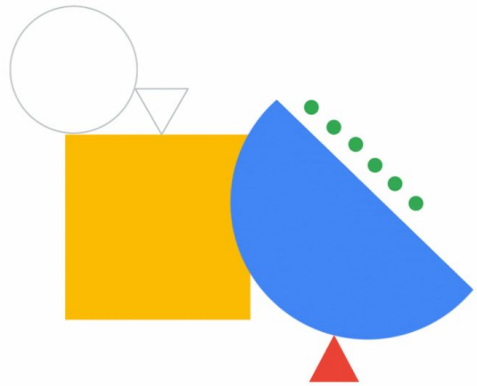# Monitoring Google Cloud Network

In this module, let's spend some time analyzing the Google Cloud Virtual Private Cloud.

Objectives

01 Collect and analyze VPC Flow Logs, load balancer logs, Firewall Rule Logs, and Cloud NAT logs.

02 Enable and monitor Packet Mirroring.

03 Explain the capabilities of the Network Intelligence Center.

Google Cloud

Specifically, you learn to:

- Collect and analyze VPC Flow Logs, Firewall Rule Logs, load balancer logs, and Cloud NAT logs so you can see what's happening to the traffic across your network.
- Enable Packet Mirroring so you can replicate packets at the virtual machine network interface, and forward it for further analysis.
- And explain the capabilities of the Network Intelligence Center.

In this section, you explore

- ✓ **VPC Flow Logs**
- ✓ Firewall Rules Logging
- ✓ Load balancer logs
- ✓ Cloud NAT logs
- ✓ Packet Mirroring
- ✓ Network Intelligence Center

Google Cloud

Let's start with monitoring the network.

VPC Flow Logs is used to monitor network
by **recording a portion** of network flows
sent and received by VM instances
(including GKE nodes).

VPC Flow Logs records a sample (about one out of ten packets) of network flows sent from and received by VM instances, including Google Kubernetes Engine nodes. These logs can be used for network monitoring, traffic analysis, forensics, real-time security analysis, and expense optimization.

VPC Flow Logs is part of Andromeda, the software that powers VPC networks. VPC Flow Logs introduces no delay or performance penalty when enabled.

VPC Flow Logs example

VM to external traffic flow

Project

VPC network

Region: us-west 1

Subnet: 10.10.0.0/26

10.10.0.2

Egress flow

Ingress flow
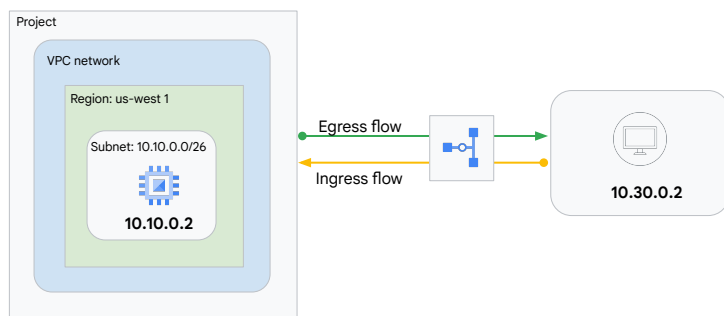
10.30.0.2

Flow logs reported from the VM:

Egress flow: 10.10.0.2 to 10.30.0.2

Ingress flow: 10.30.0.2 to 10.10.0.2

The VM and and on-premises endpoint are connected through a Cloud VPN
or Cloud Interconnect instance

Google Cloud

This slide shows an example of a VM to external flow traffic pattern. VPC Flow logs
provides visibility into traffic, which helps monitor the flow between zones, IP
addresses, network diagnosis.

In this example, the traffic flows between a VM and an external network connected
either through a Cloud VPN or Cloud Interconnect.

Traffic flows are reported from the VM only and include:
- Igness traffic: the logs reported with VM as its destination. In this example, the
  ingress traffic is reported from the source VM 10.30.0.2 to 10.10.0.2
- Egress traffic: the logs reported with VM as its source. In this example, the
  egress traffic is reported from the source VM 10.10.0.2 to 10.30.0.2

VPC Flow Logs properties

- Samples are from the VM's perspective.
  - An egress deny firewall is logged.
  - An ingress deny firewall rule is not logged.
- Samples are logged for each VMs. This includes inbound and outbound traffic:
  - VM to VM, VM to a host on the internet, VM to on-premises host, etc.

Blocked egress flow is logged.

Blocked ingress flow is not logged.

Google Cloud

These are some of the main properties you must remember when working with VPC Flow Logs:
- VPC Flow Log samples are from a VM's perspective. For this reason, if an egress firewall is denied, those packets are sampled by VPC Flow Logs. Similarly, the ingress blocked packets are not logged because they are sampled after the ingress firewall rules.
- VPC Flow Logs samples TCP, UDP, ICMP, ESP and GRE flows from each VM. It records inbound and outbound flows for each VM, thus capturing traffic between VM's, VM to on-premises, VM to another host on the internet.
- VMs support multiple network interface and can be enabled at the subnet level.

Enable VPC Flow Logs per VPC subnet

✓ VPC Flow Logs is activated or deactivated at a subnet level.

● All VMs within that subnet have VPC Flow Logs automatically enabled.

✓ You can enable VPC Flow Logs during subnet creation.

● You can optionally adjust log sampling and aggregation to adjust the metadata and sample rate written to logs.

Google Cloud

You can activate or deactivate VPC Flow Logs per VPC subnet. When enabled for a subnet, VPC Flow Logs collects data from all VM instances in that subnet.

To enable VPC Flow Logs, during subnet creation, select **On** next to Flow Logs. You can optionally adjust log sampling and aggregation to adjust the metadata and sample rate that is written to logs.

Log entries contain many useful fields

Field	Type	Description
src_ip	string	Source IP address
src_port	int32	Source port
dest_ip	string	Destination IP address
dest_port	int32	Destination port
protocol	int32	IANA protocol number

Google Cloud

Each log entry contains a record of different fields. For example, this table illustrates the IP connection information that is recorded. Information consists of the source IP address and port, the destination IP address and port, and the protocol number. This set is commonly referred to as 5-tuple.

