VPC Service Controls (https://cloud.google.com/vpc-service-controls/)
Documentation (https://cloud.google.com/vpc-service-controls/docs/) Guides

Setting up private connectivity to Google APIs and services

You can use VPC Service Controls to control access to Google APIs and services from hosts that use private IP addresses. These hosts can be VM instances in a VPC network or clients in an on-premises network.

To restrict Private Google Access within a service perimeter to only VPC Service Controls supported (https://cloud.google.com/vpc-service-controls/docs/overview) Google APIs and services, hosts must send their requests to the `restricted.googleapis.com` domain name instead of `*.googleapis.com`. The `restricted.googleapis.com` domain resolves to a VIP (virtual IP address) range `199.36.153.4/30`. This IP address range is not announced to the Internet.

Note: If you require access to other Google APIs and services that aren't supported by VPC Service Controls, you can use `199.36.153.8/30` (`private.googleapis.com`). However, we recommend that you use `restricted.googleapis.com`, which integrates with VPC Service Controls and mitigates data exfiltration risks. In either case, VPC Service Controls service perimeters are always enforced on APIs and services that support VPC Service Controls.

The following sections describe how to set up private connectivity for hosts in a VPC network or on-premises network. For an overview and example topology diagrams, refer to Private Google Access with VPC Service Controls (https://cloud.google.com/vpc-service-controls/docs/private-connectivity).

Before you begin

- You must enable the APIs (https://support.google.com/cloud/answer/6158841?hl=en) that you want to access through the APIs & services page (https://console.cloud.google.com/apis/dashboard) in the Google Cloud Console.
- Project owners, editors, and IAM members with the Network Admin (https://cloud.google.com/compute/docs/access/iam#compute.networkAdmin) role can create or update subnets and assign IP addresses. For more information on roles, read the IAM roles (https://cloud.google.com/compute/docs/access/iam) documentation.
- Private Google Access and Private Google Access for on-premises hosts requires a VPC network (https://cloud.google.com/vpc/docs/vpc). Both auto and custom mode VPC networks are supported. Legacy networks (https://cloud.google.com/vpc/docs/legacy) are not supported.
- For VM instances in a VPC network, they must have a private IP address only (no public IP address) and be in subnet with Private Google Access (https://cloud.google.com/vpc/docs/private-google-access#pga) enabled.
- For on-premises hosts, you must have an existing Cloud VPN (https://cloud.google.com/vpn/docs) tunnel or an Cloud Interconnect (https://cloud.google.com/interconnect/docs) connection to your VPC network.

Overview of procedure

To set up private connectivity, complete the following tasks:

- Configure routes for the destination `199.36.153.4/30`. For more information, see Configuring routes (#configuring-routes).
- Configure firewall rules to allow the appropriate traffic to the restricted Google APIs IP address range. For more information, see Configuring firewall rules (#configuring-firewall).
- Configure DNS so that traffic to Google APIs resolves to the Restricted Google APIs IP address range. For more information, see Configuring DNS (#configuring-dns).

Configure a route to `restricted.googleapis.com`

For both Private Google Access and Private Google Access for on-premises hosts, your VPC network must include a route for with a destination to `199.36.153.4/30` whose next hop is the default Internet gateway. Even though the next hop is a default Internet gateway, traffic sent to `199.36.153.4/30` remains within Google's network. Your VPC network might already have a default route (https://cloud.google.com/vpc/docs/routes#routingpacketsinternet) whose next hop is the default Internet gateway. If not, you can create a custom static route whose destination is `199.36.153.4/30` and whose next hop is the default Internet gateway.

Note: You can remove other routes that have a next hop of the default Internet gateway to prevent access to the Internet, such as VM instances with external IP addresses. As long as you have a custom static route with the destination `199.36.153.4/30`, you can remove other routes that have a next hop of the default Internet gateway.

In addition to the custom static route, Private Google Access for on-premises hosts requires a custom route advertisement so that hosts in the on-premises network can learn about the restricted VIP. You can create a custom dynamic route using Cloud Router to announce the restricted VIP.

