Published in Google Cloud - Community

Jasbirs Follow

Jan 2 · 8 min read · Listen

Save

GCP Hybrid Networking Patterns — Part 1

Enterprises wont be migrating their entire on-prem workloads to cloud in first go. It makes more sense to migrate portions of workloads at a time to cloud, wherever maximum benefits of migrating to cloud could be reaped. As such, you'll still need a way for your on-prem systems to communicate with your newly minted cloud resources.

There are multiple networking patterns that could be leveraged for designing hybrid connectivity on google cloud. The hybrid connectivity design depends on factors such as:

- The number of VPC networks used for the workloads
- The need for layer 7 traffic inspection of network traffic using network virtual appliances (NVA) appliances
- Access to Google-managed services that use Private Services Access

This would be a multiple part blog series where I would cover about different network topologies available on google cloud platform catering to multiple hybrid connectivity use cases. This blog would cover about Hybrid connectivity to Single VPC(or Shared VPC) patters on Google Cloud Platform. In subsequent series of blogs, I would cover about Hybrid Connectivity to Multiple VPC(or Shared VPC) networks on GCP, Hybrid connectivity using Appliances.

When you have a large number of workload VPC networks, you might need a hub and spoke architecture that supports a large number of VPC networks. This involves connecting multiple workload VPC networks to a transit Hub VPC that has connectivity to on-premises and other clouds. I will cover about Hub and Spoke with VPC Peering to Spokes, Hub and Spoke with HA-VPN to Spokes & Hybrid Connectivity using Appliances and VPC Peering to Spokes.

Hybrid Connectivity to Single VPC(or Shared VPC)

1. Interconnect to On-Premises

a) Cloud Interconnect

Set up Dedicated Interconnect or Partner Interconnect to Google cloud. Connect to two Edge Availability Domains (EAD) in the same Metro in order to achieve 99.99% SLA. You can connect your Cloud Interconnects to multiple regions in the same Shared VPC.

2) VLAN Attachment

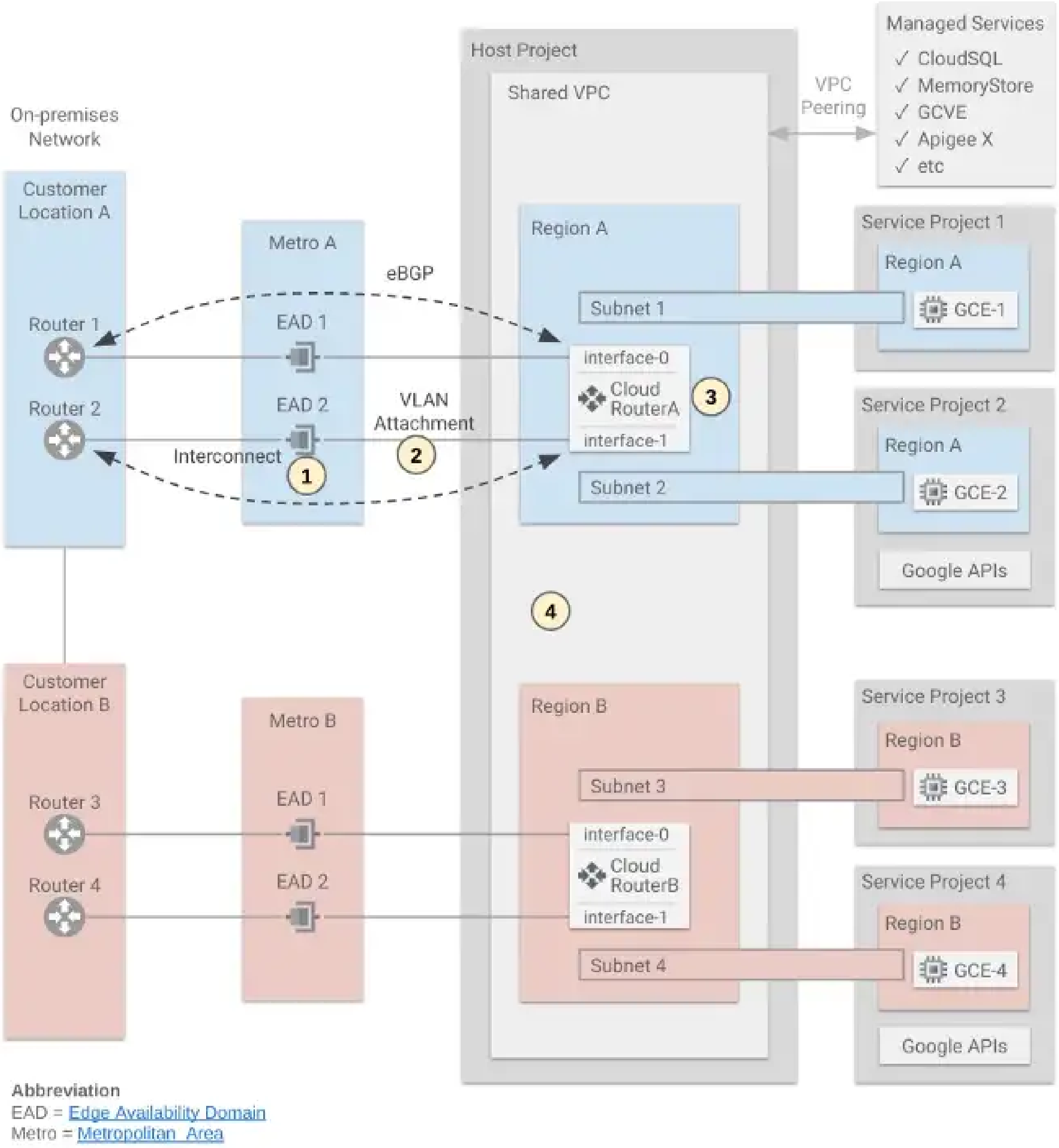
A VLAN attachment connects your interconnect in a Google point of presence (PoP) to a cloud router in a specified GCP region.

