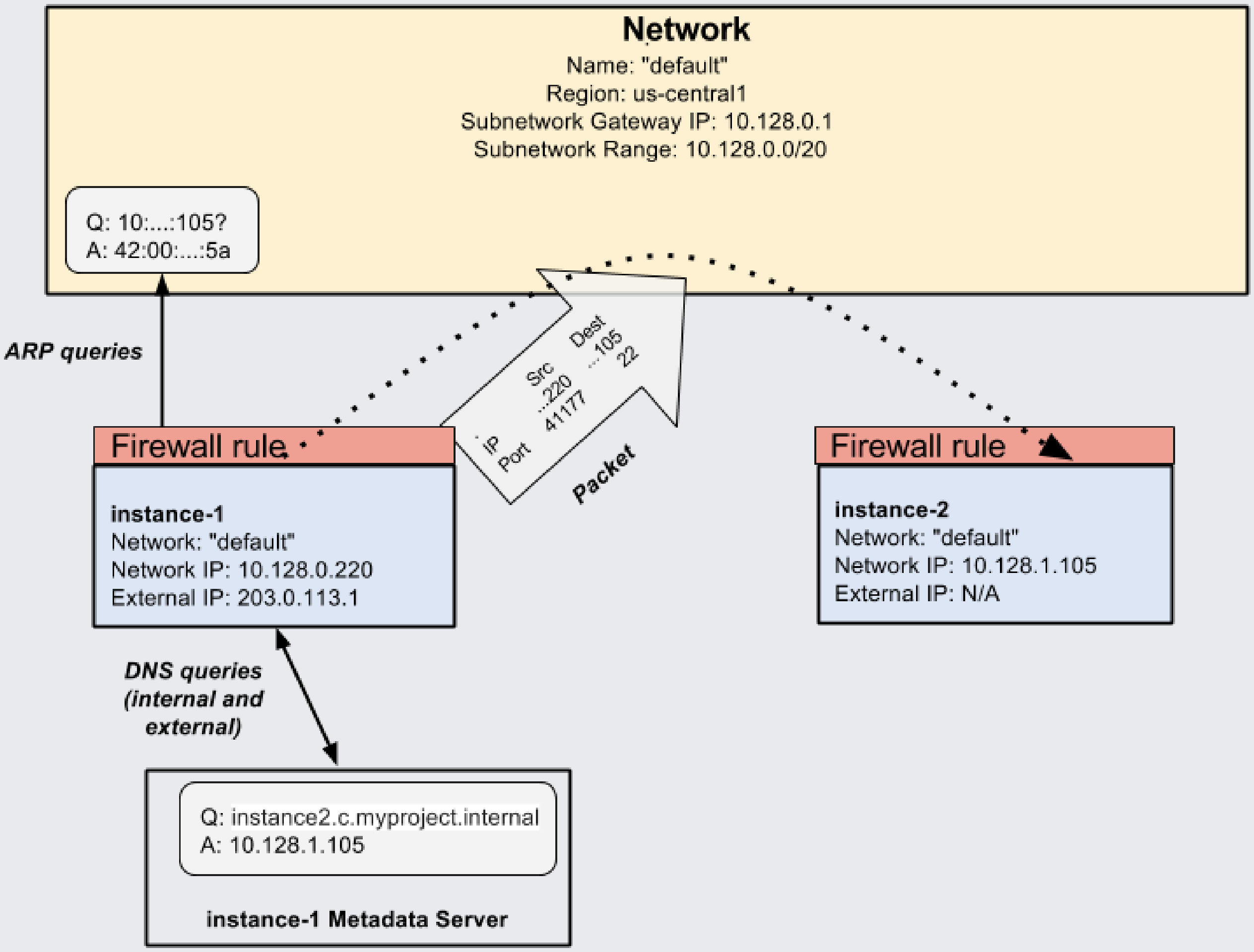
This page goes into additional detail about Virtual Private Cloud (VPC) networks. Before reading this page, see VPC network overview (/vpc/docs/vpc).

This section provides some low-level VPC networking details. You do not need to read this for typical usage, but it provides more insight about how VPC networking works. The following diagram describes these low-level details, with more information in the corresponding sections.



Under the hood, different VPC networking features are handled by different parts of the system. Some of these are standard networking features that are well documented, and some of them are specific to VPC networks. Some features you can configure, and some you cannot. VPC networks use Linux's VIRTIO network module (http://dl.acm.org/citation.cfm?id=1400097.1400108) to model Ethernet card and router functionality, but higher levels of the networking stack, such as ARP lookups, are handled using standard networking software.

ARP lookup

The instance kernel issues ARP requests (https://wikipedia.org/wiki/Address_Resolution_Protocol) and the VPC network issues ARP replies. The mapping between MAC addresses and IP addresses is handled by the instance kernel.

MAC lookup table, IP lookup table, active connection table

These tables are hosted on the underlying VPC network and cannot be inspected or configured.

DNS server

Each instance's metadata server acts as a DNS server. It stores the DNS entries for all VPC network IP addresses in the local VPC network and calls Google's public DNS server for entries outside the VPC network. You cannot configure this DNS server. The DHCP client on each instance is configured to manage the instance's `/etc/resolv.conf` file.

