

Cloud Identity API | Google Cloud Documentation

Archived: 2026-04-05 19:26:09 UTC

API for provisioning and managing identity resources.

Service: cloudidentity.googleapis.com

To call this service, we recommend that you use the Google-provided [client libraries](#). If your application needs to use your own libraries to call this service, use the following information when you make the API requests.

Discovery document

A [Discovery Document](#) is a machine-readable specification for describing and consuming REST APIs. It is used to build client libraries, IDE plugins, and other tools that interact with Google APIs. One service may provide multiple discovery documents. This service provides the following discovery documents:

- [https://cloudidentity.googleapis.com/\\$discovery/rest?version=v1](https://cloudidentity.googleapis.com/$discovery/rest?version=v1)
- [https://cloudidentity.googleapis.com/\\$discovery/rest?version=v1beta1](https://cloudidentity.googleapis.com/$discovery/rest?version=v1beta1)

Service endpoint

A [service endpoint](#) is a base URL that specifies the network address of an API service. One service might have multiple service endpoints. This service has the following service endpoint and all URIs below are relative to this service endpoint:

- `https://cloudidentity.googleapis.com`

REST Resource: [v1beta1.customers.userinvitations](#)

Methods	
cancel	<code>POST /v1beta1/{name=customers/*/userinvitations/*}:cancel</code> Cancels a UserInvitation that was already sent.
get	<code>GET /v1beta1/{name=customers/*/userinvitations/*}</code> Retrieves a UserInvitation resource.
isInvitableUser	<code>GET /v1beta1/{name=customers/*/userinvitations/*}:isInvitableUser</code> Verifies whether a user account is eligible to receive a UserInvitation (is an unmanaged account).
list	<code>GET /v1beta1/{parent=customers/*}/userinvitations</code> Retrieves a list of UserInvitation resources.

Methods	
send	<p>POST /v1beta1/{name=customers*/userinvitations*}:send</p> <p>Sends a UserInvitation to email.</p>

REST Resource: [v1beta1.devices](#)

Methods	
cancelWipe	<p>POST /v1beta1/{name=devices*}:cancelWipe</p> <p>Cancels an unfinished device wipe.</p>
create	<p>POST /v1beta1/devices</p> <p>Creates a device.</p>
delete	<p>DELETE /v1beta1/{name=devices/*}</p> <p>Deletes the specified device.</p>
get	<p>GET /v1beta1/{name=devices/*}</p> <p>Retrieves the specified device.</p>
list	<p>GET /v1beta1/devices</p> <p>Lists/Searches devices.</p>
wipe	<p>POST /v1beta1/{name=devices*}:wipe</p> <p>Wipes all data on the specified device.</p>

REST Resource: [v1beta1.devices.deviceUsers](#)

Methods	
approve	<p>POST /v1beta1/{name=devices*/deviceUsers*}:approve</p> <p>Approves device to access user data.</p>
block	<p>POST /v1beta1/{name=devices*/deviceUsers*}:block</p> <p>Blocks device from accessing user data</p>
cancelWipe	<p>POST /v1beta1/{name=devices*/deviceUsers*}:cancelWipe</p> <p>Cancels an unfinished user account wipe.</p>
delete	<p>DELETE /v1beta1/{name=devices*/deviceUsers/*}</p> <p>Deletes the specified DeviceUser.</p>
get	<p>GET /v1beta1/{name=devices*/deviceUsers/*}</p> <p>Retrieves the specified DeviceUser</p>

Methods	
list	<p>GET /v1beta1/{parent=devices/*/deviceUsers}</p> <p>Lists/Searches DeviceUsers.</p>
lookup	<p>GET /v1beta1/{parent=devices/*/deviceUsers}:lookup</p> <p>Looks up resource names of the DeviceUsers associated with the caller's credentials, as well as the properties provided in the request.</p>
wipe	<p>POST /v1beta1/{name=devices/*/deviceUsers/*}:wipe</p> <p>Wipes the user's account on a device.</p>

REST Resource: [v1beta1.devices.deviceUsers.clientStates](#)

Methods	
get	<p>GET /v1beta1/{name=devices/*/deviceUsers/*/clientStates/*}</p> <p>Gets the client state for the device user</p>
patch	<p>PATCH /v1beta1/{clientState.name=devices/*/deviceUsers/*/clientStates/*}</p> <p>Updates the client state for the device user</p> <p>Note: This method is available only to customers who have one of the following SKUs: Enterprise Standard, Enterprise Plus, Enterprise for Education, and Cloud Identity Premium</p>

REST Resource: [v1beta1.groups](#)

Methods	
create	<p>POST /v1beta1/groups</p> <p>Creates a Group .</p>
delete	<p>DELETE /v1beta1/{name=groups/*}</p> <p>Deletes a Group .</p>
get	<p>GET /v1beta1/{name=groups/*}</p> <p>Retrieves a Group .</p>
getSecuritySettings	<p>GET /v1beta1/{name=groups/*/securitySettings}</p> <p>Get Security Settings</p>

