gcloud run deploy

NAME

gcloud run deploy - create or update a Cloud Run service

SYNOPSIS

`gcloud run deploy` [[*SERVICE*](#) : SERVICE)] [--namespace(#) : namespace)=*NAMESPACE*] [--async(#) : async)]
[--concurrency(#) : concurrency)=*CONCURRENCY*] [--container(#) : container)=*CONTAINER*]
[--ingress(#) : ingress)=*INGRESS*; default="all"] [--max-instances(#) : max-instances)=*MAX_INSTANCES*]
[--min-instances(#) : min-instances)=*MIN_INSTANCES*] [--platform(#) : platform)=*PLATFORM*; default="managed"]
[--service-account(#) : service-account)=*SERVICE_ACCOUNT*] [--tag(#) : tag)=*TAG*] [--timeout(#) : timeout)=*TIMEOUT*]
[--no-traffic(#) : traffic)] [--add-volume-mount(#) : add-volume-mount)=[*volume*=*NAME*,*mount-path*=*MOUNT_PATH*,...]
--args(#) : args)=[*ARG*,...] --clear-volume-mounts(#) : clear-volume-mounts) --command(#) : command)=[*COMMAND*,...]
--cpu(#) : cpu)=*CPU* --depends-on(#) : depends-on)=[*CONTAINER*,...] --memory(#) : memory)=*MEMORY* --port(#) : port)=*PORT*
--remove-volume-mount(#) : remove-volume-mount)=[*MOUNT_PATH*,...] --[no-]use-http2(#) : [no-]use-http2)
--clear-env-vars(#) : clear-env-vars) | --env-vars-file(#) : env-vars-file)=*FILE_PATH* |
--set-env-vars(#) : set-env-vars)=[*KEY*=*VALUE*,...] | --remove-env-vars(#) : remove-env-vars)=[*KEY*,...]
--update-env-vars(#) : update-env-vars)=[*KEY*=*VALUE*,...] --clear-secrets(#) : clear-secrets) |
--set-secrets(#) : set-secrets)=[*KEY*=*VALUE*,...] | --remove-secrets(#) : remove-secrets)=[*KEY*,...]
--update-secrets(#) : update-secrets)=[*KEY*=*VALUE*,...] --image(#) : image)=*IMAGE* | --source(#) : source)=*SOURCE*]
[--clear-labels(#) : clear-labels) | --remove-labels(#) : remove-labels)=[*KEY*,...] --labels(#) : labels)=[*KEY*=*VALUE*,...] |
--update-labels(#) : update-labels)=[*KEY*=*VALUE*,...]] [--connectivity(#) : connectivity)=*CONNECTIVITY*
--clear-config-maps(#) : clear-config-maps) | --set-config-maps(#) : set-config-maps)=[*KEY*=*VALUE*,...] |
--remove-config-maps(#) : remove-config-maps)=[*KEY*,...]
--update-config-maps(#) : update-config-maps)=[*KEY*=*VALUE*,...]]
[--[no-]allow-unauthenticated(#) : [no-]allow-unauthenticated) --breakglass(#) : breakglass)=*JUSTIFICATION*
--clear-vpc-connector(#) : clear-vpc-connector) --[no-]cpu-boost(#) : [no-]cpu-boost)
--[no-]cpu-throttling(#) : [no-]cpu-throttling) --description(#) : description)=*DESCRIPTION*
--execution-environment(#) : execution-environment)=*EXECUTION_ENVIRONMENT*
--remove-containers(#) : remove-containers)=[*CONTAINER*,...]
--revision-suffix(#) : revision-suffix)=*REVISION_SUFFIX* --[no-]session-affinity(#) : [no-]session-affinity)
--vpc-connector(#) : vpc-connector)=*VPC_CONNECTOR* --vpc-egress(#) : vpc-egress)=*VPC_EGRESS*
--add-cloudsql-instances(#) : add-cloudsql-instances)=[*CLOUDSQL-INSTANCES*,...] |
--clear-cloudsql-instances(#) : clear-cloudsql-instances) |
--remove-cloudsql-instances(#) : remove-cloudsql-instances)=[*CLOUDSQL-INSTANCES*,...] |
--set-cloudsql-instances(#) : set-cloudsql-instances)=[*CLOUDSQL-INSTANCES*,...]
--add-custom-audiences(#) : add-custom-audiences)=[*CUSTOM-AUDIENCES*,...] |
--clear-custom-audiences(#) : clear-custom-audiences) |
--remove-custom-audiences(#) : remove-custom-audiences)=[*CUSTOM-AUDIENCES*,...] |
--set-custom-audiences(#) : set-custom-audiences)=[*CUSTOM-AUDIENCES*,...]
--add-volume(#) : add-volume)=[*KEY*=*VALUE*,...] --clear-volumes(#) : clear-volumes)
--remove-volume(#) : remove-volume)=[*VOLUME*,...] --binary-authorization(#) : binary-authorization)=*POLICY* |
--clear-binary-authorization(#) : clear-binary-authorization)
--clear-encryption-key-shutdown-hours(#) : clear-encryption-key-shutdown-hours) |
--encryption-key-shutdown-hours(#) : encryption-key-shutdown-hours)=*ENCRYPTION_KEY_SHUTDOWN_HOURS*
--clear-key(#) : clear-key) | --key(#) : key)=*KEY* --clear-network(#) : clear-network) | --network(#) : network)=*NETWORK*
--subnet(#) : subnet)=*SUBNET* --clear-network-tags(#) : clear-network-tags) | --network-tags(#) : network-tags)=[*TAG*,...]
--clear-post-key-revocation-action-type(#) : clear-post-key-revocation-action-type) |
--post-key-revocation-action-type(#) : post-key-revocation-action-type)=*POST_KEY_REVOCATION_ACTION_TYPE*]
[--region(#) : region)=*REGION* | --cluster(#) : cluster)=*CLUSTER*

