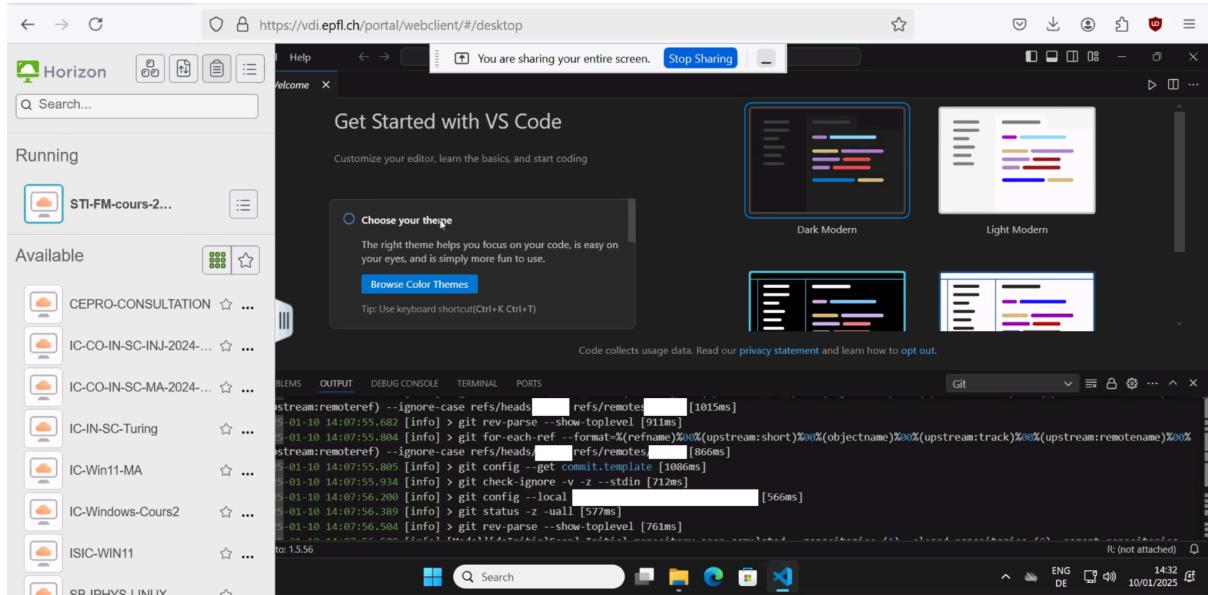


From GASPAR password to personal access token

<https://go.epfl.ch/gitlab-regained>

Scenario: you are logged in to the STI-FM-cours-2024 VM pool, and accessing your work using Visual Studio Code.



Problem: the password you set up in GitLab at the beginning of the school year (either your GASPAR password, or another one that you picked for this purpose) no longer seems to work.

Solution: read on.