Other fields include the start and end time of the first and last observed packet, the bytes and packets sent, instance details including network tags, VPC details, and geographic details. For more information on all data recorded by VPC Flow Logs, see the documentation.

Use Logging to analyze your VPC Flow Logs
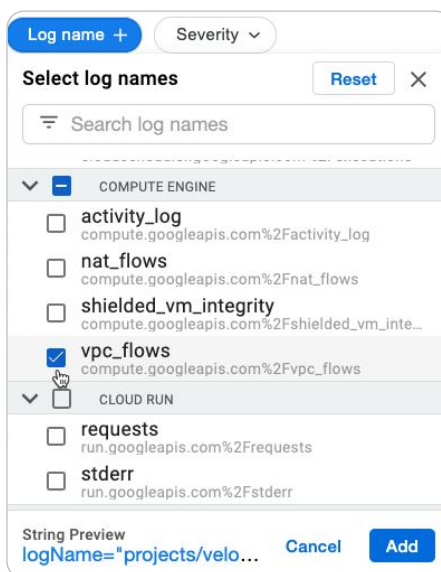
Logs Explorer can be used to access the VPC Flow Logs. The entries will be vpc_flows below the Compute Engine section. Searching the log names for *vpc_flows* works well.

For a log query, use:
logName="projects/[PROJECT_ID]/logs/compute.googleapis.com%2Fvpc_flows"

Analyze logs with Log Analytics

Log Analytics powered by BigQuery provides new capabilities to analyze log data and generate useful insights. With Log Analytics:
- You can analyze ad-hoc query-time without complex pre-processing as before.
- You can use BigQuery to query data and upgrade buckets to use Log Analytics and then create a linked dataset.

Here is a curated sample queries to get started with Flow Log Analysis.

In this section, you explore

- ✓ VPC Flow Logs
- ✓ **Firewall Rules Logging**
- ✓ load balancer logs
- ✓ Cloud NAT logs
- ✓ Packet Mirroring
- ✓ Network Intelligence Center

Google Cloud

Another essential part of knowing what's happening at the VPC network level is knowing what the firewall rules are doing.

Firewall Rules Logging

Did my firewall rules cause that application outage?

How many connections match the rule I just created?

Are my firewall rules stopping (or allowing) the correct traffic?

Google Cloud

VPC firewall rules let you allow or deny connections to or from your virtual machine (VM) instances based on a configuration that you specify.

Enabled VPC firewall rules are always enforced, and protect your instances regardless of their configuration and operating system, even if they didn't start.

Firewall Rules Logging lets you audit, verify, and analyze the effects of your firewall rules.
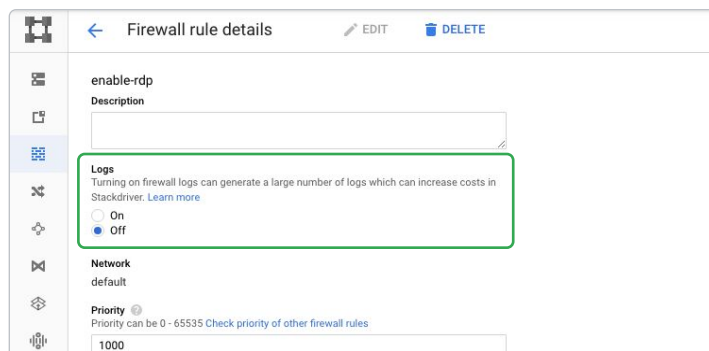
It can help answer questions like:

- Did my firewall rules cause that application outage?
- How many connections match the rule I just created?
- Are my firewall rules stopping (or allowing) the correct traffic?

See the Firewall Rule Logging [documentation](#) for details.

Enabling Firewall Rules Logging in the console

- Firewall Rules Logging is **disabled** by default.
- You enable it on a per-rule basis.

Firewall rule details   ✏ EDIT   🗑 DELETE

enable-rdp

Description

Logs
Turning on firewall logs can generate a large number of logs which can increase costs in Stackdriver. Learn more
○ On
● Off

Network
default

Priority ⓘ
Priority can be 0 - 65535 Check priority of other firewall rules
1000

Note: Firewall Rules Logging can only record TCP and UDP connections. For other protocols, use Packet Mirroring.

Google Cloud

By default, Firewall Rules Logging is disabled.

You can enable it on a per-rule basis. In the slide screenshot, you're editing the firewall rule named *enable-rdp*. Selecting the radio button will enable firewall rules.

Note: Firewall Rules Logging can only record TCP and UDP connections. For other protocols, use Packet Mirroring.

Caution: Firewall Rules Logging can generate lots of data, which might have a cost implication.

Enabling Firewall Rules Logging in the CLI

Firewall Rules Logging can also be enabled or disabled by using the following **gcloud** CLI commands. Substitute [NAME] for the name of your firewall rule.

```
Activate:
gcloud compute firewall-rules update [NAME] --enable-logging

Deactivate:
gcloud compute firewall-rules update [NAME] --no-enable-logging
```
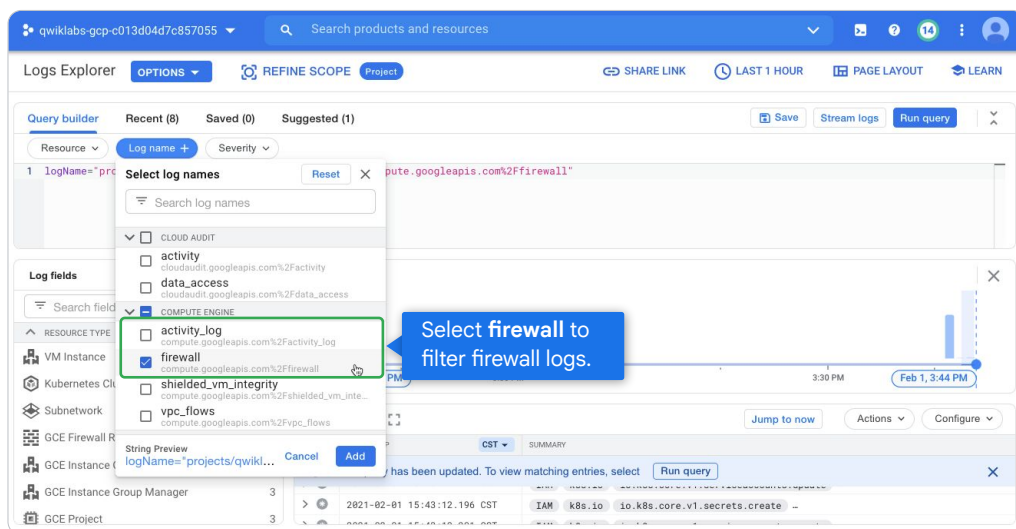
Google Cloud

Firewall Rules Logging can also be activated on existing firewall rules by using the CLI.