3) Cloud Router

A Cloud router exchanges dynamic (BGP) routes between your VPC networks and on-premises routers. You can configure dynamic routing between your on-premises routers and a cloud router in a particular region. Each cloud router is implemented by two software tasks that provide two interfaces for high availability. Configure BGP routing to each of the cloud router's interfaces.

4) VPC Global Dynamic Routing

Configure global dynamic routing in the Shared VPC to allow exchange of dynamic routes between all regions.



2. HA-VPN to On-Premises

1. Cloud HA-VPN

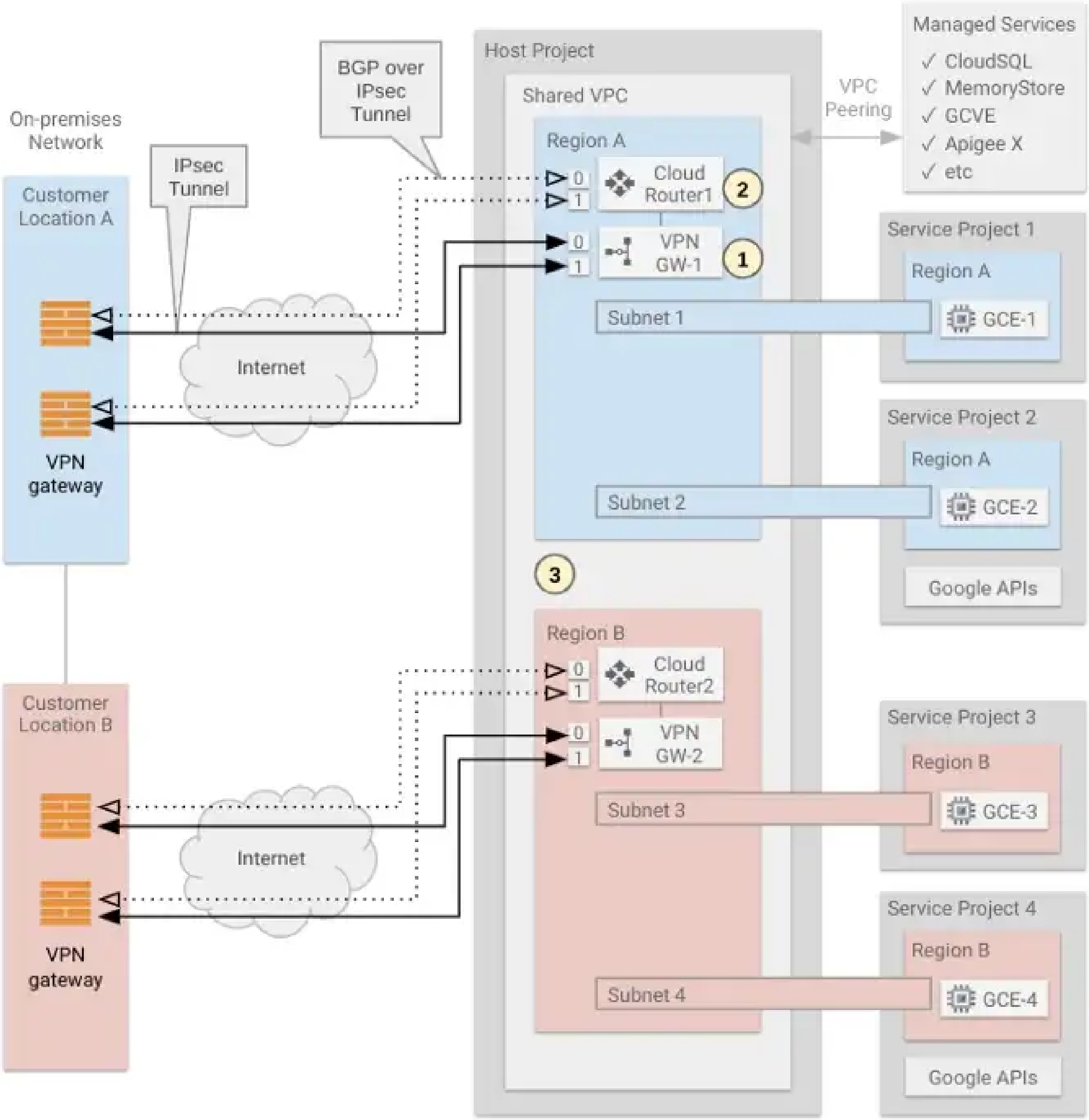
The Cloud HA-VPN gateway is used to establish IPsec tunnels to the on-premises VPN gateway over the Internet. HA-VPN offers a 99.99% SLA. You can have multiple HA-VPN tunnels into different regions in the Shared VPC.

