Firewall Rules Logging allows you to audit, verify, and analyze the effects of your firewall rules. For example, you can determine if a firewall rule designed to deny traffic is functioning as intended. Firewall Rules Logging is also useful if you need to determine how many connections are affected by a given firewall rule.

You enable Firewall Rules Logging individually for each firewall rule whose connections you need to log. Firewall Rules Logging is an option for any firewall rule, regardless of the action (`allow` or `deny`) or direction (ingress or egress) of the rule.

When you enable logging for a firewall rule, Google Cloud creates an entry called a *connection record* each time the rule allows or denies traffic. You can export these connection records to Stackdriver Logging (/logging), Pub/Sub (/pubsub/), or BigQuery (/bigquery/) for analysis.

Each connection record contains the source and destination IP addresses, the protocol and ports, date and time, and a reference to the firewall rule that applied to the traffic.

For information about viewing logs, see Using Firewall Rules Logging (/vpc/docs/using-firewall-rules-logging).

Firewall Rules Logging has the following specifications:

- You can only enable Firewall Rules Logging for rules in a Virtual Private Cloud (VPC) network (/vpc/docs/vpc). Legacy networks (/vpc/docs/legacy) are *not* supported.
- Firewall Rules Logging only records TCP and UDP connections. Although you can create a firewall rule applicable to other protocols (/vpc/docs/firewalls#protocols_and_ports), you cannot log their connections.
- You *cannot* enable Firewall Rules Logging for the implied deny ingress and implied allow egress rules (/vpc/docs/firewalls#default_firewall_rules).
- Log entries are written from the perspective of virtual machine (VM) instances. Log entries are only created if a firewall rule has logging enabled and if the rule applies to traffic sent to or from the VM. Entries are created according to the connection logging limits on a *best effort* basis.

The maximum number of connections that can be logged per VM instance depends on its machine type (/compute/docs/machine-types). Connection logging limits are expressed as the maximum number of connections that can be logged in a five-second interval.

tant: Firewall log entries are created on a best effort basis according to the following table. It is possible for entries tc
ged during periods of heavy traffic, even if the maximum number of logged connections for a machine type has not b
ed.

Instance machine type	Maximum number of connections logged in a 5-second interval
f1-micro	100 connections
g1-small	250 connections
Machine types with 1–8 vCPUs	500 connections per vCPU
Machine types with more than 8 vCPUs	4,000 (500×8) connections

A log entry is generated each time that a firewall rule with logging enabled applies to traffic. A given packet flow can generate more than one log entry in total. However, from the perspective of a given VM, at most only one log entry can be generated if the firewall rule that applies to it has logging enabled.

The following examples demonstrate how firewall logs work.