See these two examples on this slide. In both, the [NAME] tag will be the name of your firewall rule.

Viewing the Firewall Rules logs

Like all Google Cloud logs, use Logs Explorer to view logs in real time or to configure exports.

To filter for firewall logs and network policy firewall logs, below the Compute Engine resource, select **firewall**.

Firewall rules provide microsegmentation

SubNetwork

Webserver 1　　Appserver　　Webserver 2

Segmentation or gateway-centric

Google Cloud
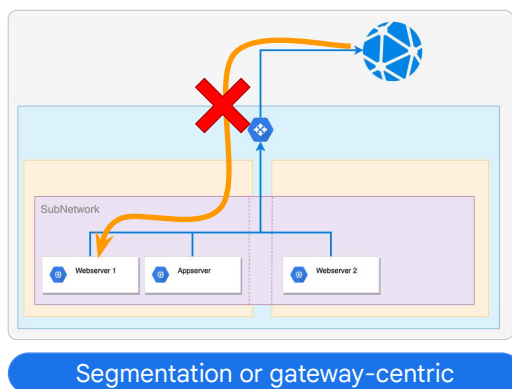
Many are familiar with classic segmentation or gateway-centric firewalls.

In this example, you can see a private network, possibly at your office or home.

At the network boundary, where the private network meets the outside internet, sits a firewall.

A segmentation firewall is designed to segment and secure a protected network from an outside insecure network.

Firewall rules provide microsegmentation

Segmentation or gateway-centric

Microsegmentation or VM-centric
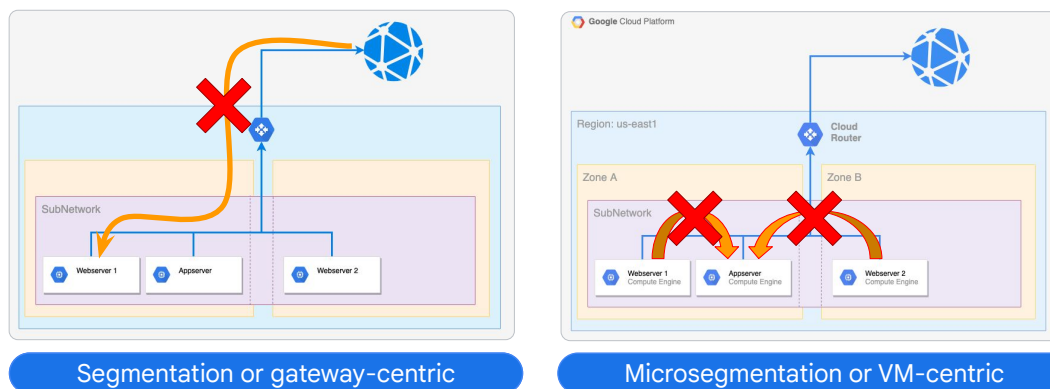
Google Cloud VPC firewalls are micro-segmentation firewalls.

These firewalls function more like a bunch of micro-firewalls, each operating over the Network Interface Controller (NIC) of every VM connected to the VPC.

The micro-firewalls can then grant or deny any configured incoming or outgoing traffic.

Now, imagine we have an issue.

We have two different web servers. After some configuration changes by a particular DevOps team, the web servers can no longer access the application server they both share.

How can we tell if the issue is firewall-related? Let's see.

Troubleshooting: Using rules to catch incorrect traffic

- Logging all denied connections will create too many log entries.
- Temporarily create a high-priority rule (low-priority value) to allow traffic to the server. Enable Cloud logging.
- If traffic now gets through, examine the logs to find the root cause.

Firewall logs
(of allowed traffic)

Cloud Logging
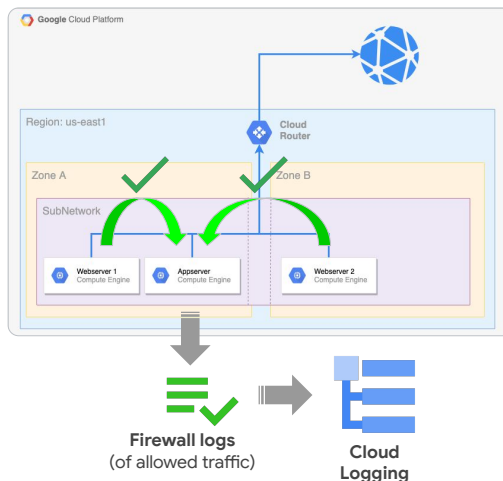
Google Cloud

If the connectivity issue is related to a firewall, then there are two major possibilities:

1) A firewall rule is actively blocking the incoming connections from the web servers.

Or

2) Network traffic is blocked by default in most networks. A firewall rule might not be allowing the traffic from the web servers as it should.

Logging all denied connections could generate significant data that would take time and effort to monitor. So, instead of starting with option one, start with option two.

Create a temporary high-priority rule designed to allow the web server traffic through to the app server. Enable Cloud logging on it so you can examine the entries.

Suddenly the traffic is getting through, so you know it's firewall related. Now examine the log entries. Also, find the existing rule supposed to be allowing the traffic and see what you can find.

Hey, look at that! The rule that's supposed to allow the traffic is based on a network tag named *webserver.* The web server machines are actually using the network tag *web-server*.

There it is, that's your problem.

In this section, you explore

- ✓ VPC Flow Logs
- ✓ Firewall Rules Logging
- ✓ **Load balancer logs**
- ✓ Cloud NAT logs
- ✓ Packet Mirroring
- ✓ Network Intelligence Center

Google Cloud

Several of Google Cloud load balancers support monitoring or logging.

