

Produit - GitLab

Contient les procédures et documentations liées à GITLAB

- [GITLAB - Configuration SSO Oauth avec keycloak](#)

GITLAB - Configuration SSO Oauth avec keycloak



Difficulté : Confirmé

Notions : Authentification, SSO

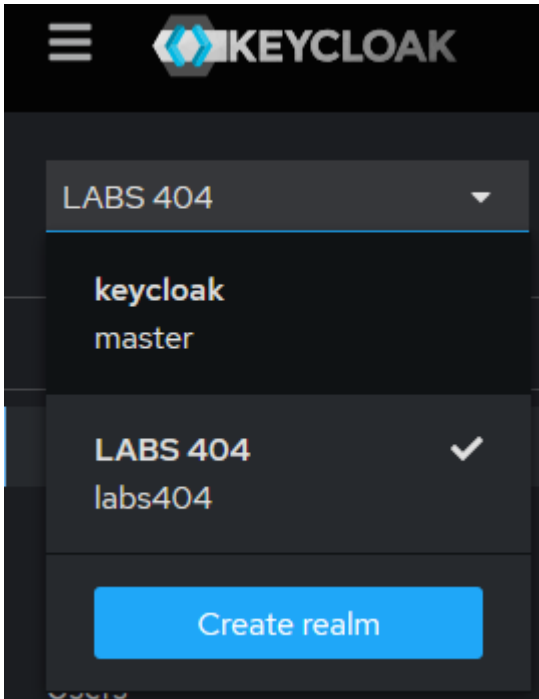
I. Introduction

Cette procédure à pour but d'expliquer le paramétrage de l'authentification SSO de Gitlab avec Keycloak.

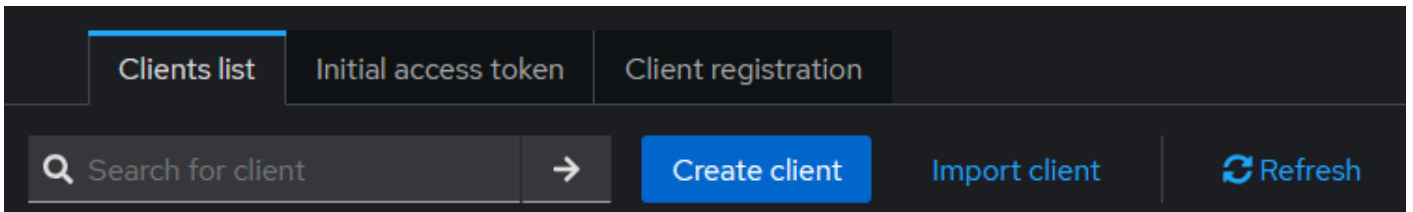
II. Côté Keycloak

2.1 Créer le client

Se connecter au keycloak et choisir le realm sur lequel ajouter le client Gtilab.



Cliquer sur '**Create client**'



Sélectionner le type '**OpenID Connect**'

Générer ou définir un '**clientID**'.

Il est possible d'utiliser un générateur afin de générer une chaîne de 32 caractères. Par exemple : [ici](#).

Best Practice : 32 caractères avec minuscules, majuscules et chiffres.

Choisir un nom parlant.

Puis faire '**Next**'.

Create client

Clients are applications and services that can request authentication of a user.

1 General settings

2 Capability config

3 Login settings

Client type ⓘ
OpenID Connect

Client ID * ⓘ
UBMudbmPLGbMvn7BEjAeUcG7xro7vftU

Name ⓘ
sc-lab-portainer

Description ⓘ

Always display in UI ⓘ ☐ Off

Back

Next

Cancel

Sur la page suivante, activer '**Client authentication**' et '**Authorization**'.

Vérifier que les cases suivantes sont cochées.

Puis faire '**Next**'.

1 General settings

2 Capability config

3 Login settings

Client authentication ?
☒ On

Authorization ?
☒ On

Authentication flow

☒ Standard flow ?

☐ Implicit flow ?

☒ OAuth 2.0 Device Authorization Grant ?