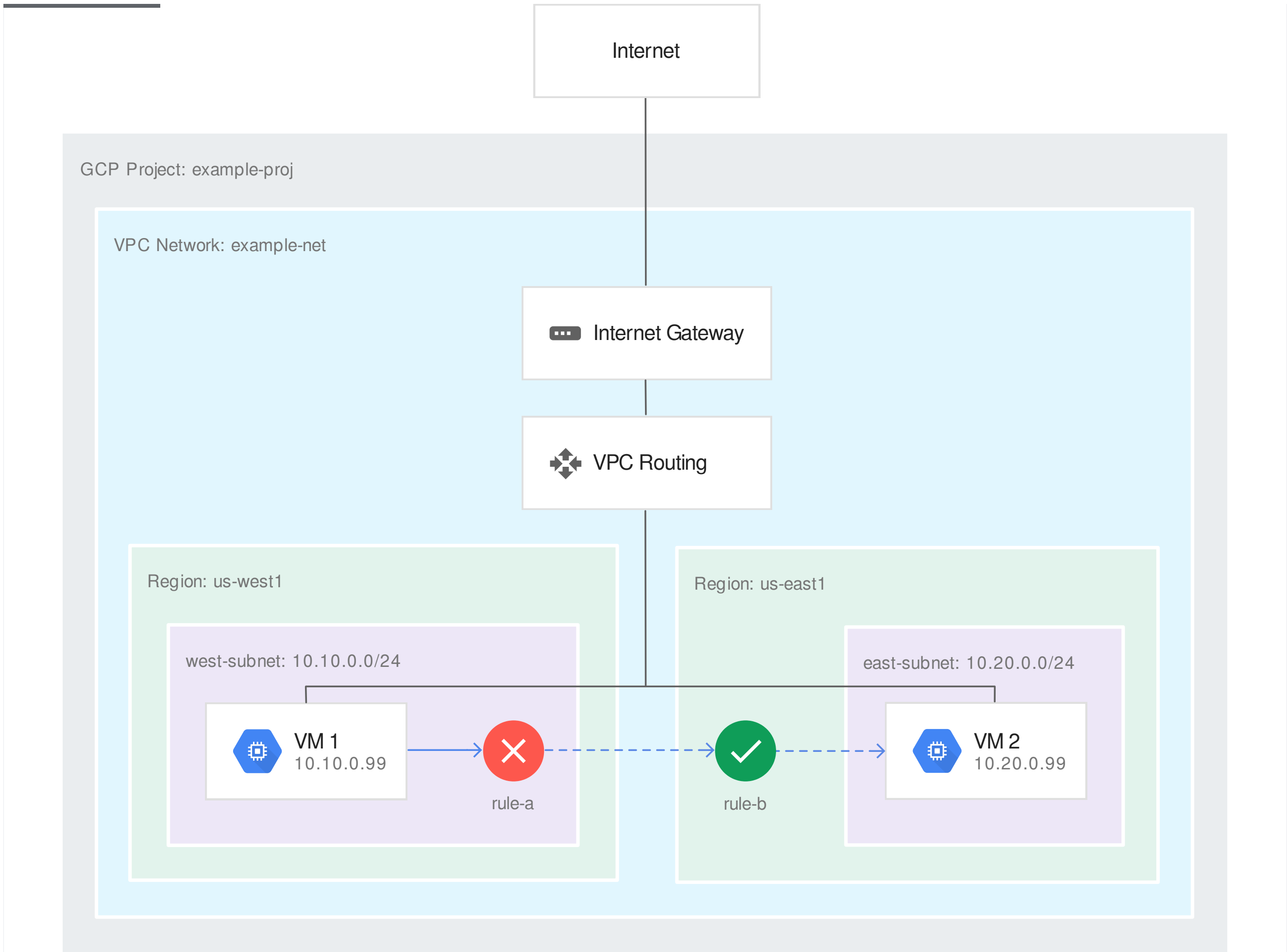
In this example:

- Traffic between VM instances in the `example-net` VPC network in the `example-proj` project is considered.
- The two VM instances are:
  - VM1 in zone `us-west1-a` with IP address `10.10.0.99` in the `west-subnet` (`us-west1` region).
  - VM2 in zone `us-east1-a` with IP address `10.20.0.99` in the `east-subnet` (`us-east1` region).

- Rule A: An egress deny firewall rule has a target of all instances in the network, a destination of `10.20.0.99` (VM2), and applies to TCP port 80.
    - Logging is enabled for this rule.
- Rule B: An ingress allow firewall rule has a target of all instances in the network, a source of `10.10.0.99` (VM1), and applies to TCP port 80.
    - Logging is also enabled for this rule.

The following `gcloud` commands can be used to create the firewall rules:

- Rule A: egress deny rule for TCP 80, applicable to all instances, destination `10.20.0.99`:
- Rule B: ingress allow rule for TCP 80, applicable to all instances, source `10.10.0.99`:



(/vpc/images/firewall-rules-logs/firewall-rules-logging-1.svg)
VM1 to VM2 connection (click to enlarge)

Suppose VM1 attempts to connect to VM2 on TCP port 80. The following firewall rules are logged:

- A log entry for rule A from the perspective of VM1 is generated as VM1 attempts to connect to `10.20.0.99` (VM2).
- Because rule A actually blocks the traffic, rule B is never considered, so there is no log entry for rule B from the perspective of VM2.

The firewall log record is generated in the following example.

