

Syncing to a read-only repo

This quickstart shows you how to get started with Anthos Config Management on a new cluster, using the [foo-corp example repo](#)

(<https://github.com/GoogleCloudPlatform/csp-config-management/>) to bootstrap a cluster with a set of configs. In this quickstart, you do not need write access to the repo. Imagine that a compliance team in your organization is responsible for creating the configs, and that each cluster is required to sync to the repo.

After you complete this quickstart, you can follow an advanced quickstart about [writing, testing, and syncing configs](#) (/anthos-config-management/docs/quickstart_writing_configs).

Before you begin

1. [Sign in](https://accounts.google.com/Login) (<https://accounts.google.com/Login>) to your Google Account.

If you don't already have one, [sign up for a new account](#)

(<https://accounts.google.com/SignUp>).

2. In the Cloud Console, on the project selector page, select or create a Cloud project.

[Go to the project selector page](#) (<https://console.cloud.google.com/projectselector2/home/dashboard>)

3. Make sure that billing is enabled for your Google Cloud project. [Learn how to confirm billing is enabled for your project](#) (</billing/docs/how-to/modify-project>).

4. Enable the Anthos API.

[Enable the API](#) (<https://console.cloud.google.com/flows/enableapi?apiid=anthos.googleapis.com&re>)

5. [Install and initialize the Cloud SDK](#) (</sdk/docs>).

Requirements

Anthos Config Management requires an active Anthos entitlement. For more information, see [Pricing](#) (</anthos/pricing>) for Anthos.

Cluster setup

GKE users

1. [Create a cluster](/kubernetes-engine/docs/how-to/creating-a-cluster) (/kubernetes-engine/docs/how-to/creating-a-cluster).
2. Set up the `kubectl` command to authenticate to the cluster and create a `RoleBinding` to make yourself a cluster administrator, using the following commands. Use your cluster name where you see `[MY-CLUSTER]`, and use your Google Cloud account's email address where you see `[USER-ACCOUNT]`. Depending on how you configured the `gcloud` command on your local system, you may need to add the `--project` and `--zone` fields.

```
gcloud container clusters get-credentials [MY-CLUSTER]
```

```
kubectl create clusterrolebinding cluster-admin-binding \  
  --clusterrole cluster-admin --user [USER_ACCOUNT]
```

GKE on-prem users

- [Create a user cluster](/gke-on-prem/docs/how-to/installation/getting-started) (/gke-on-prem/docs/how-to/installation/getting-started).

Anthos Config Management installation

1. [Install the `nomos` command](/anthos-config-management/docs/how-to/nomos-command#installing) (/anthos-config-management/docs/how-to/nomos-command#installing) onto your local system.
2. If you are installing manually, [install the Config Management Operator](/anthos-config-management/docs/how-to/installing#installing) (/anthos-config-management/docs/how-to/installing#installing) onto the cluster you just created.
3. [Register your cluster](/anthos/multicluster-management/connect/registering-a-cluster) (/anthos/multicluster-management/connect/registering-a-cluster) to an Anthos environ using Connect.

Configure your cluster

You can configure your cluster using `kubectl` or the Google Cloud Console.

[`kubectlConsole`](#) (#console)

Create a file `config-management.yaml` and copy the below YAML file into it. Because the repo is world-readable, `secretType` is set to `none`. For an explanation of the fields, see [Configuration for the Git repository](#) (`/anthos-config-management/docs/how-to/installing#configuring-git-repo`).

```
apiVersion: configmanagement.gke.io/v1
kind: ConfigManagement
metadata:
  name: config-management
spec:
  # clusterName is required and must be unique among all managed clusters
  clusterName: my-cluster
  git:
    syncRepo: https://github.com/GoogleCloudPlatform/csp-config-management/
    syncBranch: 1.0.0
    secretType: none
    policyDir: "foo-corp"
```

Apply the configuration to your cluster:

```
kubectl apply -f config-management.yaml
```