☐ OIDC CIBA Grant ?

☒ Direct access grants ?

☒ Service accounts roles ?

Back

Next

Cancel

Entrer les URL au format suivant (en modifiant les FQND).

Puis faire '**Save**'.

Access settings

Root URL ?	https://git.labs404.fr/
Home URL ?	https://git.labs404.fr/
Valid redirect URIs ?	https://git.labs404.fr/users/auth/openid_connect/callback
	+ Add valid redirect URIs
Valid post logout redirect URIs ?	https://git.labs404.fr/
	+ Add valid post logout redirect URIs
Web origins ?	https://git.labs404.fr/
	+ Add web origins
Admin URL ?	https://git.labs404.fr/

2.2 Préparer la synchro des groupes

Afin de pouvoir gérer les droits depuis le keycloak et synchroniser les groupes, aller dans l'onglet '**Client scope**'.

Choisir le client scope au nom du client suivi de '**-dedicated**'.

0a0t7W8B5fHYGfTGJZOKudHfCKo08fGg OpenID Connect
Clients are applications and services that can request authentication of a user.

Settings

Keys

Credentials

Roles

Client scopes

Service accounts roles

Sessions

Advanced

Setup

Evaluate

Name

Search by name

→

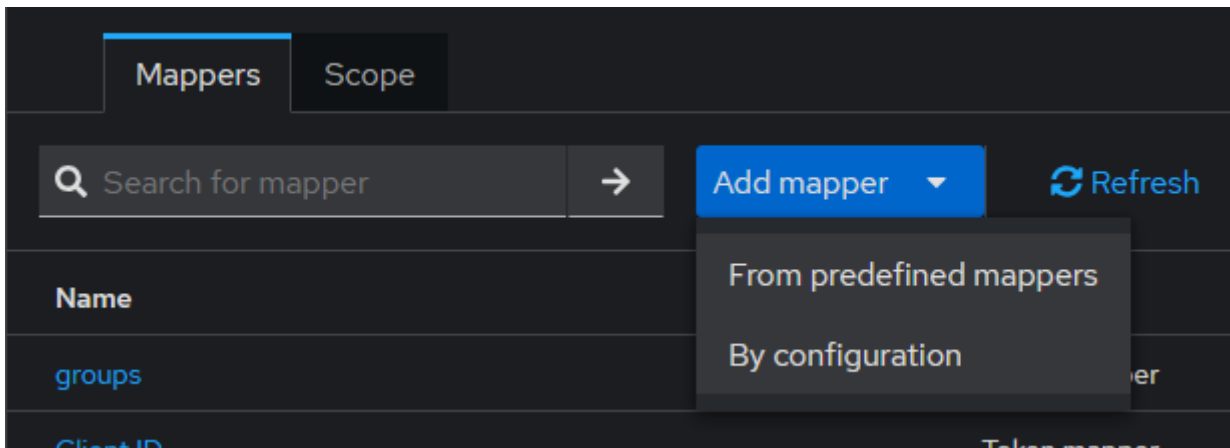
Add client scope

Change type to

Refresh

☐ Assigned client scope	Assigned type	Description
☐ 0a0t7W8B5fHYGfTGJZOKudHfCKo08fGg-dedicated	None	Dedicated scope and mappers for this client

Faire '**Add Mapper**' puis '**By Configuration**'.



Choisir '**Group Membership**'.

	into an ID token.
Group Membership	Map user group membership
Hardcoded claim	Hardcode a claim into the token

Définir le nom du **mapper** en '**groups**'.

Définir le Token **claim name** en '**groups**'.

Vérifier que les fonctions suivantes soit activées / désactivées.

Puis faire '**Save**'.