--cluster-location(=cluster-location)=*CLUSTER_LOCATION* | --context(=context)=*CONTEXT*
--kubeconfig(=kubeconfig)=*KUBECONFIG* [*GCLOUD_WIDE_FLAG* GCLOUD-WIDE-FLAGS) …]

DESCRIPTION

Creates or updates a Cloud Run service.

EXAMPLES

To deploy a container to the service `my-backend` on Cloud Run:

```
$ gcloud run deploy my-backend --image=us-docker.pkg.dev/project/image
```

You may also omit the service name. Then a prompt will be displayed with a suggested default value:

```
$ gcloud run deploy --image=us-docker.pkg.dev/project/image
```

To deploy to Cloud Run on Kubernetes Engine, you need to specify a cluster:

```
$ gcloud run deploy --image=us-docker.pkg.dev/project/image --cluster=my-cluster
```

POSITIONAL ARGUMENTS

Service resource - Service to deploy to. The arguments in this group can be used to specify the attributes of this resource.

[*SERVICE*]

ID of the service or fully qualified identifier for the service.
To set the `service` attribute:

- provide the argument `SERVICE` on the command line;
- specify the service name from an interactive prompt.

`--namespace=`*NAMESPACE*

Specific to Cloud Run for Anthos: Kubernetes namespace for the service.
To set the `namespace` attribute:

- provide the argument `SERVICE` on the command line with a fully specified name;
- specify the service name from an interactive prompt with a fully specified name;
- provide the argument `--namespace` on the command line;
- set the property `run/namespace`;
- For Cloud Run on Kubernetes Engine, defaults to "default". Otherwise, defaults to project ID.;
- provide the argument `project` on the command line;
- set the property `core/project`.

FLAGS

`--async`

Return immediately, without waiting for the operation in progress to complete.

`--concurrency=`*`CONCURRENCY`*

Set the maximum number of concurrent requests allowed per container instance. Leave concurrency unspecified or provide the special value 'default' to receive the server default value.

`--container=`*`CONTAINER`*

Specifies a container by name. Flags following --container will apply to the specified container.

`--ingress=`*`INGRESS`*; default="all"

Set the ingress traffic sources allowed to call the service. For Cloud Run (fully managed) the `--[no-]allow-unauthenticated` flag separately controls the identities allowed to call the service. *`INGRESS`* must be one of:

`all`

Inbound requests from all sources are allowed.

