

This page describes configuring a virtual machine (VM) instance with multiple network interfaces.

By default, every instance in a VPC network has a single default network interface. Use these instructions to create additional network interfaces. Each interface is attached to a different VPC network, giving that instance access to different VPC networks in Google Cloud Platform (GCP). You cannot attach multiple network interfaces to the same VPC network.

If you do not require additional network interfaces, follow the instructions in [Creating and Starting an Instance](/compute/docs/instances/create-start-instance) (/compute/docs/instances/create-start-instance).

For more information about multiple network interfaces and how they work, see [Multiple Network Interfaces](/vpc/docs/multiple-interfaces-concepts) (/vpc/docs/multiple-interfaces-concepts).

- You can only configure a network interface when you create an instance.
- Each network interface configured in a single instance must be attached to a different VPC network, and each interface must belong to a subnet whose IP range does not overlap with the subnets of any other interfaces.
- The additional VPC networks that the multiple interfaces will attach to must exist before you create the instance. See [Using VPC Networks](/vpc/docs/using-vpc) (/vpc/docs/using-vpc) for instructions on creating additional VPC networks.
- You cannot delete a network interface without deleting the instance.
- Each instance must have at least one network interface. The maximum number of network interfaces per instance is 8, depending on the instance's machine type.
 - `nic0` is required and must be configured at instance creation.
 - `nic1` to `nic7` are optional, but must also be configured at instance creation.
 - For more information, see [Maximum number of interfaces](#) (#max-interfaces).
- Every interface can optionally have an external IP address.

- GCP's DHCP server sends a default route (RFC 3442, "classless static route") only to the default NIC, `nic0`. If a default route is needed on another NIC, you must configure it in the instance. While changing the default route in the instance, use the [serial console](/compute/docs/instances/interacting-with-serial-console) (`/compute/docs/instances/interacting-with-serial-console`) in order to avoid losing connectivity to the VM.
- You can use multiple network interfaces with [Shared VPC](/vpc/docs/shared-vpc) (`/vpc/docs/shared-vpc`) networks. A VM can connect to a shared network in a different project only on the default network interface. As a result, if the VM exists in a service project, it can connect to the shared network only on the default interface, while all other interfaces must connect to local networks in the same service project. If the VM exists in the host project, it can have network interfaces connected to any network in the host project.
- You cannot add network interfaces to, or remove them from, an existing VM.

To create an instance with multiple network interfaces, you must have one of the following roles:

- Project owner or editor role
- `compute.instanceAdmin.v1` role

Creating and deleting instances and instance templates with multiple interfaces in a regular projects (a project not using Shared VPC): A user with the OWNER, EDITOR, or

`compute.instanceAdmin.v1` role at the project level can create an instance with multiple interfaces associated with VPC networks and subnets that are part of that same project.

Creating and deleting instances and instance templates with multiple interfaces in [Shared VPC](/vpc/docs/shared-vpc)

environments: A user with the OWNER, EDITOR, or `compute.instanceAdmin.v1` role at the project level can create an instance with multiple interfaces. If any of the interfaces is associated with a subnet in a Shared VPC host project, you must also have the `compute.networkUser` role at the Shared VPC host project level, or at the Shared VPC subnet level.

To learn more about permissions, read the [Compute Engine IAM documentation](/compute/docs/access/iam) (`/compute/docs/access/iam`)

- Multiple network interfaces are not supported with legacy networks.
- When you configure multiple network interfaces, attach each interface to an auto mode VPC network or custom mode VPC network.
- You must configure each network interface with a private primary IP address allocated from a subnet.
 - The private primary address allocated to each of the interfaces in a single instance must be different.
- You can optionally configure each virtual interface with its own public IP address, and such public IP address can be ephemeral or reserved.
- The maximum number of virtual interfaces allowed per instance is 8.
- The number of interfaces allowed in an instance is dependent on the instance's machine type and the number of vCPUs.

