
Git sous Windows

par Louis "Lozul" Gasnault

*La v1 de ce document fut écrite à l'aube d'un samedi pour aider les 2023 toulousains qui ne savaient pas se servir de Git la veille d'un rendu de tp.
Voici la v2 : toujours le même but, moins de faute et ... écrit en L^AT_EX*

Contact : louis.gasnault@epita.fr, Lozul#1433 sur Discord

Contents

1	Introduction	3
2	Git Bash	4
2.1	Télécharger Git pour Windows	4
2.2	Présentation de Git Bash :	4
2.3	Configurer Git :	4
2.4	Relier votre ordinateur au cri :	5
3	Conclusion	7

1 Introduction

D'après notre ami Wikipédia : Git est un logiciel de gestion de versions décentralisé. Créé par Linus Torvalds, le papa de Linux, Git vous permettra de gérer les différentes versions d'un dossier contenant des fichiers, le tout possiblement à plusieurs. Ce tuto est là pour vous aider à installer Git sous Windows 7 et +, toutefois, il est important que vous sachiez qu'au fil de l'année vous serez amené à utiliser des outils plus complexes à faire fonctionner sous Windows. Virtual Machine ou Dual Boot, à vous de voir, mais vous aurez besoin de pouvoir travailler sous Linux depuis chez vous.

Quelques informations :

- Les dossiers gérés par Git sont appelés des **repositories** (repo pour les intimes).
- Il