Load Balancers support for Cloud Logging

- All the Google Cloud load balancers support Cloud Logging and Cloud Monitoring:
  - Internal and External HTTP(S) Load Balancers
  - Internal and External TCP/UDP Network Load Balancers
  - External SSL Proxy and TCP Proxy Load Balancers
  - Internal TCP Proxy Load Balancers
- The log type, log fields, and metrics supported vary based on the the load balancers type.
- Load balancing logs is used to **debug** and **analyze** user traffic.

Google Cloud

While all the Google Cloud load balancers support Cloud Logging and Cloud Monitoring, the log type and log fields supported vary based on the type of the load balancers. These include:

- Internal and external HTTP(S) load balancers
- Internal and External TCP/UDP Network load balancers
- External SSL Proxy and TCP Proxy Load balancers
- Internal TCP Proxy load balancers

Cloud Logging for load balancing logs all the load balancing requests sent to your load balancer. These logs can be used for debugging and analyzing your user traffic. You can view request logs and export them to Cloud Storage, BigQuery, or Pub/Sub for analysis. For example, in network load balancer, per-connection logging gives you insight into how each connection is routed to serving backends.

The internal and external HTTP(S) load balancers support logging

Activated and deactivated on a per backend service basis

For external HTTP(S) load balancers with backend buckets, logging is automatically enabled and cannot be deactivated.

Logging can be enabled on a per backend service basis.

URL map might reference more than one backend service.

Use exclusion, if you do not want the logs to be stored in Cloud Logging.

Google Cloud

For external HTTP(S) load balancers with backend buckets, logging is automatically enabled and cannot be disabled. You can activate logging on a per backend service basis. A single internal HTTP(S) load balancer URL map can reference more than one backend service. You might need to enable logging for more than one backend service, depending on your configuration. It will be enabled by default for all new load balancers backends. But backends created before the Globally Available (GA) release of load balancer logging might require manual configuration.

Fields in a log record

- `LogEntry`: Contains severity, project ID, project number, and timestamp information.
- `httpRequest`: Contains a method, URL, status, remote IP address, latency string.
- `resource`: Contains the monitored resource associated with a log entry.
- `jsonPayload`: Contains `statusDetails field` that includes a string that explains why the load balancer returned the HTTP status, cache, and failure information.
  - **Redirects** (HTTP response status code 302 found) issued from the load balancer are not logged. Redirects issued from the backend instances are logged.

Google Cloud

HTTP(S) load balancing log entries contain information useful for monitoring and debugging your HTTP(S) traffic. Make sure to check the documentation for details.

Log entries contain the following types of information:

- LogEntry format includes general information shown in most logs, such as severity, project ID, project number, timestamp, and so on.
- HttpRequest log fields. However, *HttpRequest.protocol* is not populated for HTTP(S) load balancing logs. This can include a method, a URL, remote IP address, a protocol, a latency string or a user agent.
- resource contains the monitored resource type associated with the log entry.
- jsonPayload contains the *statusDetails* field. This field holds a string that explains why the load balancer returned the HTTP status that it did.
- Redirects such as HTTP response status code 302 Found) issued from the load balancer are *not* logged. Redirects issued from the backend instances *are* logged.

Example of using a load balancing log record

- **Error:** Load balancer generates an `HTTP error response code (5XX)`.
- Refer to Load balancer logs to determine the source of error.
  - **`statusDetails field: response_sent_by_backend`** field indicates it is a backend issue.
  - **`statusDetails field: failed_to_pick_backend`** field indictes that the load balancer failed to pick a healthy backend to handle a request.

Google Cloud

Let's take an example of how to use log record information to troubleshoot a load balancing issue. Consider a scenario where the load balancer generates an HTTP error resource code 5XX and sends the same error code to the client.

Refer to the load balancer logs to determine the source of an error:
- statusDetails field: the response_sent_by_backend indicates it is a backend issue.
- statusDetails field: failed_to_pick_backend indicted that the load balancer failed to pick a healthy backend to handle a request.
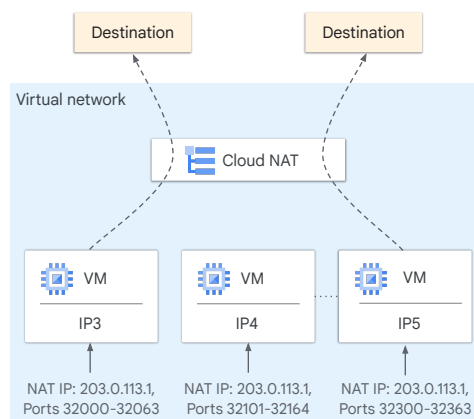
In this section, you explore

- ✓ VPC Flow Logs
- ✓ Firewall Rules Logging
- ✓ load balancer logs
- ✓ Cloud NAT logs
- ✓ Packet Mirroring
- ✓ Network Intelligence Center

Google Cloud

Another piece of the network telemetry features in Google Cloud is Cloud NAT logs.

Cloud NAT overview

- It allows Google Cloud Compute workload with no external IP to send packets to the internet.

- It's a fully managed, proxyless NAT service in Andromeda.

- These are some of its benefits:
  - Reduces the need for individual VMs to each have external IP addresses.
  - Automatically scales the number of NAT IP addresses that it uses.
  - Is not dependent on a single physical gateway device.

Destination   Destination

Virtual network

Cloud NAT

VM	VM	VM
IP3	IP4	IP5

NAT IP: 203.0.113.1, Ports 32000-32063   NAT IP: 203.0.113.1, Ports 32101-32164   NAT IP: 203.0.113.1, Ports 32300-32363

Google Cloud

Cloud NAT is the Google-managed Network Address Translation service. It lets you provision your application instances without public IP addresses, and it also lets them access the internet in a controlled and efficient manner. With Cloud NAT, your private instances can access the internet for updates, patching, configuration management, and more.