In general, the following rules apply for n1-standard-x, n1-highmem-x, n1-highcpu-x, f1-micro, g1-small and custom VMs:

- 2 network interfaces for VM \leq 2vCPU
- 1 network interface per vCPU for VM $>$ 2vCPUs, with a cap of max 8 network interfaces per VM

Use the following table to determine how many network interfaces can be attached to an instance:

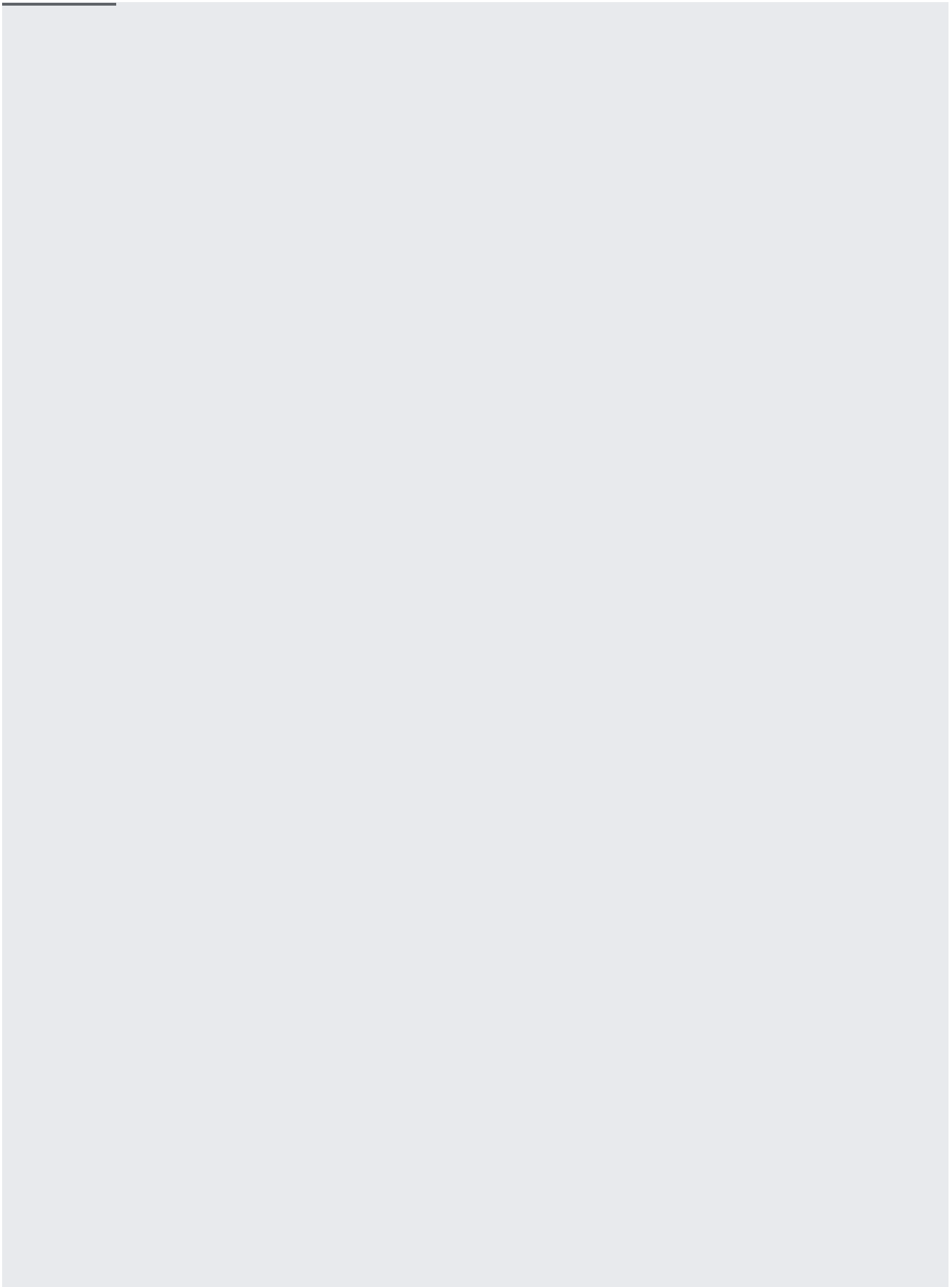
| Type of instance | Number of vCPU | Number of virtual NICs |
|---|-------------------------------------|--|
| n1-standard-x, n1-highmem-x, n1-highcpu-x, n1-highmem-x, n1-highcpu-x | As specified by the value of "x". | 1 NIC per value of "x", with minimum of 2 NICs and a maximum of 8 NICs |
| f1-micro | 0.2 vCPU | 2 NICs |
| g1-small | 0.5 vCPU | 2 NICs |
| Custom Instances | Configurable from 1 vCPU to 64 vCPU | 1 NIC per vCPU, with minimum of 2 NICs and a maximum of 8 NICs |

Creating an instance with a single interface is unchanged. For general instructions on creating instances, see [Creating and Starting an Instance](/compute/docs/instances/create-start-instance) (/compute/docs/instances/create-start-instance).

