This page discusses the Bucket Lock feature, which allows you to configure a data retention policy for a Cloud Storage bucket that governs how long objects in the bucket must be retained. The feature also allows you to lock the data retention policy, permanently preventing the policy from being reduced or removed. For examples of using this feature, see Using Retention Policies and Bucket Lock (/storage/docs/using-bucket-lock).

This feature can provide immutable storage on Cloud Storage. Bucket Lock can help with regulatory and compliance requirements (#compliance), such as those associated with FINRA, SEC, and CFTC. Bucket Lock may also help you address certain health care industry retention regulations.

- You can add a *retention policy* to a bucket to specify a *retention period*.
  - If a bucket does not have a retention policy, you can delete or overwrite objects in the bucket at any time.
  - If a bucket has a retention policy, objects in the bucket can only be deleted or overwritten once their age is greater than the retention period.
  - A retention policy retroactively applies to existing objects in the bucket as well as new objects added to the bucket.
- You can *lock* a retention policy to **permanently** set it on the bucket.
  - Once you lock a retention policy, you cannot remove it or reduce the retention period it has.
  - You cannot delete a bucket with a locked retention policy unless every object in the bucket has met the retention period.
  - You can increase the retention period of a locked retention policy.
  - Locking a retention policy can help your data comply with record retention regulations (#compliance).

Caution: Locking a retention policy is an irreversible action. Once locked, you must delete the entire bucket in order to "remove" the bucket's retention policy. However, before you can delete the bucket, you must be able to delete all the objects in the bucket, which itself is only possible if the all objects have reached the retention period set by the retention policy.

- You can place *holds* on individual objects which prevent them from being deleted or overwritten.

You can include a retention policy when creating a new bucket, or you can add a retention policy to an existing bucket (/storage/docs/using-bucket-lock#set-policy). Placing a retention policy on a bucket ensures that all current and future objects in the bucket cannot be deleted or overwritten until they reach the age you define in the retention policy. Attempts to delete or overwrite objects whose age is less than the retention period fail with a `403 - retentionPolicyNotMet` error.

For example, say you have a bucket with two objects in it: Object A you added a month ago, and Object B you added two years ago. If you apply a retention policy to your bucket that has a retention period of 1 year, you cannot delete or overwrite Object A for another 11 months: it is currently 1 month old, but must be a least 1 year old to delete or overwrite. Object B, on the other hand, can be deleted or overwritten immediately, since its age is greater than the retention period. If you decided to overwrite Object B, this new version of Object B has an age that restarts at 0 years.

To help track when individual objects are eligible for deletion, objects in a bucket with a retention policy each have *retention expiration time* metadata. This piece of metadata shows the date and time when an object fulfills the retention period.

When working with retention policies, keep in mind the following:

- Unless the retention policy is locked (#policy-locks), you can increase, decrease, or remove the retention policy from a bucket.
- Changing a retention policy is considered a single Class A operation (/storage/pricing#operations-pricing), regardless of the number of objects affected.
- An object's editable metadata (/storage/docs/metadata#editable) is not subject to the retention policy and can be modified even when the object itself cannot be.
- A retention policy contains an *effective time*, the time after which all objects in the bucket are guaranteed to be in compliance with the retention period.
- To see the earliest date when a given object is eligible for deletion in a bucket with a retention policy, view the *retention expiration date* portion of the object's metadata (/storage/docs/viewing-editing-metadata#view).

- Retention policies and Object Versioning (/storage/docs/object-versioning) are mutually exclusive features in Cloud Storage: for a given bucket, only one of these can be enabled at a time. Any versioned objects remaining in a bucket when you apply a retention policy are also protected by the retention policy.
- You can use Object Lifecycle Management (/storage/docs/lifecycle) to automatically delete objects in a bucket, including in a bucket with a locked policy (#policy-locks). A lifecycle rule won't delete an object until after the object fulfills the retention policy.
- You should not perform parallel composite uploads (/storage/docs/gsutil/commands/cp#parallel-composite-uploads) if your bucket has a retention policy, because the component pieces cannot be deleted until each has met the bucket's minimum retention period.
- You use constraints in organization policies to require that retention policies with specific retention periods be included as part of creating a new bucket or as part of adding/updating the a retention policy on an existing bucket. For more information, including instructions for setting this organization policy, see Setting Organization Policies (/storage/docs/setting-org-policies).

Retention periods are measured in seconds; however, some tools, like the Google Cloud Console (/storage/docs/gsutil) and gsutil (/storage/docs/cloud-console) allow you to set and view retention periods with other units of time for convenience. The following conversions apply in such cases:

- A day is considered to be 86,400 seconds.
- A month is considered to be 31 days, which is 2,678,400 seconds.
- A year is considered to be 365.25 days, which is 31,557,600 seconds.

You can set a maximum retention period of 3,155,760,000 seconds (100 years).

For gsutil, when specifying a retention period, you use a `[NUMBER][UNIT]` format, where `[UNIT]` can be `s`, `m`, `d`, or `y` to signify seconds, minutes, days, or years, respectively. Only one unit of time can be used in a command. For example, you can use `900s` or `15m`, but you cannot use `15m30s`.

When you lock a retention policy (/storage/docs/using-bucket-lock#lock-bucket) on a bucket, you prevent the policy from ever being removed or the retention period from ever being reduced (although you can still increase the retention period). If you try to remove or reduce the policy duration of a locked bucket, you get a `400 BadRequestException` error. Once a retention policy is locked, you cannot delete the bucket until every object in the bucket has met the retention period.