Cloud NAT benefits include:

- With Cloud NAT, VMs without external IP addresses can access destinations on the internet. For example, you might have VMs that only need internet access to download updates or complete provisioning. Cloud NAT lets you configure these VMs with an internal IP address. Thus, your organization needs fewer external IP addresses.
- Cloud NAT can be configured to automatically scale the number of NAT IP addresses that it uses. Cloud NAT supports VMs that belong to managed instance groups, including those with autoscaling enabled.
- Cloud NAT is not dependent on a single, physical gateway device. Cloud NAT is a distributed, software-defined managed service. You configure a NAT gateway on a Cloud Router, which provides the control plane for Cloud NAT. Cloud Router contains the NAT configuration parameters. Google Cloud runs and maintains processes on the physical machines that run your Google Cloud VMs.

Cloud NAT logging

- A NAT log is created when:
  - A network connection using NAT is created.
  - A packet is dropped due to port unavailability
- It lets you log NAT **connections** and/or **errors.**
  - TCP and UDP traffic only.
  - 50-100 entries per second, per vCPU.

Google Cloud

Cloud NAT logging lets you log NAT TCP and UDP connections and errors. When Cloud NAT logging is enabled, a log entry can be generated when a network connection that uses Cloud NAT is created, and/or when an egress packet is dropped because no port was available for Cloud NAT.

You can opt to log both kinds of events, or just one or the other. Logs contain TCP and UDP traffic only, and the log rate threshold will reach a maximum of 50-100 log events per vCPU before log filtering.

Cloud NAT logging might be enabled when a new Cloud NAT gateway is first created, or by editing the settings of an existing gateway.

To view the collected logs in Logs Explorer, filter to the Cloud NAT Gateway resource and optionally, restrict to a particular region or Gateway.

A full query will look like:

resource.type="nat_gateway"
logName="projects/{#project_id}/logs/compute.googleapis.com%2Fnat_flows"

In this section, you explore

- VPC Flow Logs
- Firewall Rules Logging
- load balancer logs
- Cloud NAT logs
- Packet Mirroring
- Network Intelligence Center

Another way to monitor the network traffic flowing in and out of your Compute Engine virtual machines is to use Packet Mirroring.

Packet Mirroring: Visualize and protect your network

- Clones VPC instance traffic and forwards for examination.

- Packet Mirroring policies are tied to workloads and not VPC.
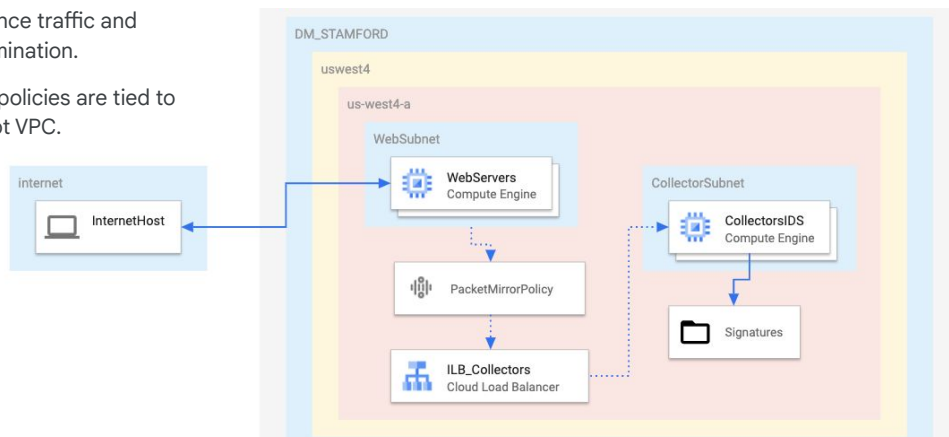
Packet Mirroring clones the traffic of specific instances in your Virtual Private Cloud (VPC) network and forwards it for examination. Packet Mirroring captures all ingress and egress traffic and packet data, such as payloads and headers.

The mirroring happens on the virtual machine (VM) instances, not on the network. Therefore, Packet Mirroring consumes additional bandwidth on the hosts.

Packet Mirroring is useful when you need to monitor and analyze your security status. It exports all traffic, not only the traffic between sampling periods. For example, you can use security software that analyzes mirrored traffic to detect all threats or anomalies.

Also, you can inspect the full traffic flow to detect application performance issues and to provide network forensics for Payment Card Industry Data Security Standards (PCI DSS) compliance and other regulatory use cases. We will elaborate on this further in the next few slides.

Obviously, Packet Mirroring can generate significant data, so collector destination is generally an instance group behind a TCP/UDP load balancer or equivalent technology.

Packet Mirroring: Overcoming bandwidth limitations

- Packet Mirroring consumes the egress bandwidth of the mirrored instances.
  - Use filters to reduce the bandwidth on mirrored instances.
  - Filters can be based on protocol, IP ranges, traffic directions, etc.
  - The current maximum of filters for Packet Mirroring is 30.

Google Cloud

One of the major limitations of Packet Mirroring is bandwidth consumption. Packet Mirroring consumes the egress bandwidth of the mirrored instances. However, there is a work around. Use filters to reduce the traffic collected for mirrored instances. This filter can be used for IP address ranges, protocols, traffic directions and lot more.

The current maximum number of filters that can be used for Packet mirroring is 30.

For more information, refer to the documentation.

Packet Mirroring: Use cases

- Network and application **monitoring**:
    - Maintain integrity of deployment.
    - Fix packet loss issues.
    - Fix reconnection and latency issues.
- **Security and compliance**:
    - Intrusion detection systems.
    - Deep Packet Inspection engines.
- **Forensics:**
    - Collect, process, and preserve network forensics.

Two main use cases where Packet Mirroring is useful in security and monitoring. Let's explore each of these use cases in detail.

Network and application monitoring
Network engineers can use the data from Packet Mirroring to:
- Maintain integrity of deployment.
- Troubleshoot packet loss issues by analyzing protocols.
- Troubleshoot reconnection and latency issues by analyzing real time traffic patterns.

Security and compliance
Implement zero-trust by monitoring network traffic across and within the trust boundaries without any network re-architecture. Packet Mirroring helps capture multiple packets for a single flow. This information can be quite useful for the implementation and usage of the following security tools:
- Intrusion detection systems match signatures with multiple packets of a single flow.
- Deep Packet Inspection engines inspect payloads for anomalies.

Network forensics for PCI compliance: Packet mirroring help capture, process and preserve forensic of different attack vectors.