You add multiple network interfaces when you create an instance.

- On the Console, you add network interfaces on the Create an Instance page, in the Networking > Network interfaces section.
- On the gcloud command line tool, use the `instances create` command. Include the `--network-interface` flag for each interface, followed by any appropriate networking keys, such as (`[network | subnet]`, `private-network-ip`, `address`).

The first interface is always created as `nic0` and it is always the default interface. This is important for some other aspects of GCP networking. For example, Google Cloud load balancers (except for [Internal TCP/UDP Load Balancing](/load-balancing/docs/internal/) (/load-balancing/docs/internal/)) only distribute traffic to `nic0`.



Sample command for creating an instance with three network interfaces:

The interfaces are created as follows:

- `nic0` with all-default settings. Associated with default VPC network, with an automatically allocated private IP address and an ephemeral public IP address
- `nic1` belongs to `net1-subnet-a`, with an internal IP address of `10.128.0.2` and a static external IP address of `35.237.93.110`.
- `nic2` belongs to `net2-subnet-b`, with an internal IP address of `10.129.0.2` and no public IP address

For a complete description of the `gcloud compute instances create` command and the `--network-interface` flag, read the [documentation for the command](#) (`/sdk/gcloud/reference/compute/instances/create`).

All of the instances within a [managed instance group](#) (`/compute/docs/instance-groups/creating-groups-of-managed-instances/`) are restricted to a single subnet.

All of the instances within an unmanaged instance group are restricted to a single subnet when they were used for load balancing. See the [VPC networks \(/vpc/docs/vpc\)](/vpc/docs/vpc) and [load balancing \(/load-balancing/docs/\)](/load-balancing/docs/) documentation for more information.

Because of these restrictions, there is only one interface per instance. GCP performs validations based on the subnet associated with the only interface in the instance.

When you configure multiple interfaces for instances in a managed or unmanaged instance group, GCP continues to validate these conditions for the default `nic0` interface.

The process for creating an instance template that uses a single interface per instance is unchanged and is described in the [gcloud reference for the `instance-templates create` command \(/sdk/gcloud/reference/compute/instance-templates/create\)](/sdk/gcloud/reference/compute/instance-templates/create).

To configure multiple network interfaces for an instance template, set the `--network-interface` flag for each network interface you want to create:

You can assign each interface the following networking keys: `network`, `subnet`, `address`.

For example, if you run the following command, each instance will have three network interfaces:

The interfaces are as follows:

- `nic0` belongs to `net0-subnet-a` and has an automatically allocated public IP address
- `nic1` belongs to `net1-subnet-b` and does not have a public IP address

- `nic2` belongs to `net2-subnet-c` and does not have a public IP address

For complete information on the flags and syntax, see [instance-templates create command](/sdk/gcloud/reference/compute/instance-templates/create) (`/sdk/gcloud/reference/compute/instance-templates/create`).

For Google supported images, when you need a secondary network interface (an interface other than `nic0`) to communicate with any IP address not local to the primary subnet range of that secondary interface's associated subnet, you need to configure policy routing to ensure that egress packets will leave through the correct interface. In such cases, you must configure a separate routing table for each network interface using policy routing.

Source-based policy routing is not supported by Windows operating systems.

Follow these steps to configure policy routing for a Linux-based instance with multiple interfaces:

1. Connect to an instance configured with multiple network interfaces:
2. Configure policy routing with `ifconfig` for `nic1`. The example below assumes that GCP has assigned the internal IP address `192.168.0.2` to `nic1` and the subnet default gateway is `192.168.0.1`.
3. Repeat the commands in step 2 for additional interfaces on the instance (`nic2`, `nic3`.... `nic7`).

You may see one of the following error messages:

- **Invalid value for field 'resource': '' . Too many network interfaces. The maximum number of network interfaces allowed for this machine type is.**

If you get this error message, you are trying to create more interfaces than the maximum interfaces supported by your instance machine type. See the [maximum number of interfaces \(#max-interfaces\)](#) table.

- **Networks must be distinct for NICs attached to a VM.**

If you get this message, you are trying to create more than one interface in the same network. Each network interface must be attached to a different VPC network.

- **Subnetwork CIDR ranges must be non-overlapping for NICs attached to a VM.**

If you get this message, the CIDR ranges associated with your VM's interfaces are overlapping. These CIDR ranges include all primary ranges of subnets associated with VM interfaces as well as secondary ranges used for [alias IP ranges \(/vpc/docs/alias-ip\)](#). Each interface belongs to a subnet, each in a different VPC network, that must not overlap with subnets of other interfaces. For example, if you are trying to create your instance in region `us-west1`, you can check subnet primary CIDR ranges using the following command or the [Google Cloud Console \(https://console.cloud.google.com/networking/networks/list\)](#).