2. Cloud Routers

Configure dynamic routing between the on-premises routers and a cloud router in each region. Each cloud router is implemented by two software tasks that provide two interfaces for high availability. Configure BGP routing to each of the cloud router's interfaces.

3. VPC Global Dynamic Routing

Configure global dynamic routing in the Shared VPC to allow exchange of dynamic routes between all regions.



3. DNS

Overview

In a hybrid environment, DNS resolution can be performed in GCP or on-premises. Let's consider a use case where on-premises DNS servers are authoritative for on-premises DNS zones, and Cloud DNS is authoritative for GCP zones.

1) On-premises DNS

Configure your on-premises DNS server to be authoritative for on-premises DNS zones. Configure DNS forwarding (for GCP DNS names) by targeting the Cloud DNS inbound forwarding IP address, which is created via the Inbound Server Policy configuration in the Shared VPC. This allows on-premises network to resolve GCP DNS names.

2) Host Project (Shared VPC) — DNS Egress Proxy

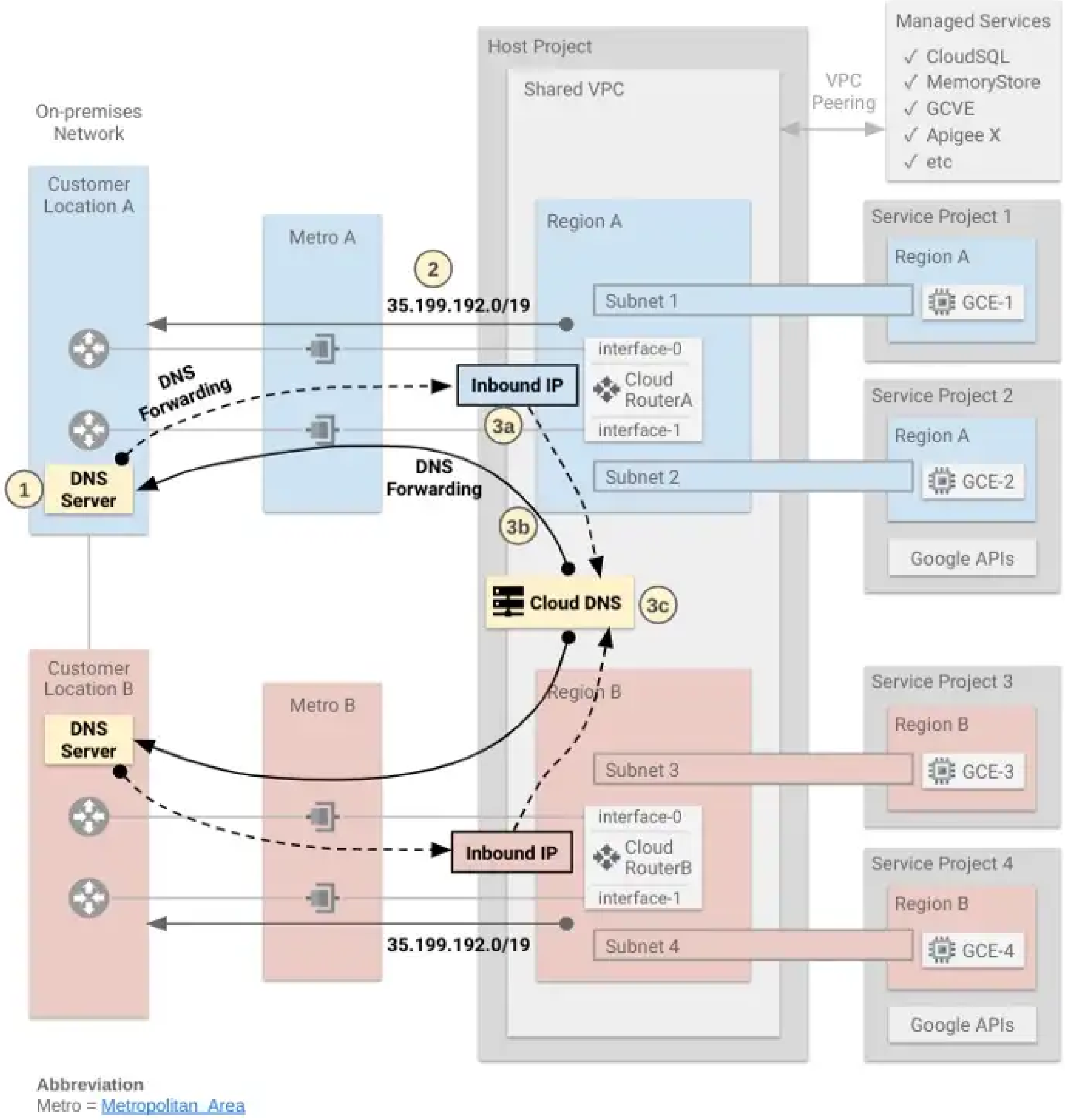
Advertise the Google DNS Egress Proxy range 35.199.192.0/19 to the on-premises network via the cloud routers. Outbound DNS requests from Google to on-premises are sourced from this IP address range.

3) Host Project (Shared VPC) — Cloud DNS

a) Configure an Inbound Server Policy for inbound DNS requests from on-premises.