Monitoring Packet Mirroring

- Metrics can verify that instances are being monitored as intended.
    - `Mirrored Packets count`
    - `Mirrored Bytes count`
    - `Dropped Packets count`
- Monitoring can also spot where packet mirroring shouldn't be happening.
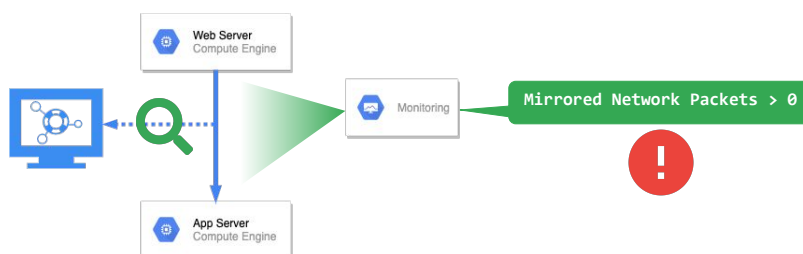
Google Cloud

- Packet Mirroring exports monitoring data about mirrored traffic to Cloud Monitoring. You can use monitoring metrics to check whether traffic from a VM instance is being mirrored as intended. For example, you can view the mirrored packet or byte count for a specific instance.

- You can view the monitoring metrics of mirrored VM instances or instances that are part of the collector destination (internal load balancer). For mirrored VM instances, Packet Mirroring provides metrics specific to mirrored packets, such as /mirroring/mirrored_packets_count, /mirroring/mirrored_bytes_count, and /mirroring/dropped_packets_count.

- Monitoring can also spot where packet mirroring is being used unnecessarily or unexpectedly. Remember that, as noted earlier, mirroring generates significant data that requires storage and processing. Also, note that it slows the network throughput of the virtual machines being monitored and might accidentally expose sensitive data.

In this section, you explore

- ✓ VPC Flow Logs
- ✓ Firewall Rules Logging
- ✓ load balancer logs
- ✓ Cloud NAT logs
- ✓ Packet Mirroring
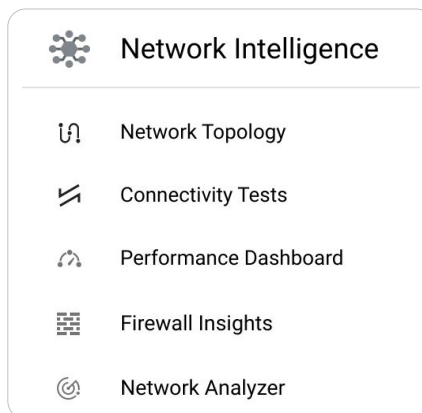- ✓ Network Intelligence Center

Google Cloud

This section is a bit of a detour, but let's at least mention the Network Intelligence Center and how it helps with network analysis.

Network Intelligence Center

Centralized network monitoring and visibility

- **Network Topology:** View VPC topology and associated metrics.
- **Connectivity Tests:** Evaluate connectivity to and from VPC resources.
- **Performance Dashboard:** Provides VPC packet loss and latency metrics.
- **Firewall Insights:** Provides visibility into firewall usage and configuration issues.
- **Network Analyzer:** View network and service issues, insights, and recommendations.

	Network Intelligence
⟳	Network Topology
⤸	Connectivity Tests
⌢	Performance Dashboard
▦	Firewall Insights
◎	Network Analyzer

Google Cloud

Network Intelligence Center gives you centralized monitoring and visibility into your network. Network Intelligence Center reduces troubleshooting time and effort and increases network security, all while improving the overall user experience. Currently, it offers five modules: Network Topology, Connectivity Tests, Performance Dashboard, Firewall Insights, and Network Analyzer.

- **Network Topology** is a visualization tool for viewing the topology of your VPC networks and the metrics that are associated with their Google Cloud resources.

- **Connectivity Tests** enables you to evaluate connectivity to and from Google Cloud resources in your Virtual Private Cloud (VPC) network by performing a static analysis of your resource configurations.

- **Performance Dashboard** gives you visibility into the performance of your VPC network. It provides packet loss and latency (round trip time) metrics between the zones where you have VMs.

- **Firewall Insights** provides visibility into firewall usage and detects firewall configuration issues.

- **Network Analyzer** automatically monitors your VPC network configurations and detects misconfigurations and suboptimal configurations. Network Analyzer identifies failures caused by the underlying network, provides root

- cause information, and suggests possible resolutions.
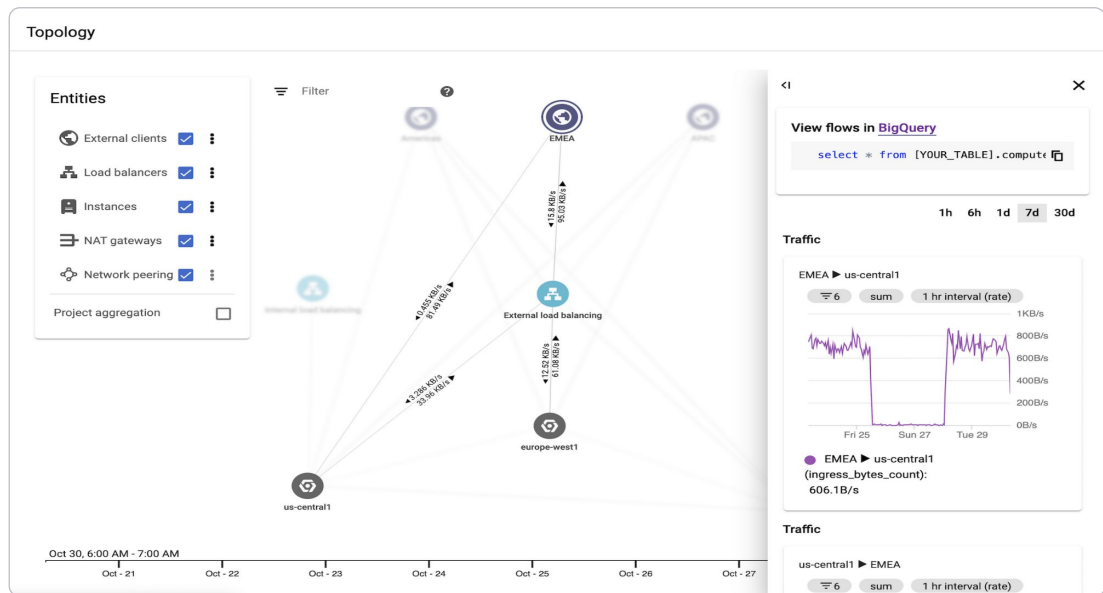
Network Topology visualizes your Google Cloud network as a graph.