Field	Values

connection	src_ip=10.10.0.99
	src_port=[EPHEMERAL_PORT]
	dest_ip=10.20.0.99
	dest_port=80
	protocol=tcp

Field	Values
disposition	DENIED

rule_details	reference = "network:example-net/firewall:rule-a"
	priority = 10
	action = DENY
	destination_range = 10.20.0.99/32
	ip_port_info = tcp:80
	direction = egress

instance	project_id="example-proj"
	instance_name=VM1
	region=us-west1
	zone=us-west1-a

vpc	project_id="example-proj"
	vpc_name=example-net
	subnetwork_name=west-subnet

remote_instance	project_id="example-proj"
	instance_name=VM2
	region=us-east1
	zone=us-east1-a

remote_vpc	project_id="example-proj"
	vpc_name=example-net
	subnetwork_name=east-subnet

remote_location	*No information. This field is only used if the destination is outside your VPC network.*

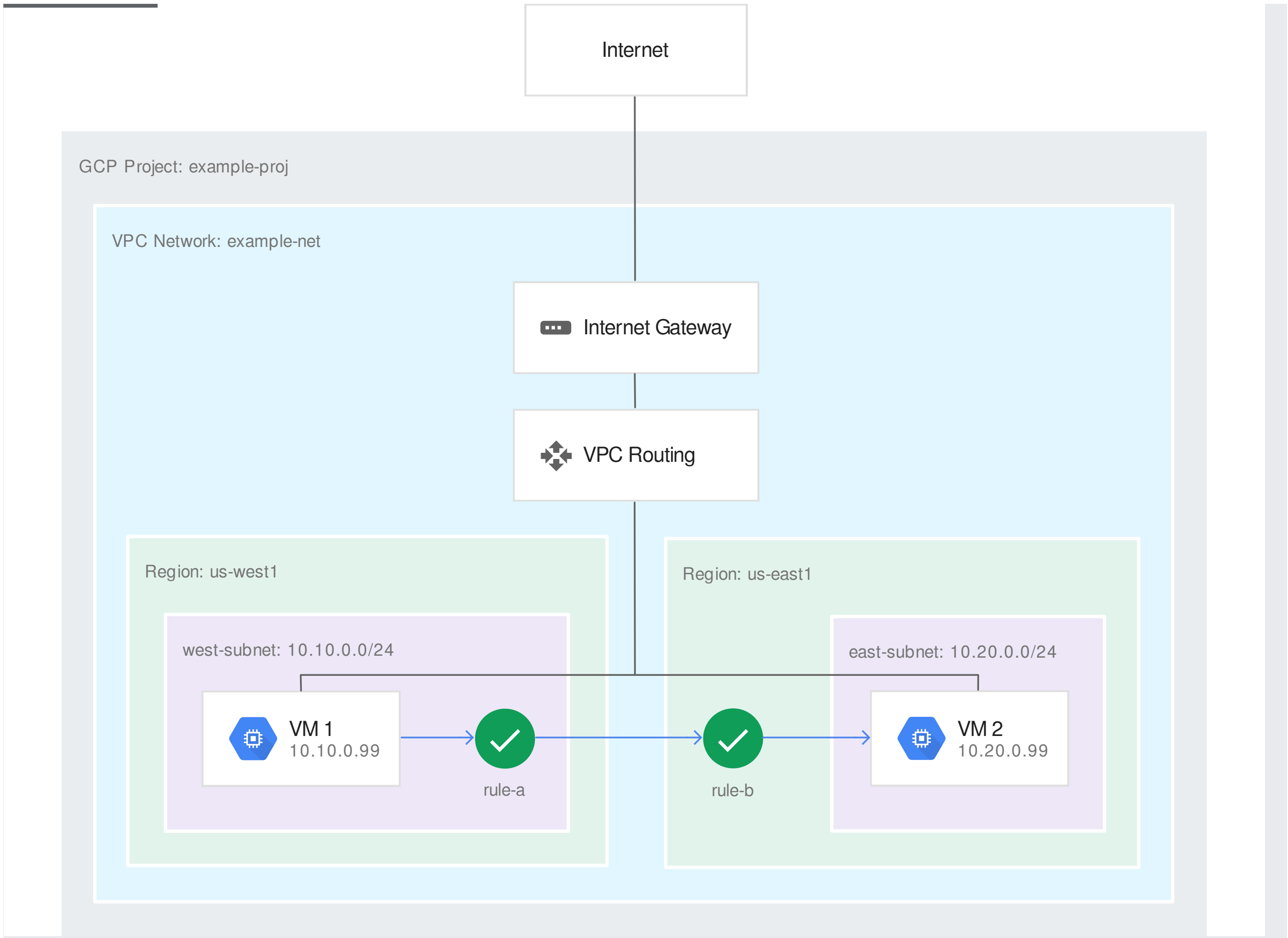
In this example:

- Traffic between VM instances in the `example-net` VPC network in the `example-proj` project is considered.
- The two VM instances are:
    - VM1 in zone `us-west1-a` with IP address `10.10.0.99` in the `west-subnet` (`us-west1` region).
    - VM2 in zone `us-east1-a` with IP address `10.20.0.99` in the `east-subnet` (`us-east1` region).

- Rule A: An egress allow firewall rule has a target of all instances in the network, a destination of `10.20.0.99` (VM2), and applies to TCP port 80.
    - Logging is enabled for this rule.
- Rule B: An ingress allow firewall rule has a target of all instances in the network, a source of `10.10.0.99` (VM1), and applies to TCP port 80.
    - Logging is also enabled for this rule.

The following `gcloud` commands can be used to create the two firewall rules:

- Rule A: egress allow rule for TCP 80, applicable to all instances, destination `10.20.0.99` (VM2):
- Rule B: ingress allow rule for TCP 80, applicable to all instances, source `10.10.0.99` (VM1):



(/vpc/images/firewall-rules-logs/firewall-rules-logging-2.svg)
VM1 to VM2 connection (click to enlarge)

Suppose VM1 attempts to connect to VM2 on TCP port 80. The following firewall rules are logged:

- A log entry for rule A from the perspective of VM1 is generated as VM1 connects to `10.20.0.99` (VM2).
- A log entry for rule B from the perspective of VM2 is generated as VM2 allows incoming connections from `10.10.0.99` (VM1).

The firewall log record reported by VM1 is generated in the following example.

Field	Values

connection	src_ip=10.10.0.99
	src_port=[EPHEMERAL_PORT]
	dest_ip=10.20.0.99
	dest_port=80
	protocol=tcp

Field	Values
disposition	ALLOWED

rule_details	reference = "network:example-net/firewall:rule-a"
	priority = 10
	action = ALLOW
	destination_range = 10.20.0.99/32
	ip_port_info = tcp:80
	direction = egress

instance	project_id="example-proj"
	instance_name=VM1
	region=us-west1
	zone=us-west1-a

vpc	project_id="example-proj"
	vpc_name=example-net
	subnetwork_name=west-subnet

remote_instance	project_id="example-proj"
	instance_name=VM2
	region=us-east1
	zone=us-east1-a

remote_vpc	project_id="example-proj"
	vpc_name=example-net
	subnetwork_name=east-subnet

remote_location	*No information. This field is only used if the destination is outside your VPC network.*

The firewall log record reported by VM2 is generated in the following example.

Field	Values

connection	src_ip=10.10.0.99
	src_port=[EPHEMERAL_PORT]
	dest_ip=10.20.0.99
	dest_port=80
	protocol=tcp

disposition	ALLOWED

Field	Values

rule_details	reference = "network:example-net/firewall:rule-b"
	priority = 10
	action = ALLOW
	source_range = 10.10.0.99/32
	ip_port_info = tcp:80
	direction = ingress

instance	project_id="example-proj"
	instance_name=VM2
	region=us-east1
	zone=us-east1-a

vpc	project_id="example-proj"
	vpc_name=example-net
	subnetwork_name=east-subnet

remote_instance	project_id="example-proj"
	instance_name=VM1
	region=us-west1
	zone=us-west1-a

remote_vpc	project_id="example-proj"
	vpc_name=example-net
	subnetwork_name=west-subnet

remote_location	*No information. This field is only used if the destination is outside your VPC network.*

In this example:

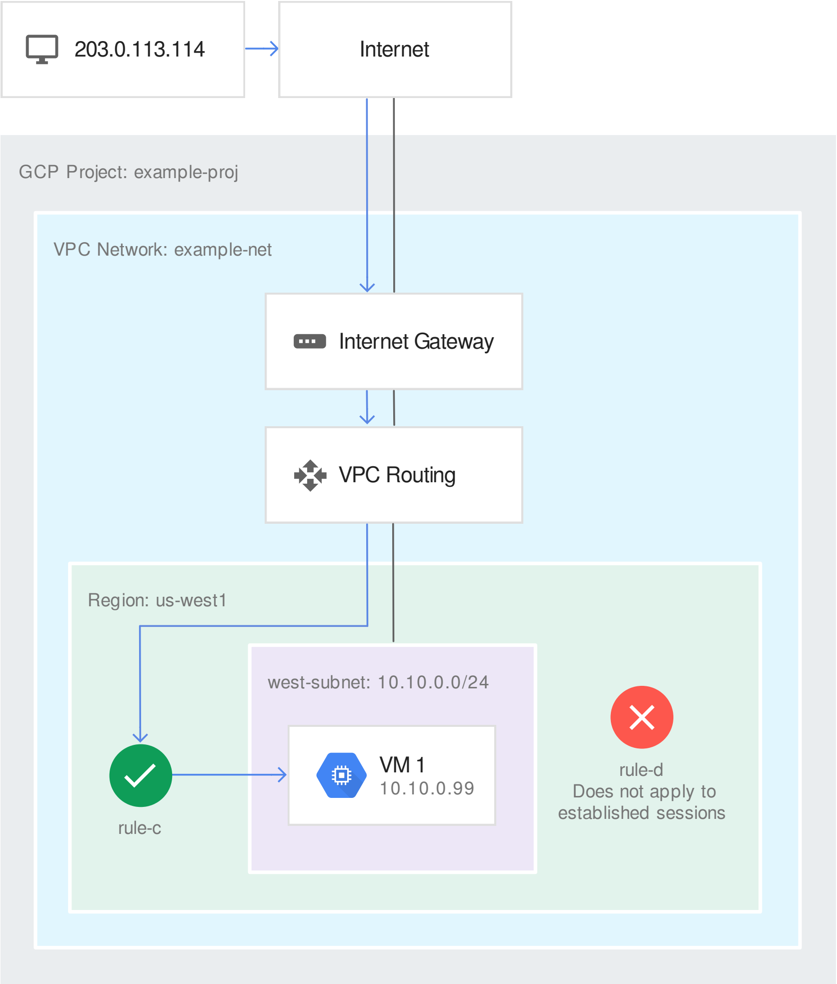
- Traffic from a system outside the `example-net` VPC network to a VM instance in that network is considered. The network is in the `example-proj` project.
- The system on the internet has IP address `203.0.113.114`.
- VM1 in zone `us-west1-a` has IP address `10.10.0.99` in the `west-subnet` (`us-west1` region).
- Rule C: An ingress allow firewall rule has a target of all instances in the network, a source of any IP address (`0.0.0.0/0`), and applies to TCP port 80.
    - Logging is enabled for this rule.
- Rule D: An egress deny firewall rule has a target of all instances in the network, a destination of any IP address (`0.0.0.0/0`), and applies to all protocols.

- Logging is also enabled for this rule.

The following `gcloud` commands can be used to create the firewall rules:

- Rule C: ingress allow rule for TCP 80, applicable to all instances, any source:

- Rule D: egress deny rule for all protocols, applicable to all instances, any destination:



(/vpc/images/firewall-rules-logs/firewall-rules-logging-3.svg)
Internet to VM connection (click to enlarge)

Suppose the system with IP address `203.0.113.114` attempts to connect to VM1 on TCP port 80. The following happens:

- A log entry for rule C from the perspective of VM1 is generated as VM1 accepts traffic from `203.0.113.114`.
- Despite rule D, VM1 is allowed to reply to the incoming request because Google Cloud firewall rules are stateful. If the incoming request is allowed, established responses cannot be blocked by any kind of egress rule.
- Because rule D does not apply, it is never considered, so there is no log entry for rule D.

The firewall log record is generated in the following example.