b) Configure DNS forwarding zone (for on-premises DNS zones) targeting the on-premises DNS resolvers.

c) Configure DNS Private Zones in the ***Host Project*** and attach ***Shared VPC*** to the zone. This allows hosts (on-premises and all service projects) to resolve the Prod DNS names.



4. Private Service Connect(PSC) for Google APIs(Access to all Supported APIs and Services)

Overview

You can use Private Service Connect (PSC) to access all supported Google APIs and services from Google Compute Engine (GCE) hosts and on-premises hosts; using the internal IP address of a PSC endpoint in the Shared VPC. Let’s consider PSC access to a service in *Service Project 4* via the Shared VPC.

Create a PSC Endpoint

1. Choose a PSC endpoint address (e.g 10.1.1.1) and create a PSC endpoint in the Shared VPC with a target of **“all-apis”**- which gives access to all supported Google APIs and services. Service Directory automatically creates a DNS record (with DNS name of **p.googleapis.com**) linked to the PSC endpoint IP address.

Access from GCE Hosts

GCE-4 host in *Service Project 4* can access all supported Google APIs and services via the PSC endpoint in the Shared VPC.

2. Enable Private Google Access on all subnets with compute instances that require access to Google APIs via PSC.

3. If your GCE clients can use custom DNS names (e.g. **storage-xyz.p.**googlepapis.com), you can use the auto-created **p.googleapis.com** DNS name.