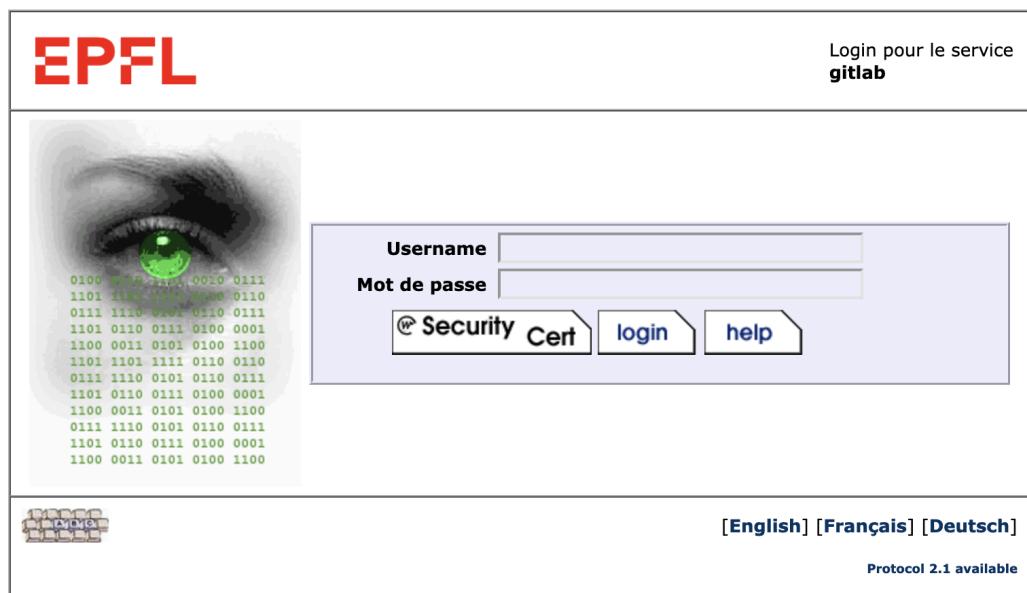
Step 1: create a personal access token

Source: https://gitlab.epfl.ch/help/user/profile/personal_access_tokens.md

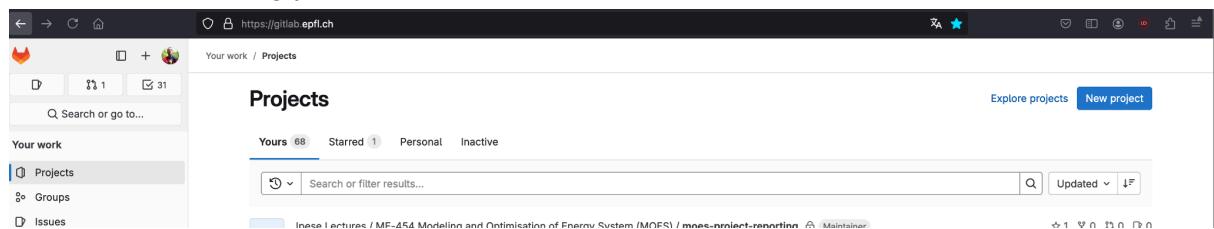
1. Navigate to <https://gitlab.epfl.ch/> and log in using your GASPAR credentials, if asked.

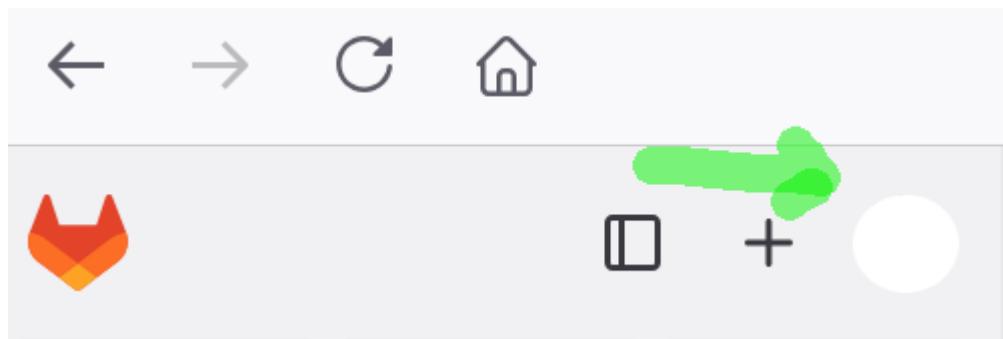


click the Tequila button, if necessary.



Fill out the form using your usual (so-called “GASPAR”) credentials.





2. Click on your avatar (round picture of you or abstract art) in the upper left corner as shown on the image above; then click Edit Profile

The image shows a mobile application interface for GitHub. At the top is a navigation bar with a logo on the left, and three icons on the right: a square, a plus sign, and a circular profile picture. Below the navigation bar is a search bar with a magnifying glass icon and the placeholder text "Search or go to...". The main content area is titled "User settings" and contains a list of options: "Profile" (indicated by a person icon), "Account" (indicated by a gear icon with an asterisk), "Applications" (indicated by a square icon with two dots), "Chat" (indicated by a speech bubble icon), "Access tokens" (indicated by a rectangle icon with three dots, and this item is highlighted with a blue vertical bar on the left), and "Emails" (indicated by an envelope icon).

Click Access tokens in the left-hand menu

Personal access tokens

You can generate a personal access token for each application you use that needs access to the GitLab API. You can also use personal access tokens to authenticate against Git over HTTP. They are the only accepted password when you have Two-Factor Authentication (2FA) enabled.

Active personal access tokens <small>0</small>					Add new token
Token name	Scopes	Created	Last Used <small>?</small>	Expires	Action
This user has no active personal access tokens.					

Click Add new token

Personal access tokens

Token name

For example, the application using the token or the purpose of the token.

Expiration date

 × Calendar icon

Select scopes

Scopes set the permission levels granted to the token. [Learn more.](#)

api
Grants complete read/write access to the API, including all groups and projects, the container registry, the dependency proxy, and the package registry.

read_api
Grants read access to the API, including all groups and projects, the container registry, and the package registry.

read_user
Grants read-only access to your profile through the /user API endpoint, which includes username, public email, and full name. Also grants access to /users.

create_runner
Grants create access to the runners.

manage_runner
Grants access to manage the runners.

k8s_proxy
Grants permission to perform Kubernetes API calls using the agent for Kubernetes.

read_repository
Grants read-only access to repositories on private projects using Git-over-HTTP or the Repository Files API.

write_repository
Grants read-write access to repositories on private projects using Git-over-HTTP (not using the API).

ai_features
Grants access to GitLab Duo related API endpoints.

sudo
Grants permission to perform API actions as any user in the system, when authenticated as an admin user.

admin_mode
Grants permission to perform API actions as an administrator, when Admin Mode is enabled.

read_service_ping
Grant access to download Service Ping payload via API when authenticated as an admin user

Set up a suitable name, expiration date and permissions as shown (“read_repository” and “write_repository” should suffice for permissions)

Create personal access token

click Create personal access token. Your token becomes available at the top of the screen:

Personal access tokens

You can generate a personal access token for each application you use that needs access to the GitLab API. You can also use personal access tokens to authenticate against Git over HTTP. They are the only accepted password when you have Two-Factor Authentication (2FA) enabled.

>Your new personal access token

.....



Make sure you save it - you won't be able to access it again.

3. Keep note of the token value, as instructed.

Step 2 : use the personal access token as a GitLab password from now on



💡 **You need to keep using your GASPAR login name** — Contrary to what the software on board the VM would have you believe, that field is **not** optional.