Field	Values

Field	Values

connection	src_ip=203.0.113.114
	src_port=[EPHEMERAL_PORT]
	dest_ip=10.10.0.99
	dest_port=80
	protocol=tcp

disposition	ALLOWED

rule_details	reference = "network:my-vpc/firewall:rule-c"
	priority = 10
	action = ALLOW
	source_range = 0.0.0.0/0
	ip_port_info = tcp:80
	direction = ingress

instance	project_id="example-proj"
	instance_name=VM1
	region=us-west1
	zone=us-west1-a

vpc	project_id="example-proj"
	vpc_name=example-net
	subnetwork_name=west-subnet

remote_location	continent country region city

Subject to the specifications (#specifications), a log entry is created in Stackdriver Logging for each firewall rule that has logging enabled if that rule applies to traffic to or from a VM instance.

Firewall rules follow the format indicated by the following table.

Stackdriver LogEntry (/logging/docs/reference/v2/rest/v2/LogEntry) JSON payload fields contain messages of the following format.

Field	Description

Field	Description
connection	IpConnection (#ipconnection) 5-Tuple describing the source and destination IP address, source and destination port, and IP protocol of this connection.
disposition	string Indicates whether the connection was ALLOWED or DENIED.
rule_details	RuleDetails (#ruledetails) Details of the rule that was applied to this connection.
instance	InstanceDetails (#instancedetails) VM instance details. In a Shared VPC configuration, `project_id` corresponds to that of the service project.
vpc	VpcDetails (#vpcdetails) VPC network details. In a Shared VPC configuration, `project_id` corresponds to that of the host project.
remote_instance	InstanceDetails If the remote endpoint of the connection was a VM in Compute Engine, this field is populated with VM instance details.
remote_vpc	VpcDetails If the remote endpoint of the connection was a VM on the VPC network, this field is populated with VPC network details.
remote_location	GeographicDetails (#geographicdetails) If the remote endpoint of the connection was external to the VPC network, this field is populated with available location metadata.

Field	Type	Description
src_ip	string	Source IP address. If the source is a Compute Engine VM, `src_ip` is the interface's internal IP address. The external, public IP address is not shown. Logging shows the IP address of the VM as the VM sees it on the packet header, the same as if you ran TCP dump on the VM.
src_port	integer	Source port
dest_ip	string	Destination IP address. If the destination is a Google Cloud VM, `dest_ip` is the interface's internal, private IP address. The external, public IP address is not shown even if it was used in making the connection.

Field	Type	Description
dest_port	integer	Destination port
protocol	integer	IP protocol of the connection

Field	Type	Description
reference	string	Reference to the firewall rule; format: `"network:{network name}/firewall:{firewall_name}"`
priority	integer	The priority for the firewall rule.
action	string	ALLOW or DENY
source_range[ ]	string	List of source ranges that the firewall rule applies to.
destination_range[ ]	string	List of destination ranges that the firewall rule applies to.
ip_port_info[ ]	IpPortDetails (#IpPortDetails)	List of ip protocols and applicable port ranges for rules.
direction	string	The direction that the firewall rule applies to (ingress or egress).
source_tag[ ]	string	List of all the source tags that the firewall rule applies to.
target_tag[ ]	string	List of all the target tags that the firewall rule applies to.
source_service_account[ ]	string	List of all the source service accounts that the firewall rule applies to.
target_service_account[ ]	string	List of all the target service accounts that the firewall rule applies to.

Field	Type	Description
ip_protocol	string	IP protocol that the firewall rule applies to. "ALL" if applies to all protocols.
port_range[ ]	string	List of applicable port ranges for rules; for example, `8080-9090`.

Field	Type	Description
project_id	string	ID of the project containing the VM
vm_name	string	Instance name of the VM
region	string	Region of the VM
zone	string	Zone of the VM

Field	Type	Description
project_id	string	ID of the project containing the network
vpc_name	string	Network on which the VM is operating
subnetwork_name	string	Subnet on which the VM is operating

Field	Type	Description
continent	string	Continent for external endpoints
country	string	Country for external endpoints
region	string	Region for external endpoints
city	string	City for external endpoints

- To set up logging and view logs, see Using Firewall Rules Logging (/vpc/docs/using-firewall-rules-logging).
- To store, search, analyze, monitor, and alert on log data and events, see Stackdriver Logging (/logging/docs).

- To export log entries, see Exporting with the Logs Viewer (/logging/docs/export/configure_export_v2).