's important in either scenario to correctly configure and test Compute Engine instance templates and MIGs ahead of time to ensure working failover capabilities to other zones. See the [Architecture Patterns](https://cloud.google.com/architecture/disaster-recovery#architecture_patterns) section above for more information.

#### 10.3.2.1 Networking for Compute Engine

For information about high-availability setups for Interconnect connections, see the following documents:

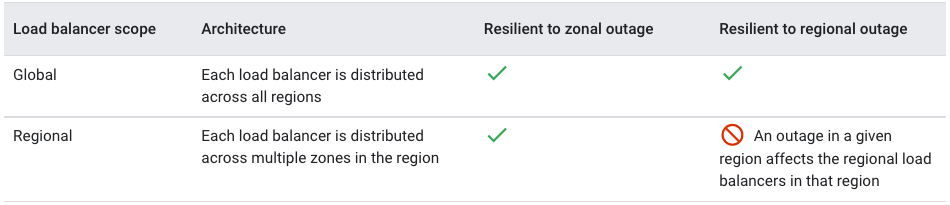
* [99.99% availability for Dedicated Interconnect](https://cloud.google.com/network-connectivity/docs/interconnect/tutorials/dedicated-creating-9999-availability)
* [99.99% availability for Partner Interconnect](https://cloud.google.com/network-connectivity/docs/interconnect/tutorials/partner-creating-9999-availability)

You can provision external IP addresses in [global or regional mode](https://cloud.google.com/compute/docs/ip-addresses/reserve-static-external-ip-address#reserve_new_static), which affects their availability in the case of a regional failure.

#### 10.3.2.2 Cloud Load Balancing resilience

Load balancers are a critical component of most highly available applications. Google Cloud offers both regional and global load balancers. In either case, it is important to understand that the resilience of your overall application depends not just on which load balancer you choose, but also on the redundancy of your backend services.

The following table summarizes load balancer resilience based on the load balancer's distribution or *scope*.



For more information about choosing a load balancer, see the [Cloud Load Balancing documentation](https://cloud.google.com/load-balancing/docs/choosing-load-balancer).

### 10.3.3 Dataproc

Dataproc provides streaming and batch data processing capabilities. Dataproc is architected as a regional control plane that enables users to manage Dataproc clusters. The control plane does not depend on an individual zone in a given region. Therefore, during a zonal outage, you retain access to the Dataproc APIs, including the ability to create new clusters.

Clusters are run in Compute Engine. Because the cluster is a zonal resource, a zonal outage makes the cluster unavailable, or destroys the cluster. Dataproc does not automatically snapshot cluster status, so a zone outage could cause loss of data being processed. Dataproc does not persist user data within the service. Users can configure their pipelines to write results to many data stores; you should consider the architecture of the data store and choose a product that offers the required disaster resilience.

If a zone suffers an outage, you may choose to recreate a new instance of the cluster in another zone, either by selecting a different zone or using the Auto Placement feature in Dataproc to automatically select an available zone. Once the cluster is available, data processing can resume. You can also run a cluster with High Availability mode enabled, reducing the likelihood a partial zone outage will impact a master node and, therefore, the whole cluster.

### 10.3.4 BigQuery

BigQuery is a serverless, highly scalable, and cost-effective cloud data warehouse designed for business agility. BigQuery supports two different availability-related configuration options for user datasets.

#### 10.3.4.1 Single region configuration

In a single region configuration, data is stored redundantly in two zones within a single region. Data written to BigQuery is first written to the primary zone and then asynchronously replicated to a secondary zone. This protects against unavailability of a single zone within the region. Data that was written to the primary zone but that hasn't been replicated to the secondary zone at the time of a zone outage is unavailable until the outage is resolved. In the unlikely case of a zone being destroyed, that data may be permanently lost.

#### 10.3.4.2 Multi-region (US / EU) configuration

Similar to the single region configuration, in the US / EU multi-region configuration, data is stored redundantly in two zones within one region. In addition, BigQuery keeps an additional backup copy of the data in a second region. If the primary region experiences an outage, data is served from the secondary region. Data that hasn't been replicated is unavailable until the primary region is restored.

Google recommends multi-region configuration for US.

### 10.3.5 Google Kubernetes Engine

Google Kubernetes Engine (GKE) offers managed Kubernetes service by streamlining the deployment of containerized applications on Google Cloud. You can choose between regional or zonal cluster topologies.

* When creating a **zonal cluster**, GKE provisions one control plane machine in the chosen zone, as well as worker machines (nodes) within the same zone.
* For **regional clusters**, GKE provisions three control plane machines in three different zones within the chosen region. By default, nodes are also spanned across three zones, though you can choose to create a regional cluster with nodes provisioned only in one zone.
* **Multi-zonal clusters** are similar to zonal clusters as they include one master machine, but additionally offer the ability to span nodes across multiple zones.

**Zonal Outage:** To avoid zonal outages, use regional clusters. The control plane and the nodes are distributed across three different zones within a region. A zone outage does not impact control plane and worker nodes deployed in the other two zones.