Methods	
list	<p>GET /v1beta1/groups</p> <p>Lists the <code>Group</code> resources under a customer or namespace.</p>
lookup	<p>GET /v1beta1/groups:lookup</p> <p>Looks up the resource name of a <code>Group</code> by its <code>EntityKey</code>.</p>
patch	<p>PATCH /v1beta1/{resource.name=groups/*}</p> <p>Updates a <code>Group</code>.</p>
search	<p>GET /v1beta1/groups:search</p> <p>Searches for <code>Group</code> resources matching a specified query.</p>
updateSecuritySettings	<p>PATCH /v1beta1/{securitySettings.name=groups*/securitySettings}</p> <p>Update Security Settings</p>

REST Resource: [v1beta1.groups.memberships](#)

Methods	
checkTransitiveMembership	<p>GET</p> <p>/v1beta1/{parent=groups*/memberships:checkTransitiveMembership</p> <p>Check a potential member for membership in a group.</p>
create	<p>POST /v1beta1/{parent=groups*/memberships</p> <p>Creates a <code>Membership</code>.</p>
delete	<p>DELETE /v1beta1/{name=groups*/memberships/*}</p> <p>Deletes a <code>Membership</code>.</p>
get	<p>GET /v1beta1/{name=groups*/memberships/*}</p> <p>Retrieves a <code>Membership</code>.</p>
getMembershipGraph	<p>GET /v1beta1/{parent=groups*/memberships:getMembershipGraph</p> <p>Get a membership graph of just a member or both a member and a group.</p>
list	<p>GET /v1beta1/{parent=groups*/memberships</p> <p>Lists the <code>Membership</code>s within a <code>Group</code>.</p>
lookup	<p>GET /v1beta1/{parent=groups*/memberships:lookup</p> <p>Looks up the resource name of a <code>Membership</code> by its <code>EntityKey</code>.</p>
modifyMembershipRoles	<p>POST /v1beta1/{name=groups*/memberships/*}:modifyMembershipRoles</p> <p>Modifies the <code>MembershipRole</code>s of a <code>Membership</code>.</p>

Methods	
searchDirectGroups	GET /v1beta1/{parent=groups/*}/memberships:searchDirectGroups Searches direct groups of a member.
searchTransitiveGroups	GET /v1beta1/{parent=groups/*}/memberships:searchTransitiveGroups Search transitive groups of a member.
searchTransitiveMemberships	GET /v1beta1/{parent=groups/*}/memberships:searchTransitiveMemberships Search transitive memberships of a group.

REST Resource: [v1beta1.inboundOidcSsoProfiles](#)

REST Resource: [v1beta1.inboundSamlSsoProfiles](#)

REST Resource: [v1beta1.inboundSamlSsoProfiles.idpCredentials](#)

REST Resource: [v1beta1.inboundSsoAssignments](#)

REST Resource: [v1beta1.orgUnits.memberships](#)

Methods	
list	GET /v1beta1/{parent=orgUnits/*}/memberships List OrgMembership resources in an OrgUnit treated as 'parent'.
move	POST /v1beta1/{name=orgUnits/*/memberships/*}:move Move an OrgMembership to a new OrgUnit.

REST Resource: [v1beta1.policies](#)

Methods	
create	POST /v1beta1/policies Create a policy.
delete	DELETE /v1beta1/{name=policies/*} Delete a policy.
get	GET /v1beta1/{name=policies/*} Get a policy.
list	GET /v1beta1/policies List policies.

Methods	
patch	<p>PATCH /v1beta1/{policy.name=policies/*}</p> <p>Update a policy.</p>

REST Resource: [v1.customers.userinvitations](#)

Methods	
cancel	<p>POST /v1/{name=customers/*/userinvitations/*}:cancel</p> <p>Cancels a UserInvitation that was already sent.</p>
get	<p>GET /v1/{name=customers/*/userinvitations/*}</p> <p>Retrieves a UserInvitation resource.</p>
isInvitableUser	<p>GET /v1/{name=customers/*/userinvitations/*}:isInvitableUser</p> <p>Verifies whether a user account is eligible to receive a UserInvitation (is an unmanaged account).</p>
list	<p>GET /v1/{parent=customers/*/}userinvitations</p> <p>Retrieves a list of UserInvitation resources.</p>
send	<p>POST /v1/{name=customers/*/userinvitations/*}:send</p> <p>Sends a UserInvitation to email.</p>

REST Resource: [v1.devices](#)

Methods	
cancelWipe	<p>POST /v1/{name=devices/*}:cancelWipe</p> <p>Cancels an unfinished device wipe.</p>
create	<p>POST /v1/devices</p> <p>Creates a device.</p>
delete	<p>DELETE /v1/{name=devices/*}</p> <p>Deletes the specified device.</p>
get	<p>GET /v1/{name=devices/*}</p> <p>Retrieves the specified device.</p>
list	<p>GET /v1/devices</p> <p>Lists/Searches devices.</p>
wipe	<p>POST /v1/{name=devices/*}:wipe</p> <p>Wipes all data on the specified device.</p>