Group Membership

509d0b59-4f86-478a-ac6c-ca1762f88988

Mapper type: Group Membership

Name * ⓘ: groups

Token Claim Name ⓘ: groups

Full group path ⓘ: ☐ Off

Add to ID token ⓘ: ☒ On

Add to access token ⓘ: ☒ On

Add to lightweight access token ⓘ: ☐ Off

Add to userinfo ⓘ: ☒ On

Add to token introspection ⓘ: ☒ On

[Save](#) [Cancel](#)

2.3 Récupérer les URL

Pour récupérer les URL, sélectionner le realm et faire '**Realm settings**' puis cliquer sur '**OpenID Endpoint Configuration**'.

Configure

Realm settings

Authentication

Identity providers

User federation

User-managed access ⓘ: ☐ Off

Unmanaged Attributes ⓘ: Disabled

Endpoints ⓘ: [OpenID Endpoint Configuration](#) [SAML 2.0 Identity Provider Metadata](#)

[Save](#) [Revert](#)

Une page s'affiche et permet de récupérer les URL.

JSON		Données brutes	En-têtes
Enregistrer		Copier	Tout réduire
		Tout développer	Filtrer le JSON
issuer:	"https://auth.labs404.fr/realms/labs404"		
▼ authorization_endpoint:	"https://auth.labs404.fr/realms/labs404/protocol/openid-connect/auth"		
▼ token_endpoint:	"https://auth.labs404.fr/realms/labs404/protocol/openid-connect/token"		
▼ introspection_endpoint:	"https://auth.labs404.fr/realms/labs404/protocol/openid-connect/token/introspect"		
▼ userinfo_endpoint:	"https://auth.labs404.fr/realms/labs404/protocol/openid-connect/userinfo"		
▼ end_session_endpoint:	"https://auth.labs404.fr/realms/labs404/protocol/openid-connect/Logout"		
frontchannel_logout_session_supported:	true		
frontchannel_logout_supported:	true		
▼ jwks_uri:	"https://auth.labs404.fr/realms/labs404/protocol/openid-connect/certs"		
▼ check_session_iframe:	"https://auth.labs404.fr/realms/labs404/protocol/openid-connect/Login-status-iframe.html"		

II. Côté Gitlab

Se connecter en **SSH** dans le container ou serveur / instance booksack.

Modifier le fichier **'/etc/gitlab/gitlab.rb'**

Ajouter le code suivant en modifiant les données nécessaires :

```
gitlab_rails['omniauth_enabled'] =
true
gitlab_rails['omniauth_allow_single_sign_on'] = ['openid_connect']

gitlab_rails['omniauth_providers'] = [
  {
    name: "openid_connect",
    label: "Login with OAuth",
    args: {
      name: "openid_connect",
      scope: ["openid","profile","email"],
      response_type: "code",
      issuer: "https://auth.labs404.fr/realms/LABS404",
      discovery: true,
      client_auth_method: "query",
      uid_field: "preferred_username",
      pkce: true,
      client_options: {
```

```

    identifier: "<Client ID>",
    secret: "<Client Secret>",
    redirect_uri: "https://git.labs404.fr/users/auth/openid_connect/callback",
    gitlab: {
      groups_attribute: "groups",
      admin_groups: ["Admin"]
    }
  }
}
}
]

```

Recharger ensuite la configuration avec la commande :

```
gitlab-ctl reconfigure
```

```

(up to date)
[2024-09-25T09:44:19+00:00] INFO: templatesymlink[Create a gitlab.yml and create a symlink to Rails root] sending run action to execute[clear the gitlab-rails cache] (delayed)
Recipe: gitlab::gitlab-rails
  * execute[clear the gitlab-rails cache] action run[2024-09-25T09:44:58+00:00] INFO: execute[clear the gitlab-rails cache] ran successfully

  - execute /opt/gitlab/bin/gitlab-rake cache:clear
[2024-09-25T09:44:58+00:00] INFO: Cinc Client Run complete in 54.720750783 seconds

Running handlers:
[2024-09-25T09:44:58+00:00] INFO: Running report handlers
Running handlers complete
[2024-09-25T09:44:58+00:00] INFO: Report handlers complete
Infra Phase complete, 3/785 resources updated in 55 seconds
gitlab Reconfigured!
root@8842223d28deb7:/# vi /etc/gitlab/gitlab.rb

```