`internal`

For Cloud Run (fully managed), only inbound requests from VPC networks in the same project or VPC Service Controls perimeter, as well as Pub/Sub subscriptions and Eventarc events in the same project or VPC Service Controls perimeter are allowed. All other requests are rejected. See https://cloud.google.com/run/docs/securing/ingress (https://cloud.google.com/run/docs/securing/ingress) for full details on the definition of internal traffic for Cloud Run (fully managed). For Cloud Run for Anthos, only inbound requests from the same cluster are allowed.

`internal-and-cloud-load-balancing`

Only supported for Cloud Run (fully managed). Only inbound requests from Google Cloud Load Balancing or a traffic source allowed by the internal option are allowed.

`--max-instances=`*`MAX_INSTANCES`*

The maximum number of container instances of the Service to run. Use 'default' to unset the limit and use the platform default.

`--min-instances=`*`MIN_INSTANCES`*

The minimum number of container instances for this Revision of the Service to run or 'default' to remove any minimum.

`--platform=`*`PLATFORM`*; default="managed"

Target platform for running commands. Alternatively, set the property [run/platform]. *`PLATFORM`* must be one of:

`managed`

Fully managed version of Cloud Run. Use with the `--region` flag or set the [run/region] property to specify a Cloud Run region.

`gke`

Cloud Run for Anthos on Google Cloud. Use with the `--cluster` and `--cluster-location` flags or set the [run/cluster] and [run/cluster_location] properties to specify a cluster in a given zone.

`kubernetes`

Use a Knative-compatible kubernetes cluster. Use with the `--kubeconfig` and `--context` flags to specify a kubeconfig file and the context for connecting.

`--service-account=`*`SERVICE_ACCOUNT`*

Service account associated with the revision of the service. The service account represents the identity of the running revision, and determines what permissions the revision has. For the managed platform, this is the email address of an IAM service account. For the Kubernetes-based platforms (gke, kubernetes), this is the name of a Kubernetes service account in the same namespace as the service. If not provided, the revision will use the default service account of the project, or default Kubernetes namespace service account respectively.

`--tag=`*`TAG`*

Traffic tag to assign to the newly created revision.

`--timeout=`*`TIMEOUT`*

Set the maximum request execution time (timeout). It is specified as a duration; for example, "10m5s" is ten minutes, and five seconds. If you don't specify a unit, seconds is assumed. For example, "10" is 10 seconds.

`--no-traffic`

True to avoid sending traffic to the revision being deployed. Setting this flag assigns any traffic assigned to the LATEST revision to the specific revision bound to LATEST before the deployment. The effect is that the revision being deployed will not receive traffic.
After a deployment with this flag the LATEST revision will not receive traffic on future deployments. To restore sending traffic to the LATEST revision by default, run the `gcloud run services update-traffic` (/sdk/gcloud/reference/run/services/update-traffic) command with `--to-latest`.

Container Flags

```
The following flags apply to a single container. If the --container flag is specified these
flags may only be specified after a --container flag. Otherwise they will apply to the primary
ingress container.
```

`--add-volume-mount=[`*`volume=NAME,mount-path=MOUNT_PATH`*`,...]`

Adds a mount to the current container. Must contain the keys `volume=NAME` and `mount-path=/PATH` where NAME is the name of a volume on this resource and PATH is the path within the container's filesystem to mount this volume.

`--args=`[*ARG*,…]

Comma-separated arguments passed to the command run by the container image. If not specified and no '--command' is provided, the container image's default Cmd is used. Otherwise, if not specified, no arguments are passed. To reset this field to its default, pass an empty string.

`--clear-volume-mounts`

Remove all existing mounts from the current container.

`--command=`[*COMMAND*,…]

Entrypoint for the container image. If not specified, the container image's default Entrypoint is run. To reset this field to its default, pass an empty string.

`--cpu=`*CPU*

Set a CPU limit in Kubernetes cpu units.
Cloud Run (fully managed) supports values 1, 2 and 4. For Cloud Run (fully managed), 4 cpus also requires a minimum 2Gi `--memory` value. Examples 2, 2.0, 2000m

Cloud Run for Anthos and Knative-compatible Kubernetes clusters support fractional values. Examples .5, 500m, 2

`--depends-on=`[*CONTAINER*,…]

List of container dependencies to add to the current container.

`--memory=`*MEMORY*

Set a memory limit. Ex: 1024Mi, 4Gi.

