

Setting up Git Repository for 802-COM

Adam Gregory, Samtec

Richard Mellitz, Samtec

Kent Lusted, Synopsys

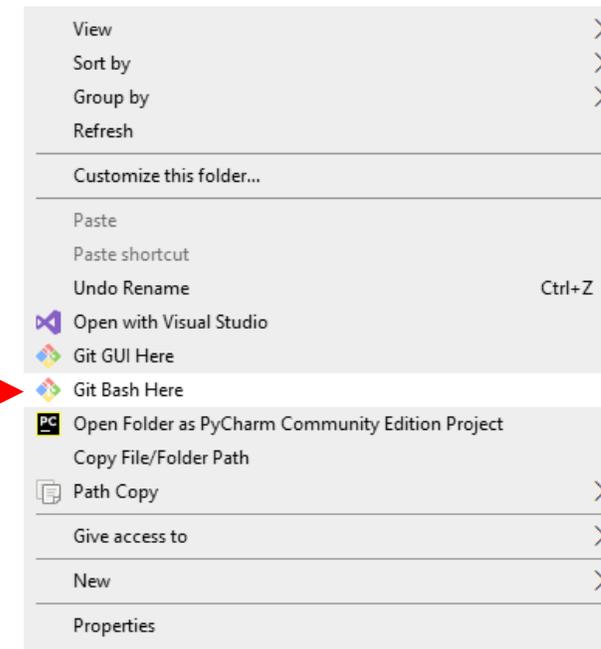
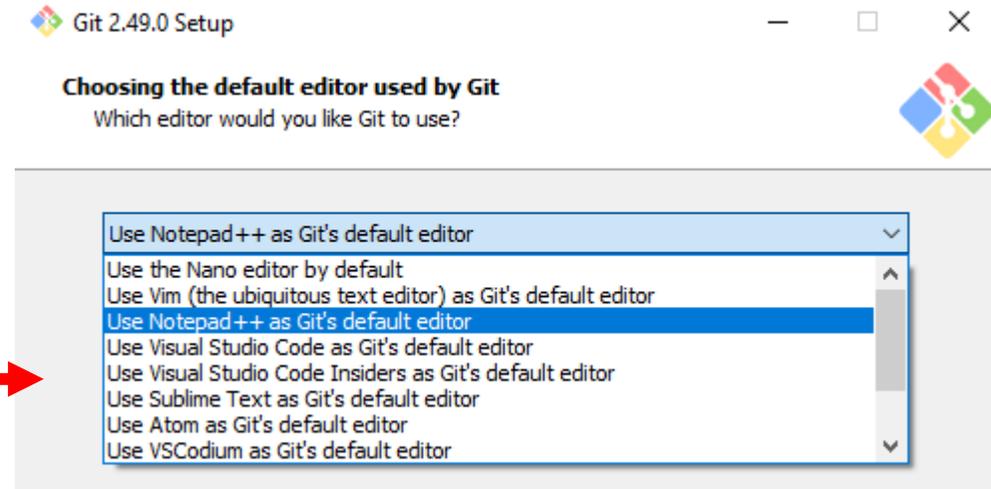
IEEE 802.3 Channel Operating Margin (COM) Open Source Project Ad Hoc,
New Orleans, LA May 2025 Interim

Summary

- Summary of steps to setup 802-COM GIT repository for viewing:
 1. Install GIT on your computer
 2. Login to IEEE Open Source website
 3. Create Personal Access Token
 4. Navigate to 802-COM Open-Source website
 5. Choose a location on your computer or network drive where the code will be downloaded
 6. Copy Clone URL
 7. Open GIT bash in the chosen folder and clone the repository
 8. Run COM
 9. To get future updates, pull the repository

