

Google API Documentation

accessapproval API Documentation

Version: v1

An API for controlling access to data by Google personnel.

Google API Documentation

Endpoint: projects.getAccessApprovalSettings

HTTP Method: GET

Path: v1/{+name}

Description: Gets the settings associated with a project, folder, or organization.

AI-Generated Documentation

Friendly Technical Description

The `GetAccessApprovalSettings` method in `accessapproval` gets the settings associated with a project, folder, or organization. This includes the approval settings (i.e. the conditions based on which access requests will be approved), the list of excluded principals, and the list of access levels for which a given approval is required.

Common Use Cases

- * Get the approval settings for a project, folder, or organization to understand the conditions under which access requests will be approved.
- * Check if a given principal is excluded from the approval process for a project, folder, or organization.
- * Determine the access levels for which a given approval is required for a project, folder, or organization.

Example Request

...

GET https://accessapproval.googleapis.com/v1/projects/my-project/accessApprovalSettings

...

Common Parameters

- * `name`: The name of the `AccessApprovalSettings` to retrieve. Format: `"{projects|folders|organizations}/{id}/accessApprovalSettings"`
- * `accessApprovalSettingsView`: The view to be applied to the returned settings. Defaults to BASIC.

Example Code

```
```python
def get_access_approval_settings(project_id: str) -> None:
 """
 Gets the settings associated with a project, folder, or organization.
 """

 from google.cloud import accessapproval_v1

 # Create a client
 client = accessapproval_v1.AccessApprovalClient()

 # The resource name of the settings to retrieve.
```

# Google API Documentation

```
Format is "{projects|folders|organizations}/{id}"
name = f"projects/{project_id}"

Retrieve the settings
settings = client.get_access_approval_settings(name=name)

Print the settings
print(f"Settings for {name}:")
print(f" Etag: {settings.etag}")
print(f" Notification emails: {settings.notification_emails}")
print(f" Enrolled services: {settings.enrolled_services}")
print(f" Enrolled ancestors: {settings.enrolled_ancestors}")
print(f" Approval request expiration: {settings.approval_request_expiration}")
print(f" Enforce single approval per request: {settings.enforce_single_approval_per_request}")
```

...

# Google API Documentation

## Endpoint: projects.updateAccessApprovalSettings

HTTP Method: PATCH

Path: v1/{+name}

Description: Updates the settings associated with a project, folder, or organization. Settings to update are determined by the value of field\_mask.

## AI-Generated Documentation

### **Technical Description**

The `UpdateAccessApprovalSettings` method updates the access approval settings of a project. These settings determine the conditions under which access approval requests are required.

### **Common Use Cases**

- \* **Enforce access approval for specific projects:** Set the `enforced` field to `true` to require access approval for all requests to access the project.
- \* **Customize approval conditions:** Configure the `notification_emails` and `approval_checklist` fields to specify who should receive approval requests and what information they should provide.
- \* **Disable access approval:** Set the `enforced` field to `false` to disable access approval for the project.

### **Example Request**