**Regional Outage:** Mitigation of a regional outage requires deployment across multiple regions. Although currently not being offered as a built-in product capability, multi-region topology is an approach taken by several GKE customers today, and can be manually implemented. You can create multiple regional clusters to replicate your workloads across multiple regions, and control the traffic to these clusters using [multi-cluster ingress](https://cloud.google.com/kubernetes-engine/docs/how-to/ingress-for-anthos).

### 10.3.6 Cloud Key Management Service

Cloud Key Management Service (Cloud KMS) provides scalable and highly-durable cryptographic key resource management. Cloud KMS stores all of its data and metadata in Cloud Spanner databases which provide high data durability and availability with synchronous replication.

Cloud KMS resources can be created in a single region, multiple regions, or globally.

In the case of zonal outage, Cloud KMS continues to serve requests from another zone in the same or different region without interruption. Because data is replicated synchronously, there is no data loss or corruption. When the zone outage is resolved, full redundancy is restored.

In the case of a regional outage, regional resources in that region are offline until the region becomes available again. Note that even within a region, at least 3 replicas are maintained in separate zones. When higher availability is required, resources should be stored in a multi-region or global configuration. Multi-region and global configurations are designed to stay available through a regional outage by geo-redundantly storing and serving data in more than one region.

### 10.3.7 Cloud Storage

Cloud Storage provides globally unified, scalable, and highly durable object storage. Cloud Storage buckets can be created in a single region, dual regions, or multi-regions within a continent.

If a zone experiences an outage, data in the unavailable zone is automatically and transparently served from elsewhere in the region. Data and metadata are stored redundantly across zones, starting with the initial write. No writes are lost upon a zone becoming unavailable.

In the case of a regional outage, regional buckets in that region are offline until the region becomes available again.

When higher availability is required, you should consider storing data in a dual-region or multi-region configuration. Cloud Storage uses Cloud Load Balancing to serve dual-region and multi-region buckets from different regions. In the case of a regional outage, serving is not interrupted.

Cloud Storage dual-region and multi-region configurations replicate written data synchronously to another zone within the same region, and asynchronously to another region or regions. See [Geo-redundancy](https://cloud.google.com/storage/docs/key-terms#geo-redundant) in the Cloud Storage documentation.

During a regional outage, data that was recently written to the affected region may not have been replicated to other regions. As a result, that data may not be accessible during the outage, and could be lost in the case of physical destruction of the data in the affected region.

Google recommends multi-region configuration along with versioning enabled for critical workloads.

### 10.3.8 Pub/Sub

Pub/Sub is a messaging service for application integration and stream analytics. Pub/Sub topics are global, meaning that they are visible and accessible from any Google Cloud location. However, any given message is stored in a single Google Cloud region, closest to the publisher and allowed by the resource location policy. Thus, a topic may have messages stored in different regions throughout Google Cloud. The Pub/Sub [message storage policy](https://cloud.google.com/pubsub/docs/resource-location-restriction) can restrict the regions in which messages are stored.

**Zonal outage**: When a Pub/Sub message is published, it is synchronously written to storage in at least two zones within the region. Therefore, if a single zone becomes unavailable, there is no customer-visible impact.

**Regional outage**: During a region outage, messages stored within the affected region are inaccessible. Administrative operations, such as creation and deletion of topics and subscriptions, are multi-regional and resilient to an outage of any single Google Cloud region. Publishing operations are also resilient to region outages, provided that:

* at least one region allowed by the message storage policy is available (by default, Pub/Sub does not restrict message storage location), and
* your application uses the global endpoint (pubsub.googleapis.com) or multiple regional endpoints, and
* the publishing client is not within the affected region.

If your application relies on message ordering, review the [detailed recommendations](https://medium.com/google-cloud/google-cloud-pub-sub-ordered-delivery-1e4181f60bc8) from the Pub/Sub team. Message ordering guarantees are provided on a per-region basis, and can become disrupted if you use a global endpoint.

### 10.3.9 Cloud Logging

Cloud Logging consists of two main parts: the Logs Router and Cloud Logging storage.

The Logs Router handles streaming log events and directs the logs to Cloud Storage, Pub/Sub, BigQuery, or Cloud Logging storage.

Cloud Logging storage is a service for storing, querying, and managing compliance for logs. It supports many users and workflows including development, compliance, troubleshooting, and proactive alerting.

**Logs Router & incoming logs**: During a zonal outage, the Cloud Logging API routes logs to other zones in the region. Normally, logs being routed by the Logs Router to Cloud Logging, BigQuery, or Pub/Sub are written to their end destination as soon as possible, while logs sent to Cloud Storage are buffered and written in batches hourly.

**Log Entries:** In the event of a zonal or regional outage, log entries that have been buffered in the affected zone or region and not written to the export destination become inaccessible. Logs-based metrics are also calculated in the Logs Router and subject to the same constraints. Once delivered to the selected log export location, logs are replicated according to the destination service. Logs that are exported to Cloud Logging storage are synchronously replicated across two zones in a region. For the replication behavior of other destination types, see the relevant section in this article. Note that logs exported to Cloud Storage are batched and written every hour. Therefore we recommend using Cloud Logging storage, BigQuery, or Pub/Sub to minimize the amount of data impacted by an outage.