For more information about working with VPC routes, see Using Routes (https://cloud.google.com/vpc/docs/using-routes) in the VPC documentation.

Configuring a custom static route in a VPC network

Add a custom static route to enable access to Google managed services that are supported by VPC Service Controls.

- Add a custom route that allows access only to Google managed services secured by VPC Service Controls.

  ```
  gcloud compute routes create ROUTE_NAME \
    --network=NETWORK_NAME \
    --destination-range=199.36.153.4/30 \
    --next-hop-gateway=default-internet-gateway
  ```

  Where:

  - ***ROUTE_NAME*** is a name for the custom route.
  - ***NETWORK_NAME*** is the name of your VPC network.

Announcing the restricted route to hosts in an on-premises network

If you're using Private Google Access for on-premises hosts, configure routes so that Google API traffic is forwarded through your Cloud VPN or Cloud Interconnect connection. To announce the restricted VIP (`199.36.153.4/30`) to your on-premises network, use Cloud Router Custom Route Advertisement (https://cloud.google.com/router/docs/how-to/advertising-custom-ip). This IP address range is only accessible to on-premises hosts that can reach your VPC network through private IP addresses.

Note: If you have multiple tunnels or interconnects, you can't create cross-region asymmetric routes back to your on-premises network. Google Cloud doesn't support them.

You can add this custom route advertisement to a Cloud Router (for all BGP sessions on the router) or a select BGP session (for a single Cloud VPN tunnel or VLAN attachment).

To create a custom route advertisement for the restricted range for all BGP sessions on an existing Cloud Router:

CONSOLE GCLOUD

1. Go to the Cloud Router page in the Google Cloud Console.

   CLOUD ROUTER LIST (HTTPS://CONSOLE.CLOUD.GOOGLE.COM/INTERCONNECT/ROUTERS/LIST)
2. Select the Cloud Router to update.
3. In the Cloud Router's detail page, select **Edit**.
4. Expand the **Advertised routes** section.
5. For the **Routes**, select **Create custom routes**.
6. Select **Advertise all subnets visible to the Cloud Router** to continue advertising the subnets available to the Cloud Router. Enabling this option mimics the Cloud Router's default behavior.
7. Select **Add custom route** to add an advertised route.
8. Configure the route advertisement.
   - **Source** — Select **Custom IP range** to specify a custom IP range.
   - **IP address range** — Specify `199.36.153.4/30`.
   - **Description** — Add a description of `Restricted Google APIs IPs`.
9. After you're done adding routes, select **Save**.

To create a custom route advertisement for the restricted range on a specific BGP session of an existing Cloud Router:

CONSOLE GCLOUD

1. Go to the Cloud Router page in the Google Cloud Console.
   CLOUD ROUTER LIST (HTTPS://CONSOLE.CLOUD.GOOGLE.COM/INTERCONNECT/ROUTERS/LIST)
2. Select the Cloud Router that contains the BGP session to update.
3. In the Cloud Router's detail page, select the BGP session to update.
4. In the BGP session details page, select **Edit**.
5. For the **Routes**, select **Create custom routes**.
6. Select **Advertise all subnets visible to the Cloud Router** to continue advertising the subnets available to the Cloud Router. Enabling this option mimics the Cloud Router's default behavior.
7. Select **Add custom route** to add an advertised route.
8. Configure the route advertisement.
   - **Source** — Select **Custom IP range** to specify a custom IP range.
   - **IP address range** — Specify `199.36.153.4/30`.
   - **Description** — Add a description of `Restricted Google APIs IPs`.
9. After you're done adding routes, select **Save**.

Configure firewall rules

For Private Google Access, VM instances use internal IP addresses and don't require external IP addresses to reach protected Google API resources. However, it's possible for VM instances to possess external IP addresses or otherwise meet the requirements for Internet access (https://cloud.google.com/vpc/docs/vpc#internet_access_reqs). In addition to custom routes (#configure-routing), you can restrict egress traffic from VM instances in your VPC network by creating firewall rules to deny egress traffic.

