LDAP commands

- LDAP commands
  - ldap:search
  - ldap:check-user
  - ldap:check-group
  - ldap:create-empty-config
  - ldap:delete-config
  - ldap:set-config
  - ldap:test-config
  - ldap:test-user-settings
  - ldap:show-remnants

LDAP commands

> These commands are only available when the "LDAP user and group backend" app (`user_ldap`) is enabled.

These LDAP commands appear only when you have enabled the LDAP app. Then you can run the following LDAP commands with `occ`:

```
ldap
 ldap:check-user              checks whether a user exists on LDAP.
 ldap:check-group             checks whether a group exists on LDAP.
 ldap:create-empty-config     creates an empty LDAP configuration
 ldap:delete-config           deletes an existing LDAP configuration
 ldap:search                  executes a user or group search
```

```
ldap:set-config              modifies an LDAP configuration
ldap:show-config             shows the LDAP configuration
ldap:show-remnants           shows which users are not available on
                             LDAP anymore, but have remnants in
                             Nextcloud.
ldap:test-config             tests an LDAP configuration
ldap:test-user-settings      runs tests and show information about user
                             related LDAP settings
```

ldap:search

Search for an LDAP user, using this syntax

> sudo -E -u www-data php occ ldap:search [--group] [--offset="..."] [--limit="..."] search

Searches will match at the beginning of the attribute value only. This example searches for givenNames that start with "rob":

```
sudo -E -u www-data php occ ldap:search "rob"
```

This will find robbie, roberta, and robin. Broaden the search to find, for example, `jeroboam` with the asterisk wildcard:

```
sudo -E -u www-data php occ ldap:search "*rob"
```

User search attributes are set with `ldap:set-config` (below). For example, if your search attributes are `givenName` and `sn` you can find users by first name + last name very quickly. For example, you'll find Terri Hanson by searching for `te ha`. Trailing whitespaces are ignored.

ldap:check-user

Check if an LDAP user exists. This works only if the Nextcloud server is connected to an LDAP server:

```
sudo -E -u www-data php occ ldap:check-user robert
```

Use `--update` to update the account fields from LDAP:

```
sudo -E -u www-data php occ ldap:check-user --update robert
```

Will not run a check when it finds a disabled LDAP connection. This prevents users that exist on disabled LDAP connections from being marked as deleted. If you know for certain that the user you are searching for is not in one of the disabled connections, and exists on an active connection, use the `--force` option to force it to check all active LDAP connections:

```
sudo -E -u www-data php occ ldap:check-user --force robert
```

You can also use `--all-seen-users` to run the check on all users that logged into Nextcloud at least once. You may use `--limit` and `--offset` to do so by batch. You can combine this with `--update` to update the information of seen users.

ldap:check-group

Checks whether a group still exists in the LDAP directory. Use with `--update` to update the group membership cache on the Nextcloud side:

```
sudo -E -u www-data php occ ldap:check-group --update mygroup
```

ldap:create-empty-config

Creates an empty LDAP configuration. The first one you create has `configID` `s01`, and all subsequent configurations that you create are automatically assigned IDs:

```
sudo -E -u www-data php occ ldap:create-empty-config
   Created new configuration with configID 's01'
```

Then you can list and view your configurations:

```
sudo -E -u www-data php occ ldap:show-config
```

And view the configuration for a single configID:

```
sudo -E -u www-data php occ ldap:show-config s01
```

ldap:delete-config

Deletes an existing LDAP configuration:

```
sudo -E -u www-data php occ ldap:delete  s01
Deleted configuration with configID 's01'
```

ldap:set-config

This command is for manipulating configurations, like this example that sets search attributes:

```
sudo -E -u www-data php occ ldap:set-config s01 ldapAttributesForUserSearch
"cn;givenname;sn;displayname;mail"
```

ldap:test-config

Tests whether your configuration is correct and can bind to the server:

```
sudo -E -u www-data php occ ldap:test-config s01
The configuration is valid and the connection could be established!
```

ldap:test-user-settings

Tests user-related LDAP settings:

```
sudo -E -u www-data php occ ldap:test-user-settings "cn=philip j.
fry,ou=people,dc=planetexpress,dc=com" --group "Everyone"

User cn=philip j. fry,ou=people,dc=planetexpress,dc=com is mapped with account name fry.
Known UUID is ce6cd914-71d5-103f-95a8-ad2dab17b2f9.
Configuration prefix is s01

Attributes set in configuration:
- ldapExpertUsernameAttr: uid
- ldapUuidUserAttribute: auto
- ldapEmailAttribute: mail
- ldapUserDisplayName: cn

Attributes fetched from LDAP using filter (|(objectclass=inetOrgPerson)):
- entryuuid: ["ce6cd914-71d5-103f-95a8-ad2dab17b2f9"]
- uid: ["fry"]
```

```
- mail: ["fry@planetexpress.com"]
- cn: ["Philip J. Fry"]

Detected UUID attribute: entryuuid

UUID for cn=philip j. fry,ou=people,dc=planetexpress,dc=com: ce6cd914-71d5-103f-95a8-
ad2dab17b2f9

Group information:
Configuration:
- ldapGroupFilter: (|(objectclass=groupOfNames))
- ldapGroupMemberAssocAttr: member

Primary group:
Group from gidNumber:
All known groups: ["Ship crew", "Everyone"]
MemberOf usage: off (0,1)

Group Everyone:
Group cn=everyone,ou=groups,dc=planetexpress,dc=com is mapped with name Everyone.
Known UUID is ce8b61c2-71d5-103f-95af-ad2dab17b2f9.
Members: ["bender", "fry", "leela"]
```

ldap:show-remnants

Used to clean up the LDAP mappings table, and is documented in ../configuration_user/user_auth_ldap_cleanup.

□□□ #2
□ cocobook-sync □□□ 2026-10-28 16:27:26 UTC
□ cocobook-sync □□□ 2026-10-29 01:09:18 UTC