To check secondary subnet CIDR ranges, use the following command or the [Google Cloud Console \(https://console.cloud.google.com/networking/networks/list\)](#).

★ **Note:** secondary ranges present in a subnet associated with VM interface but not used by the VM interface are not checked for overlap.

- **Multiple network interfaces are not supported on legacy networks.**

If you get this message, you are trying to create your instance in a legacy network (/vpc/docs/legacy). Instances with multiple interfaces are not supported with legacy networks. You can check if a network is legacy network using the following command or the Google Cloud Console (<https://console.cloud.google.com/networking/networks/list>). The **Mode** field indicates the type of network.

- **Required 'compute.instances.create' permission for 'projects/PROJECT_ID/zones/ZONE/instances/test-inst'**

If you get this message, the account you are logged in with does not have IAM permissions necessary for creating an instance. Refer to IAM Permissions (#permissions_iam_and_instances_with_multiple_network_interfaces) for detailed information on the roles required for creating instances. You can check if the IAM policy associated with your project grants you any of the following roles: **OWNER**, **EDITOR** or **compute.instanceAdmin.v1**. For creating instances in Shared VPC you would additionally need the **compute.networkUser** role. In the below example, the account **email2@gmail.com** doesn't have sufficient IAM permissions to create an instance. For more detailed instructions see the Granting, Changing, and Revoking Access to Resources (/iam/docs/granting-changing-revoking-access#grant_access) IAM guide.

You can ask your project owner or editor to either grant you a role of `OWNER`, `EDITOR`, or `compute.instanceAdmin.v1`. You would need `compute.networkUser` role as well if you are associating any interface with a subnet belonging to Shared VPC.

- Check firewall rules to see that they allow connectivity to the VM secondary interface. You can check by looking at the firewall rules for the network that is attached to the secondary interface by either visiting [Google Cloud Console](https://console.cloud.google.com/networking/networks/list) (<https://console.cloud.google.com/networking/networks/list>) and clicking the appropriate VPC network or by using the following `gcloud` command.
- Check if you are trying to connect to a secondary interface from an Internet address or from outside of the secondary interface's network. You can only connect to an interface's internal IP from within its network. If you need to reach that interface from outside the network, you can assign an external IP address to your secondary interface.
- Check if you are trying to connect to the secondary interface's internal IP from outside the subnet the secondary interface is attached to, either from another subnet of the same network or from a [peered network](/vpc/docs/vpc-peering) (</vpc/docs/vpc-peering>). [Multiple Network Interfaces per instance](/vpc/docs/vpc-peering#multiple_network_interfaces_per_instance) (/vpc/docs/vpc-peering#multiple_network_interfaces_per_instance) explains the interplay between VPC Peering and VM instances with multiple interfaces. In order to reach secondary interfaces from outside the interface's subnet, you might have to configure routes on the VM. See [DHCP behavior with multiple network interfaces](/vpc/docs/multiple-interfaces-concepts#dhcp_behavior_with_multiple_network_interfaces) (/vpc/docs/multiple-interfaces-concepts#dhcp_behavior_with_multiple_network_interfaces) for details on how DHCP programs default routes in the VM.

The DHCP server programs a default route only on the primary network interface of the VM. If you want to connect to the secondary interface using an external IP, there are two options. If you only need to connect outside the network on the secondary network interface, you can set a default route on that network interface. Otherwise, you can use [Configuring Policy Routing](#) (#configuring_policy_routing) to configure a separate routing table using source-based policy routing in your VM.

By default, the instance [metadata server](#) (/compute/docs/storing-retrieving-metadata) only responds to ARP requests for the default gateway.

To configure interfaces with a netmask other than /32, you should create an image using the flag `--guest-os-features MULTI_IP_SUBNET` and use it to create your instance. For example, if you are using a `debian-9` based image, you can create an image using following command:

To view the guest features configured on the image, run the `gcloud compute images describe` command on the guest image.

For more information on creating custom images, see [Creating, Deleting, and Deprecating Custom Images](#) (/compute/docs/images/create-delete-deprecate-private-images).

It is often a good idea to enable serial console on the VM to debug configuration-related issues. You can enable serial console for interactive debugging by following steps in [interacting with serial console](#) (/compute/docs/instances/interacting-with-serial-console).