Locking a retention policy is irreversible, and you should be familiar with the implications of doing so prior to using this feature. When you use an *unlocked* retention policy, you have the ability to remove the policy, allowing you to still delete objects when desired. When you *lock* a retention policy, you must delete the entire bucket in order to "remove" the policy. However, you can't delete the bucket if there are objects in it that haven't fulfilled their retention period. Thus, to "remove" a locked retention policy, you have to wait until every object in the bucket has fulfilled its retention period, at which point you can delete the bucket.

Additionally, when you lock a retention policy, Cloud Storage automatically applies a lien (/resource-manager/docs/project-liens) to the `projects.delete` permission for the project that contains the bucket. While in place, the lien prevents the project from being deleted. To delete the project, you must first remove all such liens (/resource-manager/docs/project-liens#removing_liens_from_a_project). Note that removing a lien requires the `resourcemanager.projects.updateLiens` permission, which is part of the `roles/owner` and `roles/resourcemanager.lienModifier` roles.

For information on how locking a retention policy can help your data comply with record retention regulations, see the compliance section (#compliance).

Object holds are metadata flags that you place on individual objects (/storage/docs/holding-objects#place-object-hold). When an object has a hold placed on it, it cannot be deleted. Cloud Storage offers the following types of holds:

- **Event-based** holds.
- **Temporary** holds.

Event-based holds can be used in conjunction with a retention policy to control retention based on the occurrence of some event, such as holding loan documents for a certain period after loan was paid. Temporary holds can be used for regulatory or legal investigation purposes, such as holding trading documents for legal investigation.

An object can have one, both, or neither hold placed on it. Both types of holds behave the same if the object is in a bucket that doesn't have a retention policy. If the object is in a bucket that has a retention policy, they have different effects on the object when the hold is released:

- An event-based hold resets the object's time in the bucket for the purposes of the retention period.
- A temporary hold does not affect the object's time in the bucket for the purposes of the retention period.

Say you have two objects - Object A and Object B - in a bucket with a 1 year retention period. When you added the objects to the bucket, you placed an event-based hold on Object A and a temporary hold on Object B. A year passes, and while you'd normally be able to delete them at this point, because both objects still have a hold on them, you can't delete either of them.

Let's say at this point you release the hold from both objects. For Object A, its time in the bucket starts from scratch for the purposes of the retention period. This means it must stay in the bucket for another year before it can be deleted or overwritten. Object B, on the other hand, can immediately be deleted or overwritten, because the temporary hold has no effect on when the object fulfilled its retention time.

In addition to placing holds on individual objects, you can enable the default event-based property (/storage/docs/holding-objects#enable-default-hold) on your bucket. When you do this, each new object that subsequently gets added to the bucket automatically has an event-based hold placed on it.

This behavior is useful when you want an object to persist in your bucket for a certain length of time after a certain event occurs. For example, your bucket may be meant to store loans that you must retain for a certain number of years once they've been paid off. With a suitable retention policy and the default event-based hold property enabled for your bucket, when you upload a loan document to your bucket, it gets an event-based hold placed on it. When the loan is paid off, you can release the hold, at which point the retention policy ensures the loan remains stored and unchangeable until it fulfills the retention period set in your retention policy.

Financial service institutions such as banks, broker-dealers and record keepers based in the U.S. are required to comply with regulations that specify requirements for electronic records retention, including retention length, record format, record quality, and record availability, amongst others. These specific regulations include:

- Securities and Exchange (SEC) Rule 17a-4(f) (https://www.ecfr.gov/cgi-bin/text-idx?SID=b6b7a79d18d000a733725e88d333ddb5&mc=true&node=pt17.4.240&rgn=div5#se17.4.240_117a_64) .
- Commodity Futures Trading Commission (CFTC) Rule 1.31(c)-(d) (https://www.ecfr.gov/cgi-bin/text-idx?SID=2404f765a6f79e0b7fcf05b6844046cb&mc=true&node=se17.1.1_131&rgn=div8) .
- Financial Industry Regulatory Authority (FINRA) Rule 4511(c) (https://www.finra.org/rules-guidance/rulebooks/finra-rules/4511).

If a Google Cloud customer determines that any of the aforementioned regulations are applicable to them, they should complete their own compliance assessment against the specific requirements with the oversight of their own legal counsel and financial regulator. To assist with your assessment process, Google Cloud engaged Cohasset Associates, Inc. (“Cohasset”) to obtain an independent and objective assessment of the compliance capabilities of Cloud Storage. Cohasset has determined that Cloud Storage, when properly configured and used with the Bucket Lock feature, may help users address certain US record retention regulations, such as: SEC Rule 17a-4(f), CFTC Rule 1.31(c)-(d) and FINRA Rule 4511(c). The Cloud Storage Bucket Lock feature includes integrated control codes that can provide you with WORM (write once read many) compliant, immutable storage on Cloud Storage. You may find the report here (/storage/certifications/cohasset-assessment-report-2018-10-09v2.pdf).

tant: To maintain your retention policies and object holds, you must keep your current Google Cloud account active a
standing. Before discontinuing your account, you must transfer your data to another compliant service to maintain
iance. Any individual tasked with removing a project lien (/resource-manager/docs/project-liens) associated with a lo
should be someone with legal or compliance accountability and outside of your IT department's accountability chai
nformation, see the Cloud Storage section of the Service Specific Terms (/terms/service-terms).