Integração do Nextcloud com Authentik OIDC

Link: https://integrations.goauthentik.io/chat-communication-collaboration/nextcloud/

Em 09/04/2026

What is Nextcloud

> Nextcloud is a suite of client-server software for creating and using file hosting services. Nextcloud is free and open-source, which means that anyone is allowed to install and operate it on their own private server devices.
>
> -- https://nextcloud.com/arning

WARNING

If you require server side encryption, you must use LDAP. OpenID and SAML will cause **irrevocable data loss**. Nextcloud server side encryption requires access to the user's cleartext password, which Nextcloud has access to only when using LDAP because the user enters their password directly into Nextcloud.aution

This setup only works when Nextcloud is running with HTTPS enabled. See here on how to configure this.nfo

If there’s an issue with the configuration, you can log in using the built-in authentication by

visiting http://nextcloud.company/login?direct=1.

Configuration methods

It is possible to configure Nextcloud to use OIDC, SAML, or LDAP for authentication. Below are the steps to configure each method.

- OIDC
- SAML

- LDAP

OIDC

Preparation

The following placeholders are used in this guide:

- `nextcloud.company` is the FQDN of the Nextcloud installation.
- `authentik.company` is the FQDN of the authentik installation.

Info

This documentation lists only the settings that you need to change from their default values. Be aware that any changes other than those explicitly mentioned in this guide could cause issues accessing your application.

WARNING

If you require server side encryption, you must use LDAP. OpenID and SAML will cause **irrevocable data loss**.

Let's start by considering which user attributes need to be available in Nextcloud:

- name
- email
- unique user ID
- storage quota (optional)
- groups (optional)

authentik already provides some default *scopes* with *claims*, such as:

- `email` scope: includes `email` and `email_verified`
- `profile` scope: includes `name`, `given_name`, `preferred_username`, `nickname`, `groups`
- `openid` scope: a default required by the OpenID spec (contains no claims)

Create property mapping *(optional)*

If you do not need storage quota, group information, or to manage already existing users in Nextcloud, skip to the next section.
If you want to control user storage and designate Nextcloud administrators, you will need to create a property mapping.

1. Log in to authentik as an administrator and open the authentik Admin interface.
2. Navigate to **Customization** > **Property mappings** and click **Create**.
   - **Select type**: select **Scope mapping**.
   - **Create Scope Mapping**:
     - **Name**: `Nextcloud Profile`
     - **Scope name**: `nextcloud`
     - **Expression**:

```
# Extract all groups the user is a member of
groups = [group.name for group in user.groups.all()]

# In Nextcloud, administrators must be members of a fixed group called "admin".

# If a user is an admin in authentik, ensure that "admin" is appended to their group list.
if user.is_superuser and "admin" not in groups:
    groups.append("admin")

return {
    "name": request.user.name,
    "groups": groups,
     # Set a quota by using the "nextcloud_quota" property in the user's attributes
    "quota": user.group_attributes().get("nextcloud_quota", None),
     # To connect an existing Nextcloud user, set "nextcloud_user_id" to the Nextcloud username.
    "user_id": user.attributes.get("nextcloud_user_id", str(user.uuid)),
}
```

3. Click **Finish**.

Info

To set a quota, define the `nextcloud_quota` attribute for individual users or groups. For example, setting it to `1 GB` will restrict the user to 1GB of storage. If not set, storage is unlimited.

To connect to an existing Nextcloud user, set the `nextcloud_user_id` attribute to match the Nextcloud username (found under the user's `Display name` in Nextcloud).

Create an application and provider in authentik

1. Log in to authentik as an administrator and open the authentik Admin interface.

2. Navigate to **Applications** > **Applications** and click **Create with Provider** to create an application and provider pair. (Alternatively you can first create a provider separately, then create the application and connect it with the provider.)
   - **Application**: provide a descriptive name, an optional group for the type of application, the policy engine mode, and optional UI settings.
   - **Choose a Provider type**: select **OAuth2/OpenID Connect** as the provider type.
   - **Configure the Provider**: provide a name (or accept the auto-provided name), the authorization flow to use for this provider, and the following required configurations.
     - Note the **Client ID** and **slug** values because they will be required later.
     - Set a `Strict` redirect URI to `https://nextcloud.company/apps/user_oidc/code`.
     - Select any available signing key.
     - Under **Advanced protocol settings**:
       - *(optional)* If you created the `Nextcloud Profile` scope mapping, add it to **Selected Scopes**.
       - **Subject Mode**: `Based on the User's UUID`
   - **Configure Bindings** *(optional)*: you can create a binding (policy, group, or user) to manage the listing and access to applications on a user's **My applications** page.
3. Click **Submit** to save the new application and provider.