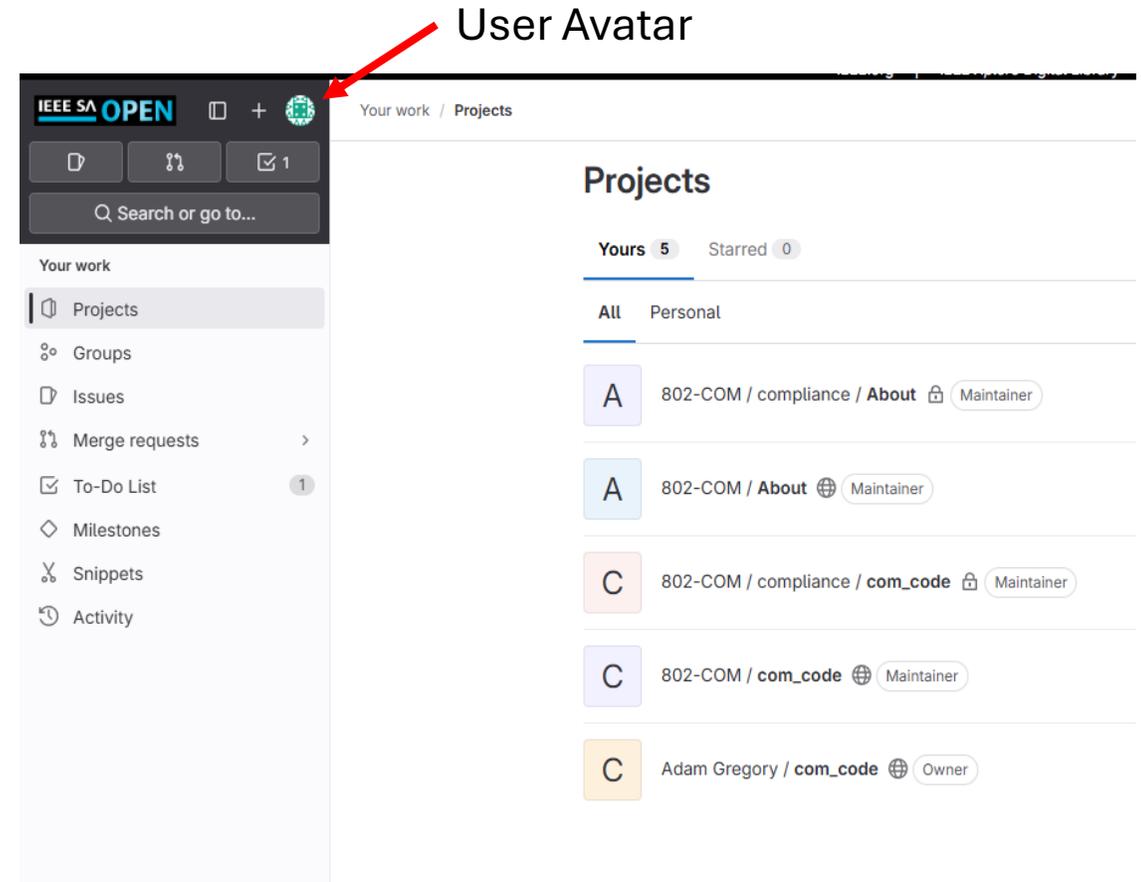
Step 1: Install GIT

- Installer URL:
 - <https://git-scm.com/downloads>
- In general, all the default options should be kept for installation
- **One suggestion: change the editor to Notepad++ or something similar. It is very awkward to use the default VIM editor if you are not familiar with it.**
- After installation, verify that you can open Git Bash. There are at least 3 ways:
 1. Usual install path: C:\Program Files\Git\git-bash.exe
 2. If on Windows, you can type “Git Bash” into Windows Search.
 3. You can also right click in a Windows Explorer window and select “Git Bash Here”