4. If your GCE clients cannot use custom DNS names, you can create Cloud DNS records using the default DNS names (e.g **storage**.googleapis.com).

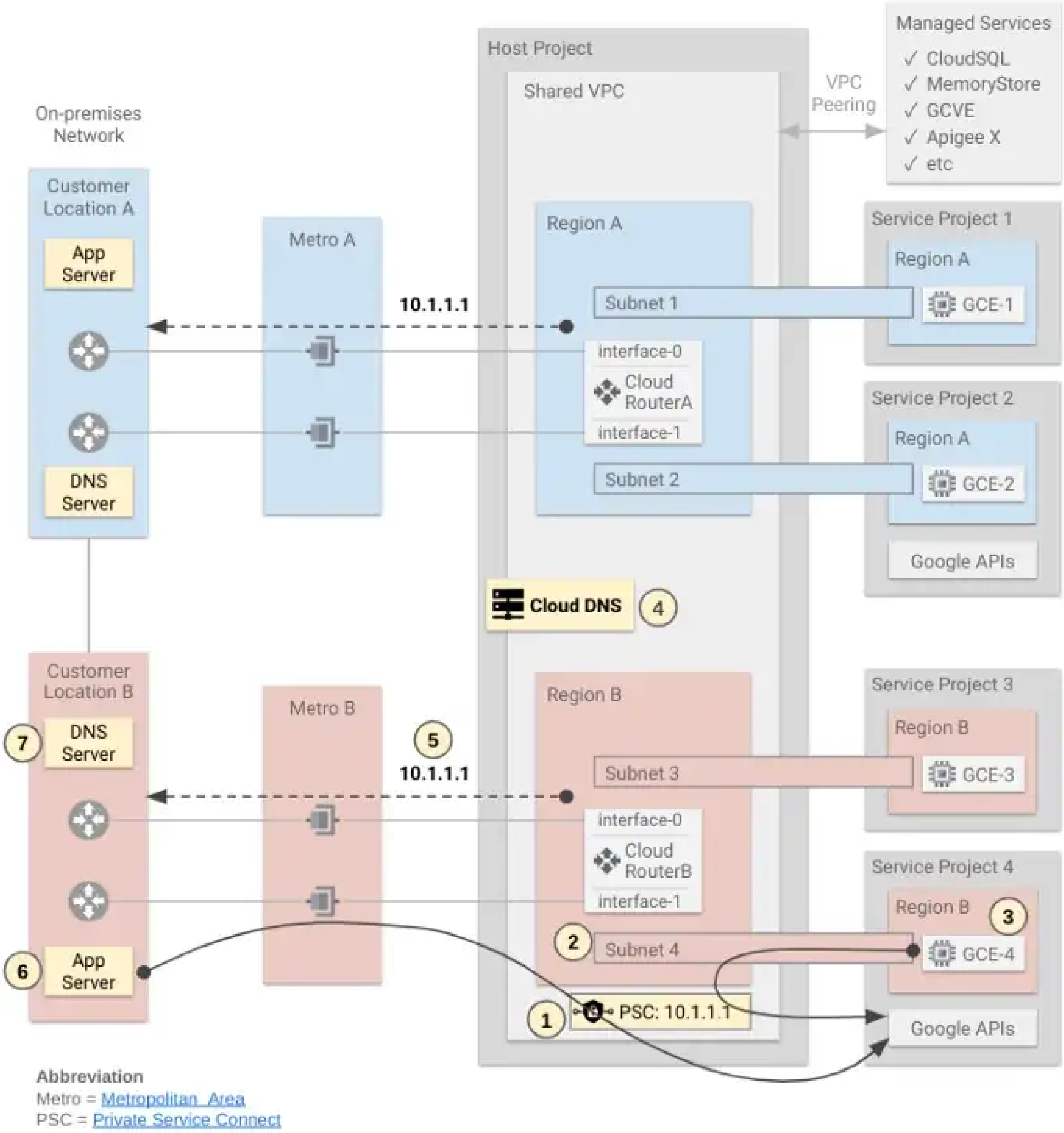
Access from On-premises Hosts

On-premises hosts can access all supported Google APIs and services via the PSC endpoint in the Shared VPC.