```
```json
{
  "name": "projects/my-project",
  "accessApprovalSettings": {
    "enforced": true,
    "notificationEmails": ["admin@example.com"],
    "approvalChecklist": {
      "questions": [
        {
          "question": "What is the purpose of this access request?",
          "type": "TEXT"
        }
      ]
    }
  }
}
```
```

### **Common Parameters**

- \* **name:** The name of the project to update. Format: `projects/{project}`
- \* **accessApprovalSettings:** The new access approval settings for the project.
  - \* **enforced:** Boolean. If `true`, access approval is enforced for the project.

# Google API Documentation

- \* \*\*notificationEmails:\*\* List of email addresses to which access approval requests should be sent.
- \* \*\*approvalChecklist:\*\* The approval checklist to use for access approval requests.

# Google API Documentation

## Endpoint: `projects.deleteAccessApprovalSettings`

HTTP Method: DELETE

Path: `v1/{+name}`

Description: Deletes the settings associated with a project, folder, or organization. This will have the effect of disabling Access Approval for the project, folder, or organization, but only if all ancestors also have Access Approval disabled. If Access Approval is enabled at a higher level of the hierarchy, then Access Approval will still be enabled at this level as the settings are inherited.

# Google API Documentation

## Endpoint: `projects.getServiceAccount`

HTTP Method: GET

Path: `v1/{+name}`

Description: Retrieves the service account that is used by Access Approval to access KMS keys for signing approved approval requests.

# Google API Documentation

## Endpoint: `projects.approvalRequests.list`

HTTP Method: GET

Path: `v1/{+parent}/approvalRequests`

Description: Lists approval requests associated with a project, folder, or organization. Approval requests can be filtered by state (pending, active, dismissed). The order is reverse chronological.



# Google API Documentation

**Endpoint: `projects.approvalRequests.get`**

HTTP Method: GET

Path: `v1/{+name}`

Description: Gets an approval request. Returns NOT\_FOUND if the request does not exist.

# Google API Documentation

**Endpoint: projects.approvalRequests.approve**

HTTP Method: POST

Path: v1/{+name}:approve

Description: Approves a request and returns the updated ApprovalRequest. Returns NOT\_FOUND if the request does not exist. Returns FAILED\_PRECONDITION if the request exists but is not in a pending state.

# Google API Documentation

## Endpoint: `projects.approvalRequests.dismiss`

HTTP Method: POST

Path: `v1/{+name}:dismiss`

Description: Dismisses a request. Returns the updated ApprovalRequest. NOTE: This does not deny access to the resource if another request has been made and approved. It is equivalent in effect to ignoring the request altogether. Returns NOT\_FOUND if the request does not exist. Returns FAILED\_PRECONDITION if the request exists but is not in a pending state.

# Google API Documentation

## Endpoint: `projects.approvalRequests.invalidate`

HTTP Method: POST

Path: `v1/{+name}:invalidate`

Description: Invalidates an existing ApprovalRequest. Returns the updated ApprovalRequest. NOTE: This does not deny access to the resource if another request has been made and approved. It only invalidates a single approval. Returns FAILED\_PRECONDITION if the request exists but is not in an approved state.

# Google API Documentation

## Endpoint: `folders.getAccessApprovalSettings`

HTTP Method: GET

Path: `v1/{+name}`

Description: Gets the settings associated with a project, folder, or organization.

# Google API Documentation

## Endpoint: folders.updateAccessApprovalSettings

HTTP Method: PATCH

Path: v1/{+name}

Description: Updates the settings associated with a project, folder, or organization. Settings to update are determined by the value of field\_mask.

# Google API Documentation

## Endpoint: `folders.deleteAccessApprovalSettings`

HTTP Method: DELETE

Path: `v1/{+name}`

Description: Deletes the settings associated with a project, folder, or organization. This will have the effect of disabling Access Approval for the project, folder, or organization, but only if all ancestors also have Access Approval disabled. If Access Approval is enabled at a higher level of the hierarchy, then Access Approval will still be enabled at this level as the settings are inherited.

# Google API Documentation

## Endpoint: `folders.getServiceAccount`

HTTP Method: GET

Path: `v1/{+name}`

Description: Retrieves the service account that is used by Access Approval to access KMS keys for signing approved approval requests.



# Google API Documentation

## Endpoint: `folders.approvalRequests.list`

HTTP Method: GET