**Log Metadata:** Metadata such as sink and exclusion configuration is stored globally but cached regionally so in the event of an outage, the regional Log Router instances would operate. Single region outages have no impact outside of the region.

## 10.4 Create reliable operational process and tools

### 10.4.1 Automate build, test, and deployment

Eliminate manual effort from your release process with the use of [continuous integration](https://cloud.google.com/architecture/devops/devops-tech-continuous-integration) and [continuous delivery](https://cloud.google.com/architecture/devops/devops-tech-continuous-delivery) (CI/CD) pipelines. Perform automated integration testing and deployment.

### 10.4.2 Test failure recovery

Regularly test your operational procedures to recover from failures in your service. Without regular tests, your procedures might not work when you need them if there's a real failure. Items to test periodically include regional failover, how to roll back a release, and how to restore data from backups.

### 10.4.3 Conduct disaster recovery tests

Like with failure recovery tests, don't wait for a disaster to strike. Periodically test and verify your disaster recovery procedures and processes.

You might create a system architecture to provide high availability (HA). This architecture doesn't entirely overlap with disaster recovery (DR), but it's often necessary to take HA into account when you think about recovery time objective (RTO) and recovery point objective (RPO) values.

HA helps you to meet or exceed an agreed level of operational performance, such as uptime. When you run production workloads on Google Cloud, you might deploy a passive or active standby instance in a second region. With this architecture, the application continues to provide service from the unaffected region if there's a disaster in the primary region. For more information, see [Architecting disaster recovery for cloud outages](https://cloud.google.com/architecture/disaster-recovery).

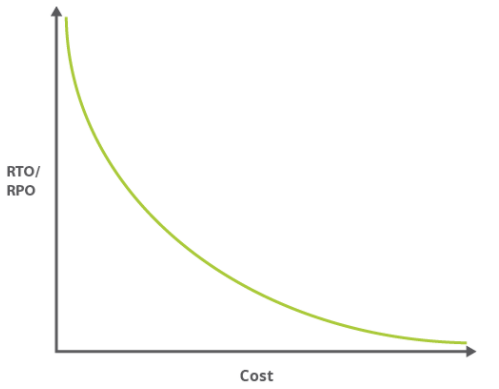
## 10.5 Disaster Recovery Planning

### 10.5.1 Basics of DR planning

DR is a subset of [business continuity planning](http://wikipedia.org/wiki/Business_continuity_planning). DR planning begins with a business impact analysis that defines two key metrics:

* A [*recovery time objective*](https://wikipedia.org/wiki/Recovery_time_objective) (RTO), which is the maximum acceptable length of time that your application can be offline. This value is usually defined as part of a larger [service level agreement (SLA)](https://wikipedia.org/wiki/Service-level_agreement).
* A [*recovery point objective*](https://wikipedia.org/wiki/Recovery_point_objective) (RPO), which is the maximum acceptable length of time during which data might be lost from your application due to a major incident. This metric varies based on the ways that the data is used. For example, user data that's frequently modified could have an RPO of just a few minutes. In contrast, less critical, infrequently modified data could have an RPO of several hours. (This metric describes only the length of time; it doesn't address the amount or quality of the data that's lost.)

Typically, the smaller your RTO and RPO values are (that is, the faster your application must recover from an interruption), the more your application will cost to run. The following graph shows the ratio of cost to RTO/RPO.



Because smaller RTO and RPO values often mean greater complexity, the associated administrative overhead follows a similar curve. A high-availability application might require you to manage distribution between two physically separated data centers, manage replication, and more.

### 10.5.2 DR patterns

DR patterns are considered to be cold, warm, or hot. These patterns indicate how readily the system can recover when something goes wrong.

**Cold:**  In a cold pattern, you have minimal resources in the DR Google Cloud project—just enough to enable a recovery scenario. When there's a problem that prevents the production environment from running production workloads, the failover strategy requires a mirror of the production environment to be started in Google Cloud.

**Warm:**  In a warm pattern you maintain a fully functional, scaled down version of your production workload.

A warm pattern is typically implemented to keep RTO and RPO values as small as possible without the effort and expense of a fully HA configuration. The smaller the RTO and RPO value, the higher the costs as you approach having a fully redundant environment that can serve traffic from two environments. Therefore, implementing a warm pattern for your DR scenario is a good trade-off between budget and availability.

**Hot:** If you have small RTO and RPO values, you can achieve these only by running full HA across your production environment. In a hot pattern you run a mirror of your full production environment in two separate GCP regions both of which actively serve production traffic.

For the AMEX Tier 3 application where RPO is 12 hours and RTO is 2 hours, Google recommends cold or warm DR patterns. For tier 1 applications where RTO and RPO are smaller, Google recommends hot DR.