5. Advertise the PSC endpoint address to the on-premises network.

6. If your on-premises clients can use custom DNS names (e.g. **storage-xyz.p.**googlepapis.com), you can create A records mapping the custom DNS names to the PSC endpoint address.

7. If your on-premises clients cannot use custom DNS names, you can create A records mapping the default DNS names (e.g **storage.**googleapis.com) to the PSC endpoint address.



5. Private Service Connect (PSC) for Google APIs (Access to APIs and Services supported on VPC Service Control)

Overview

You can use Private Service Connect (PSC) to access all supported secure Google APIs and services from Google Compute Engine (GCE) hosts and on-premises hosts; using the internal IP address of a PSC endpoint in the Shared VPC. Let's consider PSC access to a service in *Service Project 4* via the Shared VPC.

Create a PSC Endpoint

1. Choose a PSC endpoint address (e.g 10.1.1.1) and create a PSC endpoint in the Shared VPC with a target of **“vpc-sc”**- which gives access to Google APIs and services that are supported under VPC Service Control. Service Directory automatically creates a DNS record (with DNS name of **p.googleapis.com**) linked to the PSC endpoint IP address.

Access from GCE Hosts

GCE-4 host in *Service Project 4* can access (VPC service control) supported Google APIs and services via the PSC endpoint in the Shared VPC.

2. Enable Private Google Access on all subnets with compute instances that require access to Google APIs via PSC.

3. If your GCE clients can use custom DNS names (e.g. **storage-xyz.p.**googlepapis.com), you can use the auto-created **p.googleapis.com** DNS name.

4. If your GCE clients cannot use custom DNS names, you can create Cloud DNS records using the default DNS names (e.g **storage**.googleapis.com).