Info

Depending on your Nextcloud configuration, you may need to use `https://nextcloud.company/index.php/` instead of `https://nextcloud.company/`.

Nextcloud configuration

1. In Nextcloud, ensure that the **OpenID Connect user backend** app is installed.
2. Log in to Nextcloud as an administrator and navigate to **Settings** > **OpenID Connect**.
3. Click the **+** button and enter the following settings:
   - **Identifier**: `authentik`
   - **Client ID**: Client ID from authentik
   - **Client secret**: Client secret from authentik
   - **Discovery endpoint**: `https://authentik.company/application/o/<application_slug>/.well-known/openid-configuration`
   - **Scope**: `email profile nextcloud openid`
   - Under **Attribute mappings**:
     - **User ID mapping**: `sub` (or `user_id` for existing users)
     - **Display name mapping**: `name`
     - **Email mapping**: `email`
     - **Quota mapping**: `quota` (leave blank if the `Nextcloud Profile` property mapping was skipped)

- **Groups mapping**: `groups` (leave blank if the `Nextcloud Profile` property mapping was skipped)
  Tip: Enable **Use group provisioning** to allow writing to this field.

- **Use unique user ID**: If this option is disabled, Nextcloud will use the mapped user ID as the Federated Cloud ID.

Info

If authentik and Nextcloud are running on the same host, you will need to add `'allow_local_remote_servers' => true` to your nextcloud `config.php` file. This setting allows remote servers with local addresses.

Info

To avoid a hashed Federated Cloud ID, deselect **Use unique user ID** and use `user_id` for the User ID mapping.

Danger

If you're using the `Nextcloud Profile` property mapping and want administrators to retain their ability to log in, make sure that **Use unique user ID** is disabled. If this setting is enabled, it will remove administrator users from the internal admin group and replace them with a hashed group ID named "admin," which does not have real administrative privileges.

Enabling OIDC back-channel logout

To automatically log users out of their Nextcloud sessions when they log out of authentik, enable back-channel logout.

1. Log in to Nextcloud as an administrator and navigate to **Settings** > **OpenID Connect**.
2. Under **Registered Providers**, locate the provider with the identifier used earlier.
3. Copy the `back-channel-logout-URL` value for that provider.
   For example: `https://nextcloud.company/apps/user_oidc/backchannel-logout/<identifier>`
4. In authentik, navigate to **Applications** > **Providers** and edit the Nextcloud provider.
5. Under **Protocol Settings**, set the **Logout URI** to the copied back-channel logout URL.
6. Set the **Logout Method** to `Back-channel`.

Making OIDC the default login method

Automatically redirect users to authentik when they access Nextcloud by running the following command on your Nextcloud docker host:

```
Opção 1 - (Tela de Login Authentik direto)
sudo docker exec --user www-data -it nextcloud-aio-nextcloud php occ config:app:set --value=0
```

user_oidc allow_multiple_user_backends.

Opção 2 - (Tela de Login Authentik e Nextcloud)
sudo docker exec --user www-data -it nextcloud-aio-nextcloud php occ config:app:set **--value=1** user_oidc allow_multiple_user_backends.

Configuration verification

To confirm that authentik is correctly configured with Nextcloud, log out and then log back in by clicking **OpenID Connect**. You'll then be redirected to authentik to log in, and once authentication is successful, you'll reach the Nextcloud dashboard.

Resources

- Nextcloud docs - User authentication with LDAP
- Nextcloud OIDC App - User Documentation

Help us improve this content

We welcome your knowledge and expertise. If you see areas of the documentation that you can improve (fix a typo, correct a technical detail, add additional context, etc.) we would really appreciate your contribution.

- Edit on GitHub
- Contributor Guide
- Open an issue
- Get Enterprise Support

Revision #3
Created 9 April 2026 19:38:20 by Administrador
Updated 14 April 2026 00:21:38 by Administrador