REST Resource: [v1.devices.deviceUsers](#)

Methods	
approve	<p>POST /v1/{name=devices/*/deviceUsers/*}:approve</p> <p>Approves device to access user data.</p>
block	<p>POST /v1/{name=devices/*/deviceUsers/*}:block</p> <p>Blocks device from accessing user data</p>
cancelWipe	<p>POST /v1/{name=devices/*/deviceUsers/*}:cancelWipe</p> <p>Cancels an unfinished user account wipe.</p>
delete	<p>DELETE /v1/{name=devices/*/deviceUsers/*}</p> <p>Deletes the specified DeviceUser.</p>
get	<p>GET /v1/{name=devices/*/deviceUsers/*}</p> <p>Retrieves the specified DeviceUser</p>
list	<p>GET /v1/{parent=devices/*/}deviceUsers</p> <p>Lists/Searches DeviceUsers.</p>
lookup	<p>GET /v1/{parent=devices/*/deviceUsers}:lookup</p> <p>Looks up resource names of the DeviceUsers associated with the caller's credentials, as well as the properties provided in the request.</p>
wipe	<p>POST /v1/{name=devices/*/deviceUsers/*}:wipe</p> <p>Wipes the user's account on a device.</p>

REST Resource: [v1.devices.deviceUsers.clientStates](#)

Methods	
get	<p>GET /v1/{name=devices/*/deviceUsers/*/clientStates/*}</p> <p>Gets the client state for the device user</p>
list	<p>GET /v1/{parent=devices/*/deviceUsers/*/}clientStates</p> <p>Lists the client states for the given search query.</p>
patch	<p>PATCH /v1/{clientState.name=devices/*/deviceUsers/*/clientStates/*}</p> <p>Updates the client state for the device user</p> <p>Note: This method is available only to customers who have one of the following SKUs: Enterprise Standard, Enterprise Plus, Enterprise for Education, and Cloud Identity Premium</p>

REST Resource: [v1.groups](#)

Methods	
create	POST /v1/groups Creates a Group.
delete	DELETE /v1/{name=groups/*} Deletes a Group .
get	GET /v1/{name=groups/*} Retrieves a Group .
getSecuritySettings	GET /v1/{name=groups*/securitySettings} Get Security Settings
list	GET /v1/groups Lists the Group resources under a customer or namespace.
lookup	GET /v1/groups:lookup Looks up the resource name of a Group by its EntityKey .
patch	PATCH /v1/{resource.name=groups/*} Updates a Group .
search	GET /v1/groups:search Searches for Group resources matching a specified query.
updateSecuritySettings	PATCH /v1/{securitySettings.name=groups*/securitySettings} Update Security Settings

REST Resource: [v1.groups.memberships](#)

Methods	
checkTransitiveMembership	GET /v1/{parent=groups*/memberships:checkTransitiveMembership Check a potential member for membership in a group.
create	POST /v1/{parent=groups*/memberships Creates a Membership .
delete	DELETE /v1/{name=groups*/memberships/*} Deletes a Membership .
get	GET /v1/{name=groups*/memberships/*} Retrieves a Membership .

Methods	
getMembershipGraph	<p>GET /v1/{parent=groups/*}/memberships:getMembershipGraph</p> <p>Get a membership graph of just a member or both a member and a group.</p>
list	<p>GET /v1/{parent=groups/*}/memberships</p> <p>Lists the Membership s within a Group .</p>
lookup	<p>GET /v1/{parent=groups/*}/memberships:lookup</p> <p>Looks up the resource name of a Membership by its EntityKey .</p>
modifyMembershipRoles	<p>POST /v1/{name=groups/*/memberships/*}:modifyMembershipRoles</p> <p>Modifies the MembershipRole s of a Membership .</p>
searchDirectGroups	<p>GET /v1/{parent=groups/*}/memberships:searchDirectGroups</p> <p>Searches direct groups of a member.</p>
searchTransitiveGroups	<p>GET /v1/{parent=groups/*}/memberships:searchTransitiveGroups</p> <p>Search transitive groups of a member.</p>
searchTransitiveMemberships	<p>GET</p> <p>/v1/{parent=groups/*}/memberships:searchTransitiveMemberships</p> <p>Search transitive memberships of a group.</p>

REST Resource: [v1.inboundOidcSsoProfiles](#)

REST Resource: [v1.inboundSamlSsoProfiles](#)

REST Resource: [v1.inboundSamlSsoProfiles.idpCredentials](#)

REST Resource: [v1.inboundSsoAssignments](#)

REST Resource: [v1.policies](#)

Methods	
get	<p>GET /v1/{name=policies/*}</p> <p>Get a policy.</p>
list	<p>GET /v1/policies</p> <p>List policies.</p>