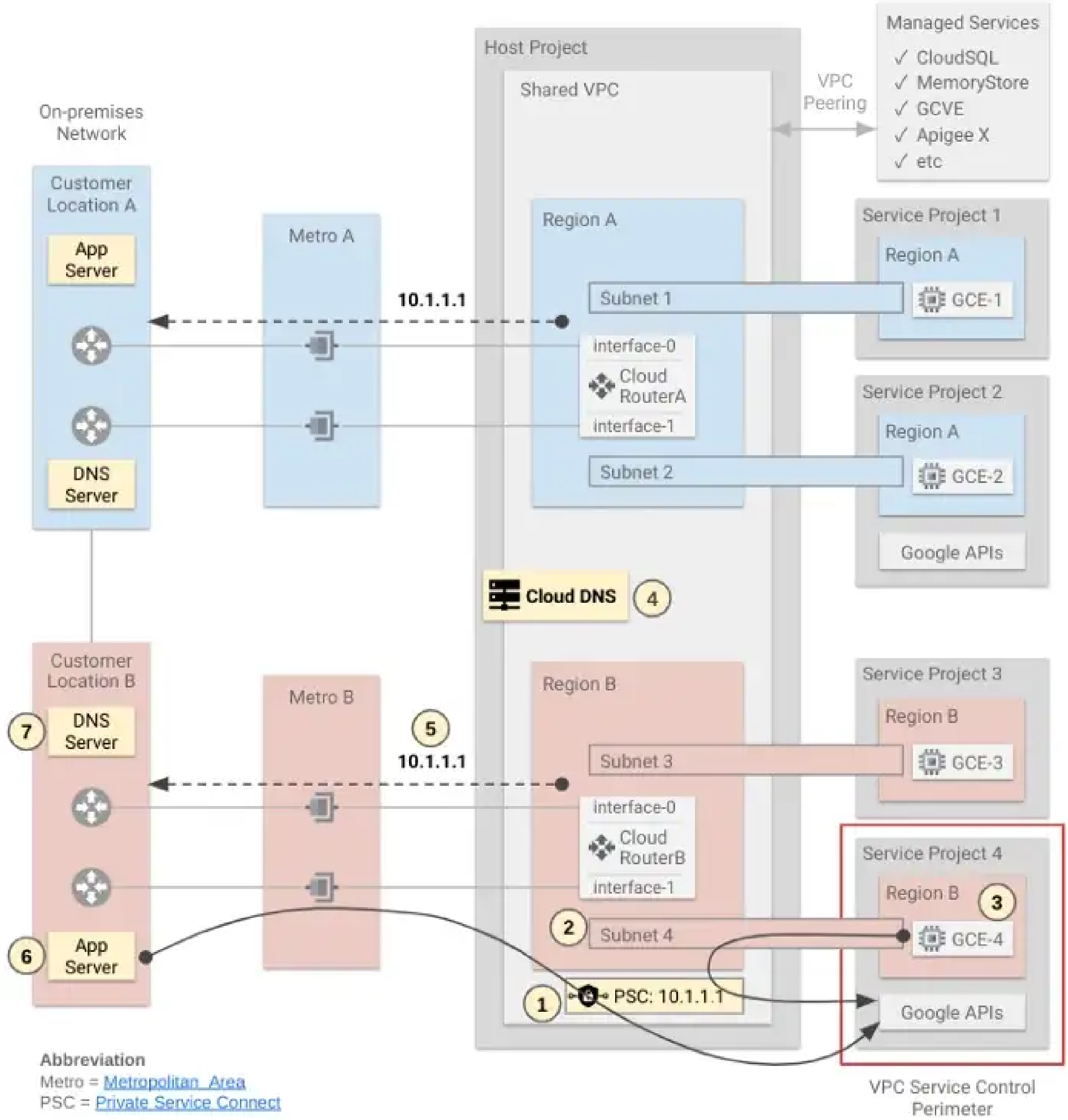
Access from On-premises Hosts

On-premises hosts can access all secure Google APIs and services via the PSC endpoint in the Shared VPC.

5. Advertise the PSC endpoint address to the on-premises network.

6. If your on-premises clients can use custom DNS names (e.g. **storage-xyz.p.**googlepapis.com), you can create A records mapping the custom DNS names to the PSC endpoint address.

7. If your on-premises clients cannot use custom DNS names, you can create A records mapping the default DNS names (e.g **storage.**googleapis.com) to the PSC endpoint address.



6. VPC Service Control

Overview

VPC Service Control uses Ingress and egress rules to control access to and from a perimeter. The rules specify the direction of allowed access to and from different identities and resources.

Let’s consider a specific use case where we require access to a protected service in *Service Project 4* via the Shared VPC.

Below is an description of VPC service control action for our specific scenario.

1) Perimeter — Service Project

The perimeter contains a service project (***Service Project 4***); and includes Google APIs and services to be protected in the service project.

2) API access from GCE Hosts

A GCE client can access secured APIs through a PSC endpoint in the Shared VPC. Let's consider the perimeter around ***Service Project 4***. The network interface of the compute instance ***GCE-4*** is in the ***Shared VPC*** of ***Host Project***. API calls from ***GCE-4*** instance to a service (e.g. storage.googleapis.com) in ***Service Project 4***, appear to originate from ***Host Project*** — where the instance interface and PSC endpoint are located.