`--port=`*PORT*

Container port to receive requests at. Also sets the $PORT environment variable. Must be a number between 1 and 65535, inclusive. To unset this field, pass the special value "default". If updating an existing service with a TCP startup probe pointing to the previous container port, this will also update the probe port.

`--remove-volume-mount=`[*MOUNT_PATH*,…]

Removes the volume mounted at the specified path from the current container.

`--[no-]use-http2`

Whether to use HTTP/2 for connections to the service. Use `--use-http2` to enable and `--no-use-http2` to disable.

At most one of these can be specified:

`--clear-env-vars`

Remove all environment variables.

`--env-vars-file=`*FILE_PATH*

Path to a local YAML file with definitions for all environment variables. All existing environment variables will be removed before the new environment variables are added. Example YAML content:

```
KEY_1: "value1" KEY_2: "value 2"
```

`--set-env-vars=`[*KEY*=*VALUE*,…]

List of key-value pairs to set as environment variables. All existing environment variables will be removed first.

Only --update-env-vars and --remove-env-vars can be used together. If both are specified, --remove-env-vars will be applied first.

`--remove-env-vars=`[*KEY*,…]

List of environment variables to be removed.

`--update-env-vars=`[*KEY*=*VALUE*,…]

List of key-value pairs to set as environment variables.

Specify secrets to mount or provide as environment variables. Keys starting with a forward slash '/' are mount paths. All other keys correspond to environment variables. Values should be in the form SECRET_NAME:SECRET_VERSION. For example: '--update-secrets=/secrets/api/key=mysecret:latest,ENV=othersecret:1' will mount a volume at '/secrets/api' containing a file 'key' with the latest version of secret 'mysecret'. An environment variable named ENV will also be created whose value is version 1 of secret 'othersecret'.

At most one of these can be specified:

`--clear-secrets`

Remove all secrets.

`--set-secrets=`[*KEY*=*VALUE*,…]

List of key-value pairs to set as secrets. All existing secrets will be removed first.

Only --update-secrets and --remove-secrets can be used together. If both are specified, --remove-secrets will be applied first.

`--remove-secrets=`[*KEY*,…]

List of secrets to be removed.

`--update-secrets=`[*KEY*=*VALUE*,…]

List of key-value pairs to set as secrets.

At most one of these can be specified:

`--image=`*IMAGE*

Name of the container image to deploy (e.g. `gcr.io/cloudrun/hello:latest`).

`--source=`*SOURCE*

The location of the source to build. If a Dockerfile is present in the source code directory, it will be built using that Dockerfile, otherwise it will use Google Cloud buildpacks. See https://cloud.google.com/run/docs/deploying-source-code (https://cloud.google.com/run/docs/deploying-source-code) for more details. The location can be a directory on a local disk or a gzipped archive file (.tar.gz) in Google Cloud Storage. If the source is a local directory, this command skips the files specified in the `--ignore-file`. If `--ignore-file` is not specified, use `.gcloudignore` file. If a `.gcloudignore` file is absent and a `.gitignore` file is present in the local source directory, gcloud will use a generated Git-compatible `.gcloudignore` file that respects your .gitignored files. The global `.gitignore` is not respected. For more information on `.gcloudignore`, see `gcloud topic gcloudignore` (/sdk/gcloud/reference/topic/gcloudignore).

At most one of these can be specified:

`--clear-labels`

Remove all labels. If `--update-labels` is also specified then `--clear-labels` is applied first.

For example, to remove all labels:

```
$ gcloud run deploy --clear-labels
```

To remove all existing labels and create two new labels, *foo* and *baz*:

```
$ gcloud run deploy --clear-labels --update-labels foo=bar,baz=qux
```

`--remove-labels=`[*KEY*,…]

List of label keys to remove. If a label does not exist it is silently ignored. If `--update-labels` is also specified then `--update-labels` is applied first.

At most one of these can be specified:

`--labels=`[*KEY*=*VALUE*,…]

List of label KEY=VALUE pairs to add.
An alias to --update-labels.

`--update-labels=`[*KEY*=*VALUE*,…]

List of label KEY=VALUE pairs to update. If a label exists, its value is modified. Otherwise, a new label is created.