You can add your own search domain or nameservers to the instance's `/etc/resolv.conf` by modifying the DHCP policy. Many Linux distributions allow these modifications to persist via `/etc/dhcp/dhclient.conf` (https://www.isc.org/wp-content/uploads/2017/08/dhcp41clientconf.html). See the Internal DNS (/compute/docs/internal-dns) documentation for more information.

Packet handling between the VPC network and the outside

Packets coming into or out of the VPC network are handled by network code that examines the packet against firewall rules, against the external IP lookup table, and against the active connections table. The VPC network also performs NAT on packets coming into and out of the VPC network.

Packets received by an instance

These packets are received and turned into a stream by the instance kernel in the standard fashion.

Packets sent by an instance

Packets are sent by the instance kernel in the standard way. Interface and network functionality are modeled using the VIRTIO network module (http://dl.acm.org/citation.cfm?id=1400097.1400108).

Here are more details about what happens when an instance makes a VPC network call.

An instance makes a call:

1. If the target address is an instance name or a URL such as www.google.com, the instance calls the DNS service on its metadata server and gets back the matching IP address. You can configure your instance to consult another DNS service, although then you will not be able to resolve instance names.
2. The destination IP address is examined against the subnet's IP address range, which every instance knows.
    a. If the IP address is outside the VPC network:
        i. The instance sends the packet to the subnet's gateway MAC address with the destination set to the packet's final destination. The instance might need to make an ARP request to resolve the gateway's MAC address.
        ii. The VPC network rewrites the IP header to declare the instance's external IP address as the source. If the instance has no external IP address, the call is not allowed, and the VPC network drops the packet without informing the sender.
        iii. The VPC network records the outgoing packet and adds the source and destination to the active connections table.
        iv. The VPC network sends the packet on to its destination.
        v. The destination gets the packet and responds if it chooses.

vi. The VPC network receives the response, consults the active connections table, notes that this is an active connection, and allows it. The VPC network consults its network/external IP lookup table and replaces the instance's external IP address with the matching network address and sends the packet to the source instance.

vii. The instance receives the packet.

b. If the destination IP address is within the VPC network:

i. The instance is configured with an IP with 255.255.255.255 mask, so the instance sends the packet to the subnet's gateway MAC address. The instance first might need to make an ARP request to resolve the gateway's MAC address.

ii. The VPC network, using Proxy ARP (https://wikipedia.org/wiki/Proxy_ARP), responds with the MAC address of the destination instance.

iii. Google Cloud forwards the packet to the destination IP within the VPC network.

iv. The target instance receives the packet. The target instance checks ingress firewall to determine if the packet is allowed. If not, the packet is dropped silently. Otherwise, the instance processes the packet.

An external instance or computer calls an instance:

1. The external caller sends a packet to an instance's external IP address, which is owned by the VPC network.
2. The VPC network compares the packet against the active connections table to see whether this is an existing connection:

   a. If it is not an existing connection, the VPC network looks for a firewall rule to allow the connection.

   b. If there is no firewall rule, the VPC network drops the packet without informing the sender.
3. If there is an existing connection or valid firewall rule, the VPC network examines its lookup table and replaces the external IP with the corresponding internal IP in the packet, logs the incoming packet in the active connections table, and sends the packet to the target instance.

4. The instance receives the packet and responds as described in **If the IP address is outside the VPC network IP range** (#ip_outside_network) when sending a packet outside the network range.
5. The VPC network receives the reply, finds the matching incoming request in the active connections table, and allows the packet through. Before sending, it modifies the source IP address by replacing the instance's internal IP with the corresponding external IP from its lookup table.

Outbound (egress) traffic from a virtual machine is subject to a per-VM egress throughput cap (/compute/docs/machine-types#network_bandwidth). The cap is a limit that can't be exceeded and doesn't indicate the actual throughput of your egress traffic. There is no guarantee that your traffic will achieve the maximum throughput, which depends on many factors other than the cap. For example, using external IP addresses to communicate between VM instances requires more overhead than using internal IP addresses. Consequently, the maximum throughput for traffic that is using external IP addresses has a lower maximum throughput than traffic that is using internal IP addresses.

To measure an instance's performance in relation to these caps, use the PerfKitBenchMarker (https://github.com/GoogleCloudPlatform/PerfKitBenchmarker) to measure the egress throughput performance of your instances.

For example, run the following commands while you are on a local computer. The command will create an instance and measure its performance, where:

- `[MACHINE_TYPE]` is the machine type you want to test (for example, n1-standard-32).
- `[ZONE]` is the zone to create the instance in.
- `[NUMBER_OF_VCPUS]` is the number of vCPUs of the instance (for example, 32 for n1-standard-32 machine type).

To measure single stream performance:

To measure multistream performance:

- To learn about VPC networks, see VPC network overview (/vpc/docs/vpc).
- To create, modify, and delete VPC networks, see Using VPC networks (/vpc/docs/using-vpc).