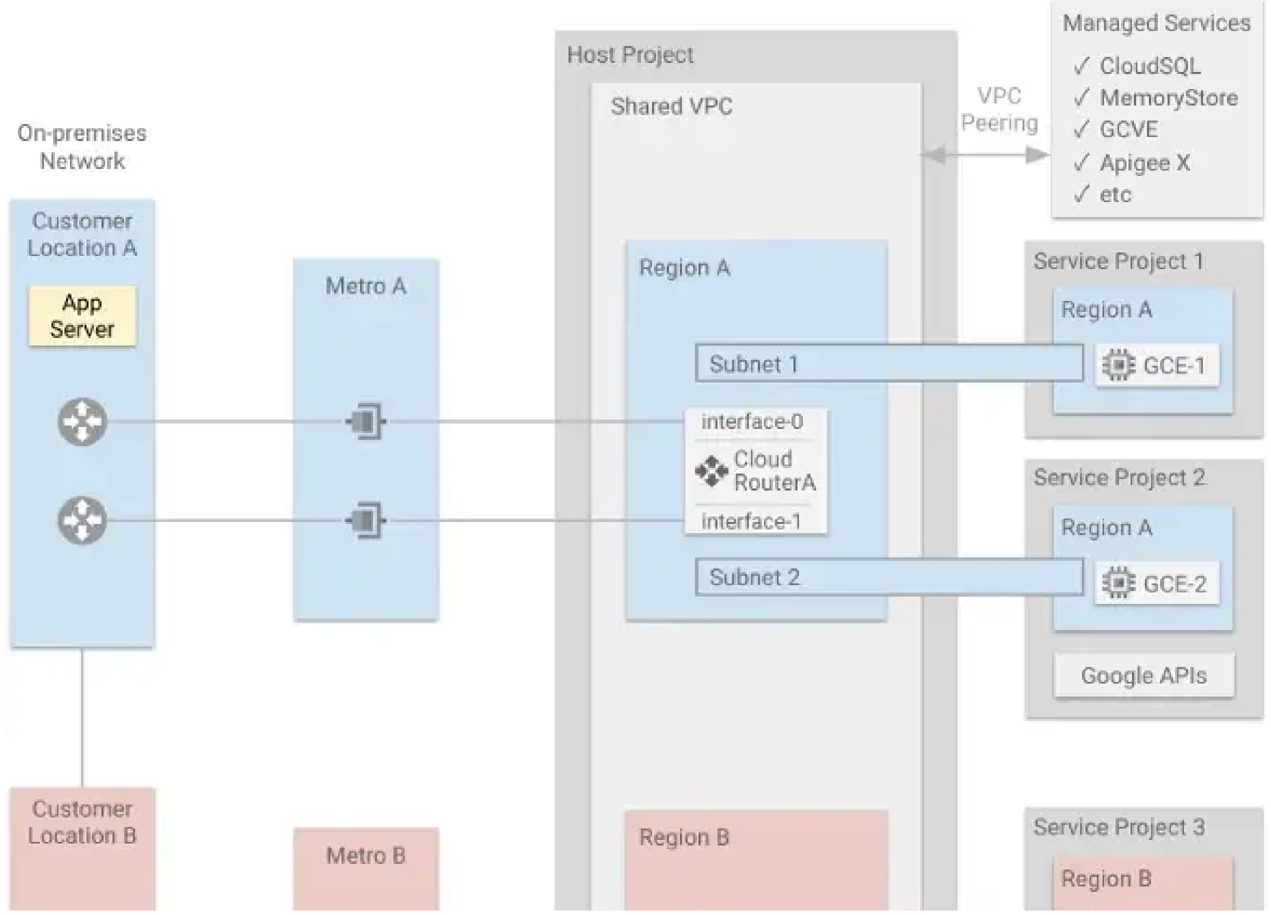
3) Ingress Rule — Host Project into Perimeter

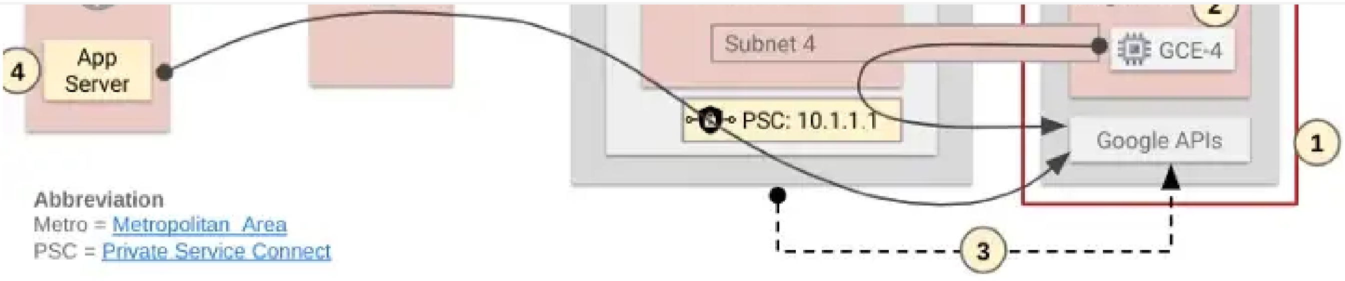
Configure an ingress rule that allows Google API calls from ***Host Project*** to the protected services in ***Service Project 4*** perimeter. This rule allows API calls from the GCE instances (e.g. ***GCE-4***) into the perimeter.

4) API access for on-premises hosts

On-premises hosts can access secured APIs in ***Service Project 4*** via the PSC endpoint in ***Shared VPC***. API calls from on-premises to services in ***Service Project 4*** appear to originate from ***Host Project*** — where the Interconnect and the PSC endpoint are located.

The ingress rule (in step 3) allows API calls from on-premises to ***Service Project 4*** perimeter via ***Host Project***.





Hybrid Connectivity

Private Service Connect

Vpc Service Control

Google Cloud Platform

Infrastructure