Network security

Redis Enterprise Cloud supports two types of network security: IP Restrictions and VPCs. These features are available in most Redis Cloud configurations, as shown here:

<table>
<thead>
<tr>
<th></th>
<th>VPC Support</th>
<th>IP Restrictions</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS</td>
<td>Flexible and Annual</td>
<td>Fixed (paid), Flexible, and Annual</td>
</tr>
<tr>
<td>GCP</td>
<td>Flexible and Annual</td>
<td>Fixed (paid), Flexible, and Annual</td>
</tr>
<tr>
<td>Azure</td>
<td>Annual</td>
<td>Annual</td>
</tr>
</tbody>
</table>

IP and subnet restrictions

You can restrict database access to a configurable set of source IP addresses and subnets. This is roughly equivalent to using iptables to limit access to a host.

Adding restrictions

To restrict a database to a specific set of source IP addresses or subnets:

1. From the admin console, navigate to the View Database screen for a particular database.

2. Click on the edit icon to enter the Edit Database screen.


4. From here, you can use the UI to add individual IP addresses and subnets, one at a time.
Virtual private clouds

A Virtual Private Cloud (VPC) is an isolated set of resources within a public cloud, usually having its own subnets and VLAN.

Databases in Flexible and Annual subscriptions are almost always deployed in a Redis VPC. In most cases, you’ll need to create a VPC peering connection to access these databases. A VPC peering connection allows unrestricted network access between two VPCs.

How you create these connections and the features supported vary somewhat by public cloud provider. You can read about VPC usage for AWS, GCP, and Azure below.

VPCs with AWS

Subscriptions that run on AWS support two VPC options. To ensure that you can securely connect to your database, you must either create a VPC peering connection or deploy your subscription in your own VPC.

Create a VPC peering connection

Below are instructions for creating a VPC peering connection for AWS. Once you’ve created this connection, you may also want to consider configuring a CIDR whitelist to allow connection only from specific IP address blocks or security groups.

To create a VPC peering connection:

1. In Subscriptions, click on the subscription requiring a VPC peering connection
2. In Security > VPC Peering, click . You’ll then see a form like the following:
3. Enter your VPC peering details:
   - AWS Account ID
   - AWS Region
   - AWS VPC ID
   - VPC CIDR (must not overlap with the Redis CIDR block)

   Then click **Initiate Peering**.

4. Next, you'll need to approve the VPC peering request. To do that, log in to your AWS management console.
   1. Go to: Services > VPC > Peering Connections
   2. Select the peering connection with the Peering ID of your peering request.
   3. Go to **Description** and note the Requester VPC CIDRs shown in the Peering Connection details.
   4. Click **Actions** and select **Accept Request**.
   5. To confirm, click **Yes, Accept**.
   6. Finally, update your routing tables for the peering connection:
      1. After you accept the peering request, click **Modify my route tables now**.
      2. Find the ID of your VPC in the list of routes and select it.
      3. Go to **Routes** and click on **Edit Routes**.
4. To add a route, click Add Route.

5. In the Destination field, enter the Requester VPC CIDRs shown when you accepted the peering request. This is the Redis Cloud VPC CIDR address, to which your application’s VPC should connect.

6. In the Target field, select Peering Connection and select the relevant Peering ID.

7. Click Save Routes and Close.

Once your VPC peering request is accepted, the status in your subscription’s VPC Peering tab will indicate ‘Peer Established’.

If you correctly follow these steps, you will be able to connect to your database. If you have any problems or questions, please don’t hesitate to contact Redis support.

Configure the CIDR whitelist

The CIDR whitelist defines a range of IP addresses and/or AWS security groups permitted to access databases in the Redis Cloud VPC.

To define the CIDR whitelist:

1. In Subscriptions, click on the subscription for VPC peering.

2. Go to: Security > CIDR Whitelist.

3. If there are no CIDR whitelist entries, click . You’ll see a form similar to this:

4. Specify the Type of whitelist entry as either:
   - **IP Address**: For the value, enter the IP block in CIDR format for the traffic that you want to allow access for.
   - **Security Group**: For the value, enter the ID of the AWS security group to grant access to.

5. Click .

6. Next, either:
   1. Add more whitelist entries by clicking .
   2. Or apply the changes to the whitelist by selecting Apply all changes.

Deploy in your own VPC

As an alternative to VPC peering, you can create and deploy a Flexible subscription directly in your own AWS VPC. You need to do this at the time you create your subscription.

1. Navigate to the New Subscription page:
2. In the **Flexible plan** section, select the **Create** button.

3. When the Create Custom Subscription screen appears, locate the **Networking** section of the **Setup** tab and then select the option to deploy in an existing VPC.
   
   Next, enter the subnet (**Deployment CIDR**) where you want your subscription deployed and your VPC ID.

4. Fill in the remaining details for your subscription.

Once your subscription and databases have been provisioned, you’ll be able to access those databases directly from within your own VPC.

### VPCs with GCP

Subscriptions that run on GCP **require** a VPC peering connection.

To create a VPC peering connection:

1. In **Subscriptions**, click on the subscription requiring a VPC peering connection

2. In **Security > VPC Peering**, click ✈️. You’ll then see a form like the following:
VPC Peering

In order to establish the peered connection we need some information about your GCP VPC

**Project ID**

high-speed-data-delivery-inc

**Network name**

data-ingestion

To finish the process after peering was initiated, please run the following gcloud command to accept the peering on your side

```bash
gcloud compute networks peerings create rl-c9904-us-central1-3-rfrcp --project high-speed-data-delivery-inc --network data-ingestion --peer-network=9904-us-central1-3-rfrcp --peer-project=gcf401e2a12bde876-tp --auto-create-routes
```

3. Enter your VPC peering details:

- GCP Project ID
- GCP Network name

Before you click **Initiate Connection**, be sure that you copy the gcloud command generated at the bottom of the form:

To finish the process after peering was initiated, please run the following gcloud command to accept the peering on your side

```bash
gcloud compute networks peerings create rl-c9904-us-central1-3-rfrcp --project high-speed-data-delivery-inc --network data-ingestion --peer-network=9904-us-central1-3-rfrcp --peer-project=gcf401e2a12bde876-tp --auto-create-routes
```

4. Run the gcloud command you just copied to approve the VPC peering connection.

Once your VPC peering request is accepted, the status in your subscription’s **VPC Peering** tab will indicate 'Peer Established'.

If you correctly follow these steps, you will be able to connect to your database. If you have any problems or questions, please don't hesitate to contact Redis support.

**VPCs with Azure**

When you request a Redis Cloud Annual subscription, all databases will be deployed in your own Azure VPC.

---

**Updated:** January 12, 2022