Path: `v1/{+parent}/approvalRequests`

Description: Lists approval requests associated with a project, folder, or organization. Approval requests can be filtered by state (pending, active, dismissed). The order is reverse chronological.

# Google API Documentation

**Endpoint: folders.approvalRequests.get**

HTTP Method: GET

Path: v1/{+name}

Description: Gets an approval request. Returns NOT\_FOUND if the request does not exist.

# Google API Documentation

## Endpoint: `folders.approvalRequests.approve`

HTTP Method: POST

Path: `v1/{+name}:approve`

Description: Approves a request and returns the updated `ApprovalRequest`. Returns `NOT_FOUND` if the request does not exist. Returns `FAILED_PRECONDITION` if the request exists but is not in a pending state.

# Google API Documentation

## Endpoint: `folders.approvalRequests.dismiss`

HTTP Method: POST

Path: `v1/{+name}:dismiss`

Description: Dismisses a request. Returns the updated ApprovalRequest. NOTE: This does not deny access to the resource if another request has been made and approved. It is equivalent in effect to ignoring the request altogether. Returns NOT\_FOUND if the request does not exist. Returns FAILED\_PRECONDITION if the request exists but is not in a pending state.

# Google API Documentation

## Endpoint: folders.approvalRequests.invalidate

HTTP Method: POST

Path: v1/{+name}:invalidate

Description: Invalidates an existing ApprovalRequest. Returns the updated ApprovalRequest. NOTE: This does not deny access to the resource if another request has been made and approved. It only invalidates a single approval. Returns FAILED\_PRECONDITION if the request exists but is not in an approved state.

# Google API Documentation

**Endpoint: organizations.getAccessApprovalSettings**

HTTP Method: GET

Path: v1/{+name}

Description: Gets the settings associated with a project, folder, or organization.

# Google API Documentation

**Endpoint: organizations.updateAccessApprovalSettings**

HTTP Method: PATCH

Path: v1/{+name}

Description: Updates the settings associated with a project, folder, or organization. Settings to update are determined by the value of field\_mask.

# Google API Documentation

## Endpoint: organizations.deleteAccessApprovalSettings

HTTP Method: DELETE

Path: v1/{+name}

Description: Deletes the settings associated with a project, folder, or organization. This will have the effect of disabling Access Approval for the project, folder, or organization, but only if all ancestors also have Access Approval disabled. If Access Approval is enabled at a higher level of the hierarchy, then Access Approval will still be enabled at this level as the settings are inherited.



# Google API Documentation

## Endpoint: organizations.getServiceAccount

HTTP Method: GET

Path: v1/{+name}

Description: Retrieves the service account that is used by Access Approval to access KMS keys for signing approved approval requests.

# Google API Documentation

## Endpoint: organizations.approvalRequests.list

HTTP Method: GET

Path: v1/{+parent}/approvalRequests

Description: Lists approval requests associated with a project, folder, or organization. Approval requests can be filtered by state (pending, active, dismissed). The order is reverse chronological.

# Google API Documentation

**Endpoint: organizations.approvalRequests.get**

HTTP Method: GET

Path: v1/{+name}

Description: Gets an approval request. Returns NOT\_FOUND if the request does not exist.

# Google API Documentation

**Endpoint: organizations.approvalRequests.approve**

HTTP Method: POST

Path: v1/{+name}:approve

Description: Approves a request and returns the updated ApprovalRequest. Returns NOT\_FOUND if the request does not exist. Returns FAILED\_PRECONDITION if the request exists but is not in a pending state.

# Google API Documentation

## Endpoint: organizations.approvalRequests.dismiss

HTTP Method: POST

Path: v1/{+name}:dismiss

Description: Dismisses a request. Returns the updated ApprovalRequest. NOTE: This does not deny access to the resource if another request has been made and approved. It is equivalent in effect to ignoring the request altogether. Returns NOT\_FOUND if the request does not exist. Returns FAILED\_PRECONDITION if the request exists but is not in a pending state.

# Google API Documentation

## Endpoint: organizations.approvalRequests.invalidate

HTTP Method: POST

Path: v1/{+name}:invalidate

Description: Invalidates an existing ApprovalRequest. Returns the updated ApprovalRequest. NOTE: This does not deny access to the resource if another request has been made and approved. It only invalidates a single approval. Returns FAILED\_PRECONDITION if the request exists but is not in an approved state.