You can use the graph to explore your existing configurations and quickly troubleshoot networking issues.

You can select network entities, filter, see lines of communication with bandwidth information, expand and collapse hierarchies, and select time boundaries. You can also see details for the item selected.

Connectivity Tests

Connectivity Tests	➕ CREATE CONNECTIVITY TEST	↻ RERUN	🗑 DELETE

This test lets you check connectivity between network endpoints. It analyzes your configuration and, if the configuration is eligible, sends packets through the live data plane. Learn more ⬈

⇆ Filter Filter by test name or protocol

	Name	Protocol	Source	Destination	Destination port	Last test time
☐	http	tcp	10.150.0.3 (default)	10.150.0.2 (default)	80	2023-05-16 (14:04:09)
☐	test	tcp	10.0.0.1 (default)	10.1.1.1 (default)	80	2023-05-16 (13:31:27)
☐	vm-test1	icmp	grafana-ent (default, 10.150.0.3)	ray (default, 10.150.0.2)	-	2023-05-16 (14:19:15)

- Quickly diagnose connectivity issues and prevent outages.
- Verify the configuration change effect to help prevent outages.

The Connectivity Tests tool in Network Intelligence Center helps you to quickly diagnose connectivity issues and prevent outages.

These tests let you self-diagnose connectivity issues within Google Cloud or from Google Cloud to an external IP address (the connectivity issue could be on-premises or in another cloud). The results help to isolate whether the issue is in Google Cloud.

Run tests to help verify the effect of configuration changes and ensure that network intent captured by these tests is not violated, proactively preventing network outages.

These tests also help assure network security and compliance.

Performance Dashboard

Packet loss metrics aggregated across zones
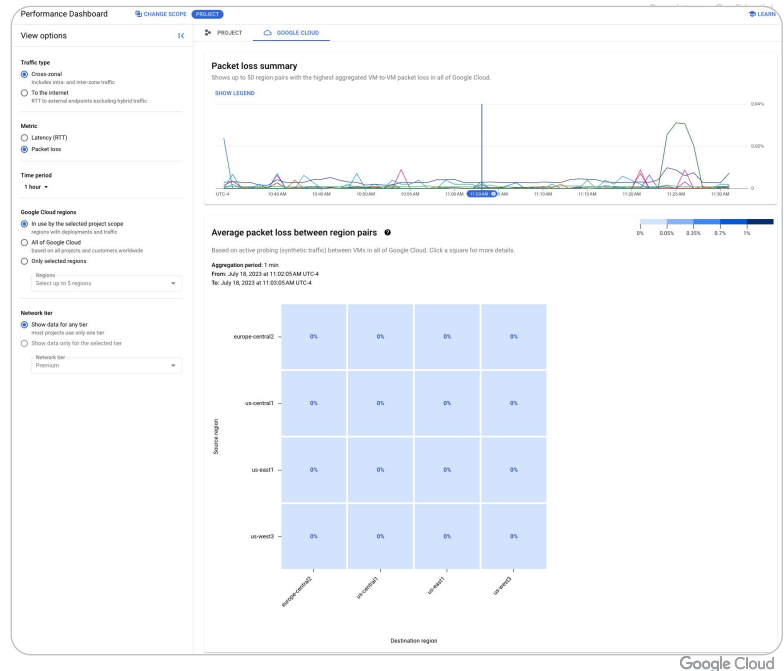
Performance Dashboard gives you visibility into the performance of your VPC.

The **Packet Loss** tab shows the results of active probing between your VMs in a given VPC.

To get this data, it runs workers on the physical hosts that house your VMs.

These workers insert and receive probe packets that run on the same network as your traffic, revealing issues on that network.

Workers run on the physical host and not on your VM. Therefore, these workers do not consume VM resources and the traffic is not visible on your VMs.

Packet loss is aggregated for all zone pairs.

The **Latency** tab aggregates latency information based on a sample of your actual Transmission Control Protocol (TCP) VM traffic. The method used is similar to the one used for VPC Flow Logs.

The latency is calculated as the time that elapses between sending a TCP sequence number (SEQ) and receiving a corresponding Acknowledgement (ACK) that contains the network Round Trip Time (RTT) and TCP stack related delay.

The latency metric is only available if TCP traffic is around 1,000 packets per minute or higher.

Firewall Insights helps you better understand and safely optimize your firewall rules

Cloud Monitoring - Metrics

Network Intelligence Center Firewall Insights

Recommender - Insights

Monitoring
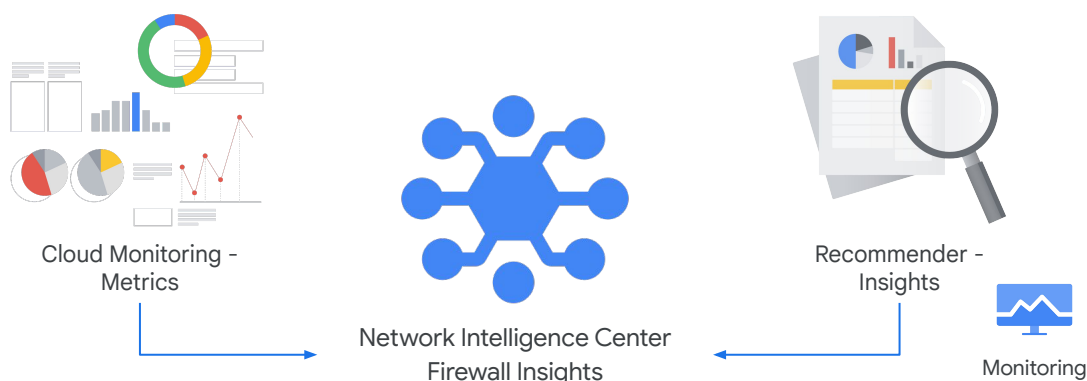
Google Cloud

Firewall Insights, a component product of Network Intelligence Center, produces metrics and insights that let you make better decisions about your firewall rules. It provides data about how your firewall rules are being used, exposes misconfigurations, and identifies rules that could be made more strict.

