This page shows you how to set Cloud Storage-specific organization policies (/resource-manager/docs/organization-policy/overview) at or above the project level, which can be useful for managing bucket settings across your organization. Cloud Storage has two such organization policies currently available: one for enforcing the use of Bucket Lock retention policies (/storage/docs/bucket-lock) and one for enforcing the use of uniform bucket-level access (/storage/docs/uniform-bucket-level-access).

To require that buckets across your organization be created with proper retention policies (/storage/docs/bucket-lock):

To require that buckets across your organization be created with uniform bucket-level access (/storage/docs/uniform-bucket-level-access) enabled and prevent existing buckets from disabling uniform bucket-level access:

To prevent HMAC keys (/storage/docs/authentication/hmackeys) from being created for service accounts in your organization:

To remove an existing organization policy constraint:

- You can apply a constraint to any resource at the project-level or higher (/resource-manager/docs/cloud-platform-resource-hierarchy#resource-hierarchy-detail), including for an Organization resource.
- The `retentionPolicySeconds` and `uniformBucketLevelAccess` constraints are enforced when creating new buckets in the resource, as well as when adding/updating the relevant parameter on existing buckets in the resource.
- The `retentionPolicySeconds` and `uniformBucketLevelAccess` constraints are not enforced retroactively on existing buckets, except when the relevant parameter is being set on such a bucket.
- If a resource has existing HMAC keys when you enable the `disableServiceAccountHmacKeyCreation` constraint, those keys continue to exist.
- For retention policy constraints, if you set multiple constraints at different resource levels, they are enforced hierarchically (/resource-manager/docs/organization-policy/understanding-hierarchy#hierarchy_evaluation_rules).

For this reason, it's recommended that you set the `inheritFromParent` field to `true`, ensuring that policies at higher layers are also considered.