Only applicable if connecting to Cloud Run for Anthos deployed on Google Cloud or Cloud Run for Anthos deployed on VMware. Specify `--platform=gke` or `--platform=kubernetes` to use:

`--connectivity=`*CONNECTIVITY*

(DEPRECATED) Defaults to 'external'. If 'external', the service can be invoked through the internet, in addition to through the cluster network.

The --connectivity flag is deprecated but will continue to be supported. Prefer to use the --ingress flag instead. *CONNECTIVITY* must be one of:

`external`

Visible from outside the cluster.

`internal`

Visible only within the cluster.

Specify config map to mount or provide as environment variables. Keys starting with a forward slash '/' are mount paths. All other keys correspond to environment variables. The values associated with each of these should be in the form CONFIG_MAP_NAME:KEY_IN_CONFIG_MAP; you may omit the key within the config map to specify a mount of all keys within the config map. For example: '--update-config-maps=/my/path=myconfig,ENV=otherconfig:key.json' will create a volume with config map 'myconfig' and mount that volume at '/my/path'. Because no config map key was specified, all keys in 'myconfig' will be included. An environment variable named ENV will also be created whose value is the value of 'key.json' in 'otherconfig. Not supported on the fully managed version of Cloud Run.

At most one of these can be specified:

`--clear-config-maps`

Remove all config-maps.

`--set-config-maps=`[*KEY*=*VALUE*,…]

List of key-value pairs to set as config-maps. All existing config-maps will be removed first.

Only --update-config-maps and --remove-config-maps can be used together. If both are specified, --remove-config-maps will be applied first.

`--remove-config-maps=`[*KEY*,…]

List of config-maps to be removed.

`--update-config-maps=`[*KEY*=*VALUE*,…]

List of key-value pairs to set as config-maps.

Only applicable if connecting to Cloud Run (fully managed). Specify `--platform=managed` to use:

`--[no-]allow-unauthenticated`

Whether to enable allowing unauthenticated access to the service. This may take a few moments to take effect. Use `--allow-unauthenticated` to enable and `--no-allow-unauthenticated` to disable.

`--breakglass=`*JUSTIFICATION*

Justification to bypass Binary Authorization policy constraints and allow the operation. See https://cloud.google.com/binary-authorization/docs/using-breakglass (https://cloud.google.com/binary-authorization/docs/using-breakglass) for more information. Next update or deploy command will automatically clear existing breakglass justification.

`--clear-vpc-connector`

Remove the VPC connector for this resource.

`--[no-]cpu-boost`

Whether to allocate extra CPU to containers on startup to reduce the perceived latency of a cold start request. Enabled by default when unspecified on new services. Use `--cpu-boost` to enable and `--no-cpu-boost` to disable.

`--[no-]cpu-throttling`

Whether to throttle the CPU when the container is not actively serving requests. Use `--cpu-throttling` to enable and `--no-cpu-throttling` to disable.

`--description=`*`DESCRIPTION`*

Provides an optional, human-readable description of the service.

`--execution-environment=`*`EXECUTION_ENVIRONMENT`*

Selects the execution environment where the application will run. *`EXECUTION_ENVIRONMENT`* must be one of:

`gen1`

Run the application in a first generation execution environment.

`gen2`

Run the application in a second generation execution environment.

`--remove-containers=[`*`CONTAINER`*`,...]`

List of containers to remove.

`--revision-suffix=`*`REVISION_SUFFIX`*

Specify the suffix of the revision name. Revision names always start with the service name automatically. For example, specifying [--revision-suffix=v1] for a service named 'helloworld', would lead to a revision named 'helloworld-v1'. Set empty string to clear the suffix and resume server-assigned naming.

`--[no-]session-affinity`

Whether to enable session affinity for connections to the service. Use `--session-affinity` to enable and `--no-session-affinity` to disable.

`--vpc-connector=`*`VPC_CONNECTOR`*

Set a VPC connector for this resource.

`--vpc-egress=`*`VPC_EGRESS`*

Specify which of the outbound traffic to send through Direct VPC egress or the VPC connector for this resource. This resource must have Direct VPC egress enabled or a VPC connector to set this flag. *`VPC_EGRESS`* must be one of:

`all`

(DEPRECATED) Sends all outbound traffic through Direct VPC egress or the VPC connector. Provides the same functionality as 'all-traffic'. Prefer to use 'all-traffic' instead.