Step 2: Login to IEEE Open-Source Site

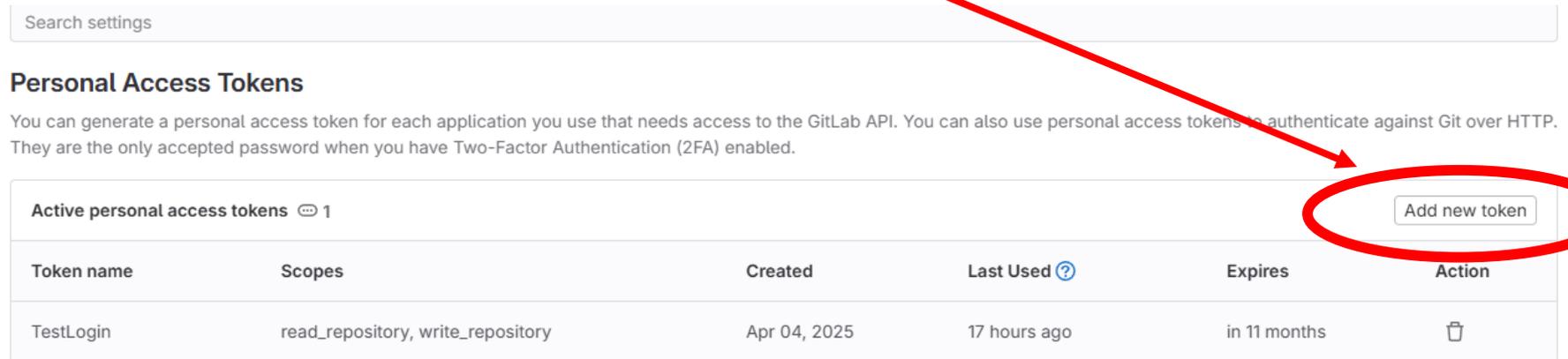
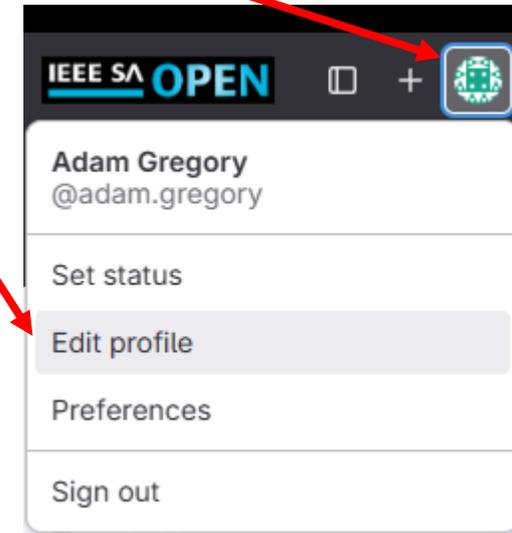
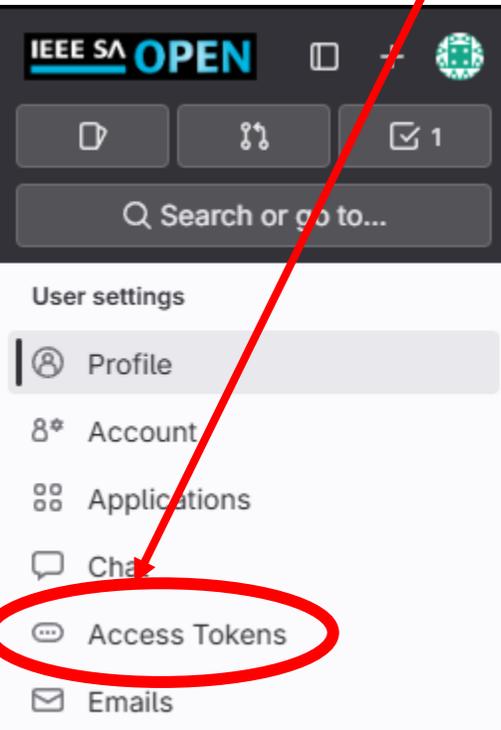
- Main page:
 - <https://opensource.ieee.org/>
- If you do not have an IEEE account, create one first:
 - [Create an IEEE Account](#)
- Go to the main Open-Source IEEE page to sign in:
 - [Sign in · GitLab](#)
- After signing in, you should see something like the image shown here.