If the command succeeds, Kubernetes updates the Config Management Operator on your cluster to begin syncing your cluster's configuration from the repository. To verify that the Config Management Operator is running, list all Pods running in the `config-management-system` namespace:

```
kubectl get pods -n config-management-system
```

Output:

NAME	READY	STATUS	RESTARTS	AGE
git-importer-5f8bdb59bd-7nn5m	2/2	Running	0	2m
monitor-58c48fbc66-ggrmd	1/1	Running	0	2m
syncer-7bbfd7686b-dxb45	1/1	Running	0	2m

Examine your cluster and repo

The `foo-corp` repo includes configs in the `cluster/` and `namespaces/` directories. These configs are applied as soon as the Config Management Operator is configured to read from the repo.

All objects managed by Anthos Config Management have the `app.kubernetes.io/managed-by` label set to `configmanagement.gke.io`.

List namespaces managed by Anthos Config Management:

```
tl get ns -l app.kubernetes.io/managed-by=configmanagement.gke.io
```

Output:

	STATUS	AGE
	Active	4m
ing-dev	Active	4m
ing-prod	Active	4m
ing-staging	Active	4m

Examine the configs that caused these namespaces to be created, such as `namespaces/audit/namespace.yaml` and `namespaces/online/shipping-app-backend/shipping-dev/namespace.yaml`.

List ClusterRoles managed by Anthos Config Management:

```
tl get clusterroles -l app.kubernetes.io/managed-by=configmanagement.gke.io
```

Output:

	AGE
pace-reader	6m52s
reator	6m52s

Examine the ClusterRole configs declaring:

- `cluster/namespace-reader-clusterrole.yaml`

- `cluster/pod-creator-clusterrole.yaml`

You can examine other objects, such as Roles and PodSecurityPolicies, in the same way.

Attempt to manually modify a managed object

If you manually modify a Kubernetes object that is managed by Anthos Config Management, that object's configuration is automatically updated to match the object's config in your repo. To test this, delete the `shipping-dev` namespace.

```
tl delete namespace shipping-dev
```

If the `shipping-dev` namespace contains any objects that are not managed by Anthos Config Management, they are deleted as well. For the purposes of this example, it is assumed that all objects are managed by Anthos Config Management, but deleting a namespace is a destructive operation on a live cluster.

If you check immediately, the namespace may be missing, but within a few seconds, it exists again. For example:

```
tl get ns shipping-dev
```

Output:

```
from server (NotFound): namespaces "shipping-dev" not found
```

Seconds later:

```
tl get ns shipping-dev
```

Output:

	STATUS	AGE
ing-dev	Active	3s

Cleaning up

After you finish the exercises in this topic, you can clean up by deleting the cluster you used for testing.

If you want to follow the [advanced quickstart](/anthos-config-management/docs/quickstart_writing_configs) (/anthos-config-management/docs/quickstart_writing_configs) set up yet. This quickstart is a prerequisite for that topic, which also includes instructions for cleaning up.

What's next

- Work through the advanced quickstart about [writing, testing, and syncing configs](/anthos-config-management/docs/quickstart_writing_configs) (/anthos-config-management/docs/quickstart_writing_configs).
- Learn more about [writing configs](/anthos-config-management/docs/how-to/configs) (/anthos-config-management/docs/how-to/configs).
- Learn about [validating configs](/anthos-config-management/docs/how-to/validating-configs) (/anthos-config-management/docs/how-to/validating-configs).

Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/) (https://creativecommons.org/licenses/by/4.0/), and code samples are licensed under the [Apache 2.0 License](https://www.apache.org/licenses/LICENSE-2.0) (https://www.apache.org/licenses/LICENSE-2.0). For details, see the [Google Developers Site Policies](https://developers.google.com/site-policies) (https://developers.google.com/site-policies). Java is a registered trademark of Oracle and/or its affiliates.

Last updated 2020-08-12 UTC.