`all-traffic`

Sends all outbound traffic through Direct VPC egress or the VPC connector.

`private-ranges-only`

Default option. Sends outbound traffic to private IP addresses (RFC 1918 and Private Google Access IPs) through Direct VPC egress or the VPC connector.
Traffic to other Cloud Run services might require additional configuration. See https://cloud.google.com/run/docs/securing/private-networking#send_requests_to_other_services_and_services (https://cloud.google.com/run/docs/securing/private-networking#send_requests_to_other_services_and_services) for more information.

These flags modify the Cloud SQL instances this Service connects to. You can specify a name of a Cloud SQL instance if it's in the same project and region as your Cloud Run service; otherwise specify <project>:<region>:<instance> for the instance.

At most one of these can be specified:

`--add-cloudsql-instances=`[*CLOUDSQL-INSTANCES*,…]

Append the given values to the current Cloud SQL instances.

`--clear-cloudsql-instances`

Empty the current Cloud SQL instances.

`--remove-cloudsql-instances=`[*CLOUDSQL-INSTANCES*,…]

Remove the given values from the current Cloud SQL instances.

`--set-cloudsql-instances=`[*CLOUDSQL-INSTANCES*,…]

Completely replace the current Cloud SQL instances with the given values.

These flags modify the custom audiences that can be used in the audience field of ID token for authenticated requests.

At most one of these can be specified:

`--add-custom-audiences=`[*CUSTOM-AUDIENCES*,…]

Append the given values to the current custom audiences.

`--clear-custom-audiences`

Empty the current custom audiences.

`--remove-custom-audiences=`[*CUSTOM-AUDIENCES*,…]

Remove the given values from the current custom audiences.

`--set-custom-audiences=`[*CUSTOM-AUDIENCES*,…]

Completely replace the current custom audiences with the given values.

`--add-volume=`[*KEY*=*VALUE*,…]

Adds a volume to the Cloud Run resource. To add more than one volume, specify this flag multiple times. Volumes must have a `name` and `type` key. Only certain values are supported for `type`. Depending on the provided type, other keys will be required. The following types are supported with the specified additional keys:

`cloud-storage`: A volume representing a Cloud Storage bucket. This volume type is mounted using Cloud Storage FUSE. See https://cloud.google.com/storage/docs/gcs-fuse (https://cloud.google.com/storage/docs/gcs-fuse) for the details and limitations of this filesystem. Additional keys:

- bucket: (required) the name of the bucket to use as the source of this volume
- readonly: (optional) A boolean. If true, this volume will be read-only from all mounts.

`nfs`: Represents a volume backed by an NFS server. Additional keys:

- location: (required) The location of the NFS Server, in the form SERVER:/PATH
- readonly: (optional) A boolean. If true, this volume will be read-only from all mounts.

`--clear-volumes`

Remove all existing volumes from the Cloud Run resource, including volumes mounted as secrets

`--remove-volume=`[*VOLUME*,…]

Removes volumes from the Cloud Run resource.

At most one of these can be specified:

`--binary-authorization=`*POLICY*

Binary Authorization policy to check against. This must be set to "default".

`--clear-binary-authorization`

Remove any previously set Binary Authorization policy.

At most one of these can be specified:

`--clear-encryption-key-shutdown-hours`

Remove any previously set CMEK key shutdown hours setting.

`--encryption-key-shutdown-hours=`*ENCRYPTION_KEY_SHUTDOWN_HOURS*

The number of hours to wait before an automatic shutdown server after CMEK key revocation is detected.

At most one of these can be specified:

`--clear-key`

Remove any previously set CMEK key reference.

`--key=`*KEY*

CMEK key reference to encrypt the container with.

At most one of these can be specified:

`--clear-network`

Disconnect this Cloud Run service from the VPC network it is connected to.

Direct VPC egress setting flags group.

`--network=`*NETWORK*

The VPC network that the Cloud Run service will be able to send traffic to. If --subnet is also specified, subnet must be a subnetwork of the network specified by this --network flag. To clear existing VPC network settings, use --clear-network.

`--subnet=`*SUBNET*

The VPC subnetwork that the Cloud Run service will get IPs from. The subnetwork must be `/26` or larger. If --network is also specified, subnet must be a subnetwork of the network specified by the --network flag. If --network is not specified, network will be looked up from this subnetwork. To clear existing VPC network settings, use --clear-network.