Step 3: Create a Personal Access Token

- Click your user avatar in the top left, and select Edit Profile
- Choose the Access Tokens option in the left panel
- You can also use this direct link:
 - https://opensource.ieee.org/-/user_settings/personal_access_tokens
- Click the “Add new token” button
- Follow the steps on the next slide.

User Avatar



Step 3: Create a Personal Access Token

- Give the token a name, set an expiration date, and at a minimum check the following fields:
 - read_repository
 - write_repository
- The maximum expiration date can be set up to a year in the future.
- Click “Create personal access token”
- Refer to the next slide for details on saving the token

Add a personal access token

Token name

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

Expiration date

Select scopes

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

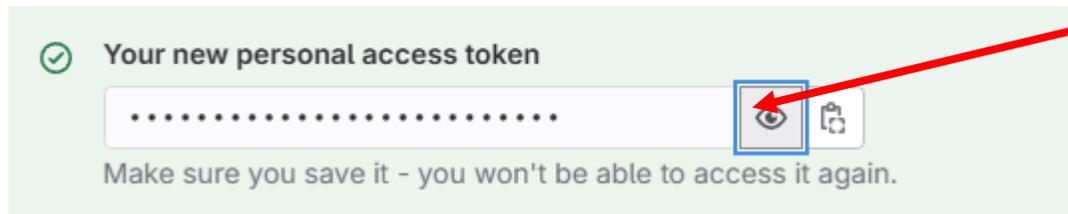
- api
Grants complete read/write access to the API, including all groups and projects, the container registry
- read_api
Grants read access to the API, including all groups and projects, the container registry, and the packa
- read_user
Grants read-only access to the authenticated user's profile through the /user API endpoint, which incl API endpoints under /users.
- create_runner
Grants create access to 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 Fil
- write_repository
Grants read-write access to repositories on private projects using Git-over-HTTP (not using the API).
- read_registry
Grants read-only access to container registry images on private projects.
- write_registry
Grants write access to container registry images on private projects. You need both read and write ac
- ai_features
Grants access to GitLab Duo related API endpoints.

Step 3: Create a Personal Access Token

- After creating the new Token, you will be returned to the personal access tokens page
- The dialog below showing the token will appear
- You must keep track of the token that is generated since it will be used as a password later
- You can copy it to your clipboard and paste it somewhere to save
- **Note: as the dialog states, you must save it now since you will not be able to access it again**

Personal Access Tokens

You can generate a personal access token for each application you use that need
They are the only accepted password when you have Two-Factor Authentication (



Click to view your token

Step 4: Navigate to 802-COM Website

- <https://opensource.ieee.org/802-com>
- Select the com_code project:
 - https://opensource.ieee.org/802-com/com_code

802-COM / com_code

The Auto DevOps pipeline has been enabled and will be used if no alternative CI configuration file is found.

