Setting up Traffic Director has three phases:

- Performing prerequisite tasks, such as ensuring that required accounts have the correct permissions
- Preparing the hosts where your microservices run, whether these are VMs or Kubernetes pods
- Setting up the components that manage traffic routing and load balancing

This guide describes how to perform the prerequisite tasks. Before you read this guide, familiarize yourself with Traffic Director Concepts (/traffic-director/docs/traffic-director-concepts).

Additional guides describe how to prepare the hosts and set up the traffic routing and load balancing components.

Regardless of the Traffic Director configuration you choose, complete the following tasks before you start the configuration process:

- Decide on an Envoy binary (#decide-envoy-binary)
- Ensure that a Linux host is available (#linux-host)
- Grant the required permissions (#grant-permissions)
- Enable the Traffic Director API for your project (#enable-api)
- Ensure that the service account has sufficient permissions to access the Traffic Director API (#enable-service-account)

The following sections provide instructions for each task.

During the configuration process, you install Envoy binaries on the host.

Traffic Director supports Envoy version 1.9.1 or later. We strongly recommend using the most recent Envoy version to ensure that all known security vulnerabilities are mitigated.

The Envoy binary is currently distributed as a Docker image. If you decide to use the most recent binary, you need Docker tools to unpack the Envoy proxy binary. You also need Docker permissions to pull the image from Docker. If you run the Docker tools as a non-root user, follow Docker's post-installation instructions at https://docs.docker.com/install/linux/linux-postinstall/ (https://docs.docker.com/install/linux/linux-postinstall/).

If you are installing Traffic Director on Google Cloud VMs, some setup tasks require that you have access to a Linux host. The host can be a local machine or a VM running on your Virtual Private Cloud network.

To configure Traffic Director, you must be able to create instances and modify a network in a project. If you are the project owner or editor (/iam/docs/understanding-roles#primitive_roles) in the project where you are enabling Traffic Director, you automatically have the correct permissions.

Otherwise, you must have all of the following Compute Engine IAM roles (/compute/docs/access/iam) and permissions:

Task	Required Role and Permissions
Set IAM policy for a service account	iam.serviceAccounts.setIamPolicy (/sdk/gcloud/reference/iam/service-accounts/set-iam-policy) permission with the iam.serviceAccountAdmin (/compute/docs/access/iam#iam.serviceAccountAdmin) role This permission is also included in the Service Usage Admin (/iam/docs/understanding-roles#service-usage-roles) role. compute.globalForwardingRules.get (/compute/docs/reference/rest/v1/globalForwardingRules/get) permission with the compute.networkViewer (/iam/docs/understanding-roles#compute-engine-roles) role
Enable Traffic Director	serviceusage.services.enable

	(/service-usage/docs/reference/rest/v1/services/enable) on the project This permission is included in the Service Usage Admin role.
Create networks, subnets, and load balancer components	Network Admin (/compute/docs/access/iam#compute.networkAdmin)
Add and remove firewall rules	Security Admin (/compute/docs/access/iam#compute.securityAdmin)
Create instances	Instance Admin (/compute/docs/access/iam#compute.instanceAdmin)

In addition, if you are configuring Traffic Director with a Google Kubernetes Engine cluster, the GKE node-pool must have the https://www.googleapis.com/auth/cloud-platform (https://www.googleapis.com/auth/cloud-platform) scope.

When the sidecar proxy connects to the xDS-server (`trafficdirector.googleapis.com`), the proxy uses the service account of the Compute Engine VM host or of the GKE node instance. Unless you modify the configuration, Google Cloud uses the Compute Engine default service account (/compute/docs/access/service-accounts#compute_engine_default_service_account). The service account must have the `compute.globalForwardingRules.get` project-level IAM permission.

To enable this permission, assign the role `compute.networkViewer` to the service account.

- If you use the default service account for your VM or GKE cluster nodes, you can use the procedure below to assign the `compute.networkViewer` role.
- If you use a non-default service account, replace the `${SERVICE_ACCOUNT_EMAIL}` variable with the correct email address.
- Alternatively, you can create a custom role that has the `compute.globalForwardingRules.get` permission.

To set up Traffic Director, use one of the following procedures, depending on whether your microservices run on Compute Engine VMs or Kubernetes pods:

- Setting up Traffic Director for Compute Engine with VMs (/traffic-director/docs/set-up-gce-vms)
- Setting up Traffic Director for Google Kubernetes Engine with pods (/traffic-director/docs/set-up-gke-pods)