At most one of these can be specified:

`--clear-network-tags`

Clears all existing Compute Engine tags from the Cloud Run service.

`--network-tags=`[*TAG*,…]

Applies the given Compute Engine tags (comma separated) to the Cloud Run service. To clear existing tags, use --clear-network-tags.

At most one of these can be specified:

`--clear-post-key-revocation-action-type`

Remove any previously set post CMEK key revocation action type.

`--post-key-revocation-action-type=`*POST_KEY_REVOCATION_ACTION_TYPE*

Action type after CMEK key revocation. *POST_KEY_REVOCATION_ACTION_TYPE* must be one of:

`prevent-new`

No new instances will be started after CMEK key revocation.

`shut-down`

No new instances will be started and the existing instances will be shut down after CMEK key revocation.

Arguments to locate resources, depending on the platform used.

At most one of these can be specified:

Only applicable if connecting to Cloud Run (fully managed). Specify `--platform=managed` to use:

`--region=`*`REGION`*

Region in which the resource can be found. Alternatively, set the property [run/region].

Only applicable if connecting to Cloud Run for Anthos deployed on Google Cloud. Specify `--platform=gke` to use:

Cluster resource - Kubernetes Engine cluster to connect to. The arguments in this group can be used to specify the attributes of this resource. (NOTE) Some attributes are not given arguments in this group but can be set in other ways.

To set the `project` attribute:

- provide the argument `--cluster` on the command line with a fully specified name;
- set the property `run/cluster` with a fully specified name;
- specify the cluster from a list of available clusters with a fully specified name;
- provide the argument `--project` on the command line;
- set the property `core/project`.

`--cluster=`*`CLUSTER`*

ID of the cluster or fully qualified identifier for the cluster.
To set the `cluster` attribute:

- provide the argument `--cluster` on the command line;
- set the property `run/cluster`;
- specify the cluster from a list of available clusters.

`--cluster-location=`*`CLUSTER_LOCATION`*

Zone in which the cluster is located. Alternatively, set the property [run/cluster_location].
To set the `location` attribute:

- provide the argument `--cluster` on the command line with a fully specified name;
- set the property `run/cluster` with a fully specified name;
- specify the cluster from a list of available clusters with a fully specified name;
- provide the argument `--cluster-location` on the command line;
- set the property `run/cluster_location`;
- specify the cluster location from a list of available zones.

Only applicable if connecting to Cloud Run for Anthos deployed on VMware. Specify `--platform=kubernetes` to use:

`--context=`*`CONTEXT`*

The name of the context in your kubectl config file to use for connecting.

`--kubeconfig=`*`KUBECONFIG`*

The absolute path to your kubectl config file. If not specified, the colon- or semicolon-delimited list of paths specified by $KUBECONFIG will be used. If $KUBECONFIG is unset, this defaults to `~/.kube/config`.

GCLOUD WIDE FLAGS

These flags are available to all commands: `--access-token-file` (/sdk/gcloud/reference#--access-token-file), `--account` (/sdk/gcloud/reference#--account), `--billing-project` (/sdk/gcloud/reference#--billing-project), `--configuration` (/sdk/gcloud/reference#--configuration), `--flags-file` (/sdk/gcloud/reference#--flags-file), `--flatten` (/sdk/gcloud/reference#--flatten), `--format` (/sdk/gcloud/reference#--format), `--help` (/sdk/gcloud/reference#--help), `--impersonate-service-account` (/sdk/gcloud/reference#--impersonate-service-account), `--log-http` (/sdk/gcloud/reference#--log-http), `--project` (/sdk/gcloud/reference#--project), `--quiet` (/sdk/gcloud/reference#--quiet), `--trace-token` (/sdk/gcloud/reference#--trace-token), `--user-output-enabled` (/sdk/gcloud/reference#--user-output-enabled), `--verbosity` (/sdk/gcloud/reference#--verbosity).

Run `$ gcloud help` (/sdk/gcloud/reference) for details.

NOTES

These variants are also available:

$ **`gcloud alpha run deploy`** (/sdk/gcloud/reference/alpha/run/deploy)

$ **`gcloud beta run deploy`** (/sdk/gcloud/reference/beta/run/deploy)