The REST API is now versioned. For more information, see "About API versioning."

LDAP

Use the REST API to update account relationships between a GitHub Enterprise Server user or team and its linked LDAP entry or queue a new synchronization.

About LDAP 🔗

You can use these endpoints to update the Distinguished Name (DN) that a user or team maps to. Note that in most cases, you must have LDAP Sync enabled for your GitHub Enterprise Server appliance. The "Update LDAP mapping for a user" endpoint can be used when LDAP is enabled, even if LDAP Sync is disabled.

These endpoints only support authentication using a personal access token (classic). For more information, see "Managing your personal access tokens."

Update LDAP mapping for a team 🔗

Updates the distinguished name (DN) of the LDAP entry to map to a team. LDAP synchronization must be enabled to map LDAP entries to a team. Use the Create a team endpoint to create a team with LDAP mapping.

Parameters for "Update LDAP mapping for a team"

Headers

accept string

Setting to `application/vnd.github+json` is recommended.

Path parameters

team_id integer   Required

The unique identifier of the team.

Body parameters

ldap_dn string   Required

The distinguished name (DN) of the LDAP entry to map to a team.

HTTP response status codes for "Update LDAP mapping for a team"

Status code	Description

`200`   OK

Code samples for "Update LDAP mapping for a team"

PATCH	`/admin/ldap/teams/{team_id}/mapping`

cURL	JavaScript	GitHub CLI	⧉

```
curl -L \ -X PATCH \ -H "Accept: application/vnd.github+json" \ -H "Authorization: Bearer <YOUR-TOKEN>" \ -H "X-GitHub-Api-Version:
2022-11-28" \ http(s)://HOSTNAME/api/v3/admin/ldap/teams/TEAM_ID/mapping \ -d '{"ldap_dn":"cn=Enterprise
Ops,ou=teams,dc=github,dc=com"}'
```

Response

Example response	Response schema

Status: 200

```
{ "ldap_dn": "cn=Enterprise Ops,ou=teams,dc=github,dc=com", "id": 1, "node_id": "MDQ6VGVhbTE=", "url": "https://HOSTNAME/teams/1",
"html_url": "https://github.com/orgs/github/teams/justice-league", "name": "Justice League", "slug": "justice-league", "description":
"A great team.", "privacy": "closed", "notification_setting": "notifications_enabled", "permission": "admin", "members_url":
"https://HOSTNAME/teams/1/members{/member}", "repositories_url": "https://HOSTNAME/teams/1/repos", "parent": null }
```

Sync LDAP mapping for a team 🔗

Note that this API call does not automatically initiate an LDAP sync. Rather, if a `201` is returned, the sync job is queued successfully, and is performed when the instance is ready.

Parameters for "Sync LDAP mapping for a team"

Headers

`accept` string

Setting to `application/vnd.github+json` is recommended.

Path parameters

`team_id` integer   Required

The unique identifier of the team.

HTTP response status codes for "Sync LDAP mapping for a team"

Status code	Description
`201`	Created

Code samples for "Sync LDAP mapping for a team"

`POST` /admin/ldap/teams/{team_id}/sync

cURL   JavaScript   GitHub CLI

```
curl -L \ -X POST \ -H "Accept: application/vnd.github+json" \ -H "Authorization: Bearer <YOUR-TOKEN>" \ -H "X-GitHub-Api-Version:
2022-11-28" \ http(s)://HOSTNAME/api/v3/admin/ldap/teams/TEAM_ID/sync
```

Response

Example response   Response schema

Status: 201

```
{ "status": "queued" }
```

Update LDAP mapping for a user 🔗

Parameters for "Update LDAP mapping for a user"

Headers

`accept` string

Setting to `application/vnd.github+json` is recommended.

Path parameters

`username` string Required

The handle for the GitHub user account.

Body parameters

`ldap_dn` string Required

The distinguished name (DN) of the LDAP entry to map to a team.

HTTP response status codes for "Update LDAP mapping for a user"

Status code	Description
`200`	OK

Code samples for "Update LDAP mapping for a user"

`PATCH` `/admin/ldap/users/{username}/mapping`

| cURL | JavaScript | GitHub CLI | |

```
curl -L \ -X PATCH \ -H "Accept: application/vnd.github+json" \ -H "Authorization: Bearer <YOUR-TOKEN>" \ -H "X-GitHub-Api-Version:
2022-11-28" \ http(s)://HOSTNAME/api/v3/admin/ldap/users/USERNAME/mapping \ -d '{"ldap_dn":"uid=asdf,ou=users,dc=github,dc=com"}'
```

Response

| Example response | Response schema |

Status: 200

```
{ "ldap_dn": "uid=asdf,ou=users,dc=github,dc=com", "login": "octocat", "id": 1, "node_id": "MDQ6VXNlcjE=", "avatar_url":
"https://github.com/images/error/octocat_happy.gif", "gravatar_id": "", "url": "https://HOSTNAME/users/octocat", "html_url":
"https://github.com/octocat", "followers_url": "https://HOSTNAME/users/octocat/followers", "following_url":
"https://HOSTNAME/users/octocat/following{/other_user}", "gists_url": "https://HOSTNAME/users/octocat/gists{/gist_id}",
"starred_url": "https://HOSTNAME/users/octocat/starred{/owner}{/repo}", "subscriptions_url":
"https://HOSTNAME/users/octocat/subscriptions", "organizations_url": "https://HOSTNAME/users/octocat/orgs", "repos_url":
```

Sync LDAP mapping for a user 🔗

Note that this API call does not automatically initiate an LDAP sync. Rather, if a `201` is returned, the sync job is queued successfully, and is performed when the instance is ready.

Parameters for "Sync LDAP mapping for a user"

Headers

`accept` string

Setting to `application/vnd.github+json` is recommended.

Path parameters

`username` string Required

The handle for the GitHub user account.

HTTP response status codes for "Sync LDAP mapping for a user"

Status code	Description
`201`	Created

Code samples for "Sync LDAP mapping for a user"

`POST` `/admin/ldap/users/{username}/sync`

| cURL | JavaScript | GitHub CLI | |

```
curl -L \ -X POST \ -H "Accept: application/vnd.github+json" \ -H "Authorization: Bearer <YOUR-TOKEN>" \ -H "X-GitHub-Api-Version:
2022-11-28" \ http(s)://HOSTNAME/api/v3/admin/ldap/users/USERNAME/sync
```

Response

Example response	Response schema

Status: 201

```
{ "status": "queued" }
```