By default, the implied allow egress firewall rule (https://cloud.google.com/vpc/docs/firewalls#default_firewall_rules) permits VM instances to send traffic to any destination if an applicable route exists. You can create an egress deny rule to block all outbound traffic, and then create higher priority egress allow rules to permit traffic to selected destinations in your VPC network and to the `199.36.153.4/30` (`restricted.googleapis.com`) IP address range. All communication to `restricted.googleapis.com` is on TCP port `443`.

Note: Before blocking Internet access, carefully consider the impact on your VM instances. Blocking Internet access can reduce your risk of data exfiltration, but it can also block legitimate traffic, including essential traffic for software updates. Without Internet access, you are only able to access your VM instances through an on-premises network connected through a Cloud VPN tunnel or Cloud Interconnect connection. VM instances also won't be able to call third-party APIs and services.

For more information about working with VPC firewall rules, see Using Firewall Rules (https://cloud.google.com/vpc/docs/using-firewalls) in the VPC documentation.

Firewall rules in on-premises networks

You must configure your on-premises firewall rules to allow traffic from your on-premises hosts to reach `199.36.153.4/30`.

Configuring DNS

To make use of the Restricted Google APIs IP addresses, configure your DNS server to resolve `*.googleapis.com` as a CNAME to `restricted.googleapis.com`, and configure an A record for `restricted.googleapis.com`. For general use of VPC Service Controls, we recommend that you use Cloud DNS managed private zones for your VPC networks.

For on-premises access, you can configure a Cloud DNS inbound forwarding policy to enable on-premises name servers to query a Cloud DNS managed private zone, or you can configure an on-premises name server, such as one using BIND (https://www.wikipedia.org/wiki/BIND):

Note: You might need to also configure DNS for gcr.io if, for example, you're using GKE. For more information, refer to Setting up Container Registry for GKE private clusters (https://cloud.google.com/vpc-service-controls/docs/set-up-gke#configure-dns).

- Cloud DNS private DNS zones enable you to host a DNS zone accessible from authorized VPC networks and, if you configure forwarding, from certain on-premises name servers.

You can create a private zone for `googleapis.com` with an A record for `restricted.googleapis.com` and appropriate CNAME records for each `*.googleapis.com` name. Cloud DNS private zones do not support partial overrides, which means that you can only choose to redirect **all** requests to `*.googleapis.com` to `restricted.googleapis.com`. As a result, you won't be able to use any Google APIs and services that do not support use of the `restricted.googleapis.com` VIP. For more information, see Managing Zones (https://cloud.google.com/dns/zones/).

- Custom BIND and the `restricted.googleapis.com` VIP cannot be used for Dataflow (https://cloud.google.com/dataflow/docs/guides/routes-firewall#dns_limitations) because DNS resolution of Dataflow can't be customized.

Note: For general use of VPC Service Controls, we strongly recommend that you use private DNS zones for your VPC networks. We suggest that BIND is used only for on-premises implementations.

Configuring DNS with Cloud DNS

Use Cloud DNS to enable DNS resolution for VM instances in your VPC network, hosts in an on-premises network, or both. If you're using Shared VPC, see Private zones and Shared VPC (https://cloud.google.com/dns/docs/overview#shared-vpc) in the Cloud DNS documentation. Additionally, if you're using Shared VPC, ensure that the Shared VPC network host project is included in the same service perimeter as projects that connect to the network.

Note: If you use Dataflow, you can't customize it's DNS resolution.

1. Create a managed private zone for your VPC network.

```
gcloud beta dns managed-zones create ZONE_NAME \
 --visibility=private \
 --networks=https://www.googleapis.com/compute/v1/projects/PROJECT_ID/global/ne
 --description=DESCRIPTION \
 --dns-name=googleapis.com
```

   - ***ZONE_NAME*** is a name for the zone that you are creating. For example, `vpc`. This name will be used in each of the following steps.
   - ***PROJECT_ID*** is the ID of the project that hosts your VPC network.
   - ***NETWORK_NAME*** is the name of your VPC network.