Settings More information

C com_code

main com_code / +

History Find file Edit Code

revert SNDR ref commits
Adam Gregory authored 1 day ago c3aa888f

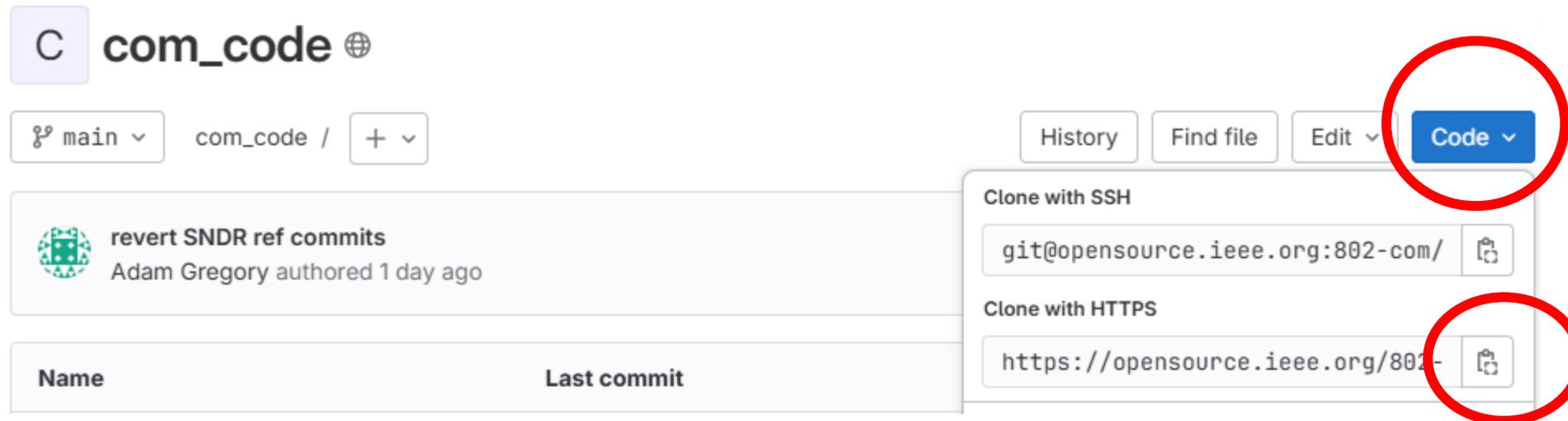
Name	Last commit	Last update
release	Initial Commit	1 week ago
src	revert SNDR ref commits	1 day ago
.gitignore	Initial Commit	1 week ago
AUTHORS.md	Initial Commit	1 week ago
CONTRIBUTING.md	Initial Commit	1 week ago
CONTRIBUTORS.md	Initial Commit	1 week ago
GOVERNANCE.md	Initial Commit	1 week ago
LICENSE	Initial Commit	1 week ago
README.md	Initial Commit	1 week ago

Step 5: Choose a location on your computer to download the git repository

- Can be any local folder or network folder.
- When the git repo is cloned, a new folder called “com_code” will appear
- Example:
 - You choose C:\COM_Open_Source\
 - After cloning (explained later in this document), you will have a folder at C:\COM_Open_Source\com_code\ containing the entire repository

Step 5: Copy Clone URL

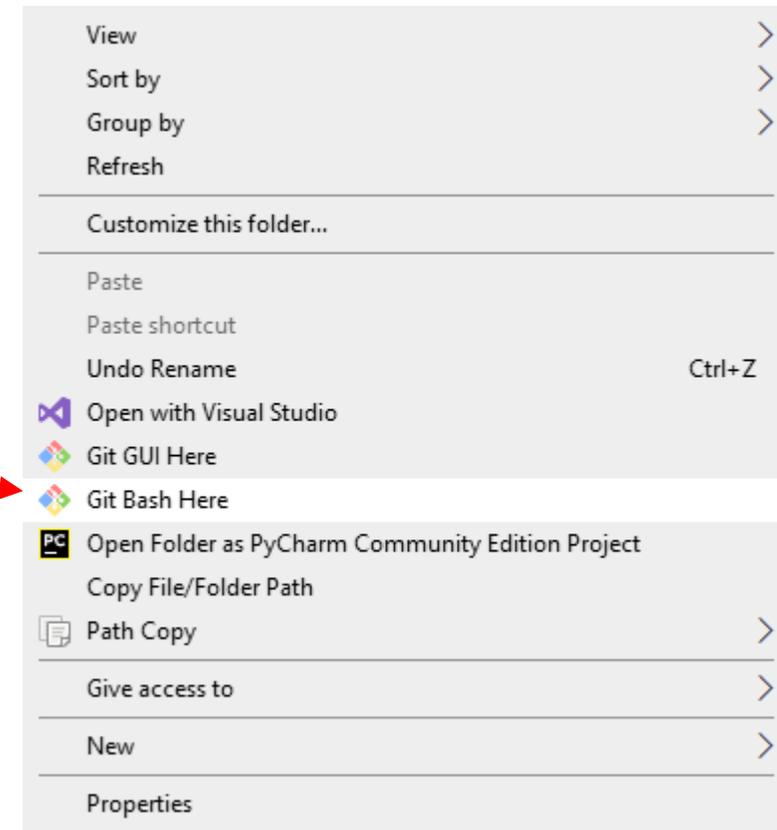
- Click the “Code” button and select the clone button under “Clone with HTTPS”
- Can also use this direct URL
 - https://opensource.ieee.org/802-com/com_code.git