Firewall Insights uses Cloud Monitoring metrics and Recommender insights.

Cloud Monitoring collects measurements to help you understand how your applications and system services are performing. A collection of these measurements is generically called a metric. The applications and system services being monitored are called monitored resources. Measurements might include the latency of requests to a service, the amount of disk space available on a machine, the number of tables in your SQL database, the number of widgets sold, and so forth. Resources might include virtual machines, database instances, disks, and so forth.

Recommender is a service that provides recommendations and insights for using resources on Google Cloud. These recommendations and insights are per-product or per-service, and are generated based on heuristic methods, machine learning, and current resource usage. You can use insights independently from recommendations. Each insight has a specific insight type. Insight types are specific to a single Google Cloud product and resource type. A single product can have multiple insight types, where each provides a different type of insight for a different resource.

Using Cloud Monitoring for metrics:

https://cloud.google.com/monitoring/api/v3/metrics
Using Recommender for insights:
https://cloud.google.com/recommender/docs/insights/using-insights

Firewall Insights let you analyze how your firewall rules are used

✓ Verify that firewall rules are being used in the intended way.

✓ Verify that firewall rules allow or block their intended connections.

✓ Perform live debugging of connections that are inadvertently dropped.

✓ Discover malicious attempts to access your network.

Google Cloud

Firewall Insights metrics let you analyze the way that your firewall rules are being used. Firewall Insights metrics are available through Cloud Monitoring and the Google Cloud console. Metrics are derived through Firewall Rules Logging.

With Firewall Insights metrics, you can perform the following tasks:

- Verify that firewall rules are being used in the intended way.
- Over specified time periods, verify that firewall rules allow or block their intended connections.
- Perform live debugging of connections that are inadvertently dropped because of firewall rules.
- Discover malicious attempts to access your network, in part by getting alerts about significant changes in the hit counts of firewall rules.

Firewall Insights provides analysis about the firewall rule configuration

✓ Identify firewall misconfigurations.

✓ Identify security attacks.

✓ Optimize firewall rules and tighten security boundaries.

Firewall Insights provides analysis about your firewall rule configuration and usage of your firewall rules. They use the `google.compute.firewall.Insight` insight type.

With Firewall Insights, you can perform the following tasks:

- Identify firewall misconfigurations.
- Identify security attacks.
- Optimize firewall rules and tighten security boundaries by identifying overly permissive allow rules and reviewing predictions about their future usage. At the time of writing, these capabilities are in preview.

Network Analyzer

- Automatically monitors your VPC network configurations and detects misconfigurations and suboptimal configurations.
- Provides insights on network topology, firewall rules, routes, configuration dependencies, and connectivity to services and applications.
- Identifies network failures, provides root cause information, and suggests possible resolutions.

Google Cloud
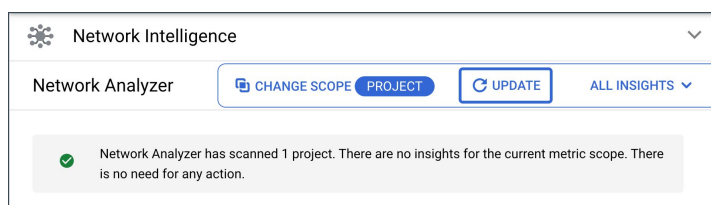
Network Analyzer automatically monitors your VPC network configurations and detects misconfigurations and suboptimal configurations. It provides insights on Network Topology, firewall rules, routes, configuration dependencies, and connectivity to services and applications. It identifies network failures, provides root cause information, and suggests possible resolutions.

Network Analyzer runs continuously and triggers relevant analyses based on near real-time configuration updates in your network. If a network failure is detected, it tries to correlate the failure with recent configuration changes to identify root causes. Wherever possible, it provides recommendations to suggest details on how to fix the issues.

# Insights help surface issues and identify the root cause

Network Analyzer provides insights that help identify common issues such as connectivity blockage, load balancing errors, external IP address that are not used but allocated, invalid next hop, GKE network misconfiguration and lot more. It also identifies the root cause of the insights and also provides recommended fix.

In the example above, an insight of the type *Error*, a GKE node to control plane connectivity is generate. The insight page also describes the following:

- The root cause: an ingress firewall rule is blocking the connection between the node and the plane. This indicated that the default drywall rules were modified, removed, or shadowed by another firewall rule.
- A solution: if the root of the problem is a deleted firewall, create a new firewall rule. If it's a shadowed firewall rule, then increase the priority.

Lab intro

Analyzing Network Traffic with
VPC Flow Logs (optional)

Google Cloud
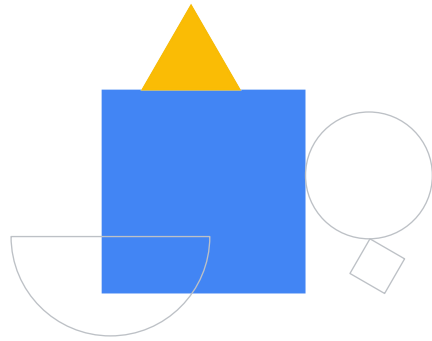
In this lab, you configure a network to record traffic to and from an Apache web server
by using VPC Flow Logs. You then export the logs to BigQuery to analyze them.

Recap

01 Collect and analyze VPC Flow Logs, Firewall Rule Logging, and Cloud NAT logs.

02 Enable and monitor Packet Mirroring.

03 Explain the capabilities of the Network Intelligence Center.

Google Cloud

Very good. After completing this module, you know how to:

- Collect and analyze VPC Flow Logs, Firewall Rules Logging, and Cloud NAT logs so you can see what's happening to the traffic across your network.
- Enable Packet Mirroring so you can replicate packets at the virtual machine network interface and forward it for further analysis.
- And explain the capabilities of the Network Intelligence Center.

Fantastic job!

Google Cloud