- ***DESCRIPTION*** is an optional, human-readable description of the managed zone.

2. Start a transaction.

```
gcloud dns record-sets transaction start --zone=ZONE_NAME
```

    - ***ZONE_NAME*** is the name of the zone you created in the first step.

3. Add DNS records.

```
gcloud dns record-sets transaction add --name=*.googleapis.com. \
    --type=CNAME restricted.googleapis.com. \
    --zone=ZONE_NAME \
    --ttl=300
```

    - ***ZONE_NAME*** is the name of the zone you created in the first step.

```
gcloud dns record-sets transaction add --name=restricted.googleapis.com. \
    --type=A 199.36.153.4 199.36.153.5 199.36.153.6 199.36.153.7 \
    --zone=ZONE_NAME \
    --ttl=300
```

    - ***ZONE_NAME*** is the name of the zone you created in the first step.

4. Execute the transaction.

```
gcloud dns record-sets transaction execute --zone=ZONE_NAME
```

    - ***ZONE_NAME*** is the name of the zone you created in the first step.

5. Optional. To enable on-premises hosts to reach the restricted VIP, complete the following steps:

    a. Create a DNS policy and enable inbound DNS forwarding to make VPC network's name resolution services externally available to systems in on-premises networks,

    ```
    gcloud beta dns policies create POLICY_NAME \
     --networks=https://www.googleapis.com/compute/v1/projects/PROJECT_ID/glob
     --enable-inbound-forwarding \
     --description=DESCRIPTION
    ```

        - ***POLICY_NAME*** is a name for the policy that you are creating. For example, `apipolicy`.

- ***PROJECT_ID*** is the ID of the project that hosts your VPC network.
- ***NETWORK_NAME*** is the name of your VPC network.
- ***DESCRIPTION*** is an optional, human-readable description of the managed zone.

b. In your on-premises network, point your on-premises DNS to the Cloud DNS forwarder IP address. To find the forwarder IP address, use the `compute addresses list` command:

```
gcloud compute addresses list --filter='name ~ ^dns-forwarding.*' \
 --format='csv[no-heading](address, subnetwork)'
```

★ **Note:** Your VPC network only has a forwarder IP address if you have configured an inbound DNS forwarding policy for it. For more information about DNS policies, see the Cloud DNS overview (https://cloud.google.com/dns/docs/overview#dns-server-policy) and Creating a DNS policy that enables inbound DNS forwarding (https://cloud.google.com/dns/zones/#creating_a_dns_policy_that_enables_inbound_dns_forwarding)
.

Configuring DNS with BIND

If you use BIND (https://www.wikipedia.org/wiki/BIND) for DNS resolution, you can configure it to resolve Google API requests to the restricted Google APIs. Use the following example BIND configuration, which makes use of response policy zones (https://wikipedia.org/wiki/Response_policy_zone) (RPZ) to achieve this behavior:

1. Add the following lines to `/etc/bind/named.conf`:

```
include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
```

2. Add the following lines to `/etc/bind/named.conf.options`:

```
options {
  directory "/var/cache/bind";

  dnssec-validation no;
  auth-nxdomain no;    # conform to RFC 1035
  listen-on-v6 { any; };
  listen-on { any; };
  response-policy { zone "googleapis.zone"; };
  allow-query { any;};
};
```

3. Add the following lines to `/etc/bind/named.conf.local`:

```
include "/etc/bind/named.conf.default-zones";



zone "googleapis.zone" {
  type master;
  file "/etc/bind/db.googleapis.zone";
  allow-query {none;};
};
```

4. Add the following lines to `/etc/bind/db.googleapis.zone`:

```
$TTL 1H
@                       SOA LOCALHOST. noreply.localhost(1 1h 15m 30d 2h)
                        NS  LOCALHOST.



*.googleapis.com CNAME restricted.googleapis.com.
restricted.googleapis.com CNAME rpz-passthru.
```

Last updated November 21, 2019.