The screenshot shows the GitHub interface for the repository 'com_code'. At the top, there is a 'C' icon and the repository name 'com_code' with a globe icon. Below this, there are navigation buttons for 'main', 'com_code /', and a '+' button. A commit card is visible, titled 'revert SNDR ref commits' by Adam Gregory, authored 1 day ago. Below the commit card is a table with columns 'Name' and 'Last commit'. On the right side, there are buttons for 'History', 'Find file', 'Edit', and 'Code'. The 'Code' button is circled in red. A dropdown menu is open under 'Code', showing 'Clone with SSH' and 'Clone with HTTPS' options. The 'Clone with HTTPS' option is also circled in red, and the copy icon next to the URL is circled in red.

Step 7: Clone GIT repository

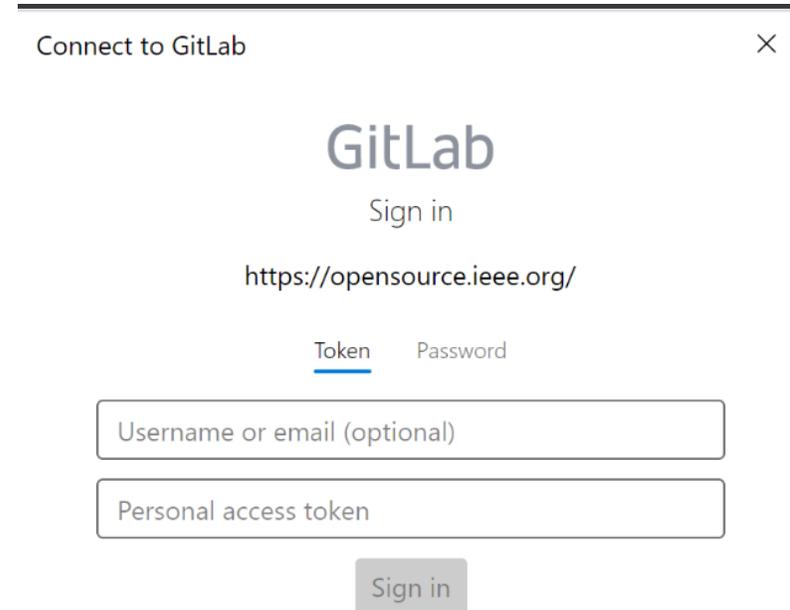
- Open Git Bash and navigate to the previously chosen folder. There are multiple ways to do this:
 - Option 1: Right click in Windows Explorer from the target directory and choose “Git Bash Here”
 - Option 2: Open Git Bash. Run: `cd <target folder>`
- Then run the following command to clone the git repository:
 - `git clone https://opensource.ieee.org/802-com/com_code.git`
- This will make the folder “com_code” in the target directory
- Run this command to place git bash working directory into the COM repository:
 - `cd com_code`



```
adamg@SCDC-ADAMGR MINGW64 /c/COM_Open_Source
$ git clone https://opensource.ieee.org/802-com/com_code.git
Cloning into 'com_code'...
remote: Enumerating objects: 356, done.
remote: Counting objects: 100% (176/176), done.
remote: Compressing objects: 100% (118/118), done.
remote: Total 356 (delta 132), reused 73 (delta 58), pack-reused 180
Receiving objects: 100% (356/356), 408.74 KiB | 12.02 MiB/s, done.
Resolving deltas: 100% (221/221), done.
```

Step 7: Clone GIT repository (Permission and SSL issues)

- There are 2 potential issues to work through when running git clone for the first time
 1. SSL certificate problem:
 - If you get an error message regarding SSL, run the following command in git bash:
 - `git config --global http.sslbackend schannel`
 2. Login Prompt:
 - If you haven't run a git command with your current personal access token, you will likely need to log in with that token. Sometimes you may have to run git clone twice to make it show up, but eventually a popup should appear requesting your username and password
 - Your username is your login email address or IEEE user name
 - Your password is your personal access token (See the previous slides. Make sure to save your access token)
 - After you initially login with your token, you should never have to login again through git until your token expires. At that point, you will need to repeat this git login process.



The screenshot shows a 'Connect to GitLab' dialog box with a close button (X) in the top right corner. The GitLab logo is centered, with 'Sign in' below it. The URL 'https://opensource.ieee.org/' is displayed. There are two tabs: 'Token' (selected) and 'Password'. Below the tabs are two input fields: 'Username or email (optional)' and 'Personal access token'. A 'Sign in' button is located at the bottom right of the form.

Step 7: Clone GIT repository

- After cloning, open your com_code folder
- You should see the directory structure shown below
- The .git folder contains all the Git tracking information. It should not be deleted or edited. If deleted, the repository will become static and have no way of communicating with the remote repository for updates.
 - If you wanted to create a static workspace, you could run a git clone followed by a delete of the .git folder. This would be equivalent to just downloading the code.

Name	Date modified	Type	Size
.git	4/22/2025 4:02 PM	File folder	
release	4/22/2025 3:00 PM	File folder	
src	4/22/2025 3:00 PM	File folder	
.gitignore	4/22/2025 3:00 PM	Text Document	1 KB
AUTHORS.md	4/22/2025 3:00 PM	Markdown Source...	1 KB
CONTRIBUTING.md	4/22/2025 3:00 PM	Markdown Source...	5 KB
CONTRIBUTORS.md	4/22/2025 3:00 PM	Markdown Source...	1 KB
GOVERNANCE.md	4/22/2025 3:00 PM	Markdown Source...	4 KB
LICENSE	4/22/2025 3:00 PM	File	2 KB
README.md	4/22/2025 3:00 PM	Markdown Source...	4 KB

Step 8: Run COM

- Open Matlab and set your working directory to the “src” folder
 - The full path will be: <Target Directory>/com_code/src
- If you are familiar with COM as a single file, note that every subfunction is now a separate file.
- The main file is called com_ieee8023_
- This can be run as normal:
 - `Result = com_ieee8023_(config_file,0,0,channel_file);`
- There is also a “release” folder. This holds the familiar distribution files where all the subfunctions are contained in a single file.

Step 9: Pull the repository (in the future)

- Cloning the repository provides you with the current state of 802-COM.
- Future updates to the main repository are never reflected in your local repository until you pull them down. Remember that you only do a clone once. All future updates are done by pull.
- To get the latest updates, open Git Bash in the `com_code` directory
- Run this command:
 - `git pull`
- If there are no updates, it will say “Already up to date”. Otherwise it will provide a summary of which files were updated (changed, added, deleted)

```
adamg@SCDC-ADAMGR MINGW64 /c/COM_Open_Source/com_code (main)
$ git pull
Already up to date.
```

Repository Edits

- You should not edit files in your local repository if you intend to pull updates from the remote repository. If a file you have edited is updated on the remote repository, the git pull will fail. If you want to modify files for personal testing purposes, you should instead copy the data to a location that is not tracked by git.
- If you intend to collaborate on 802-COM and want your updates to be incorporated into COM, refer to the 802-COM Contribution document. It provides details on making a Forked Repository and creating Feature Branches. This is the method for providing updates which will eventually be submitted as Merge Requests in the 802-COM repository.
- **There is likely no circumstance in which a user will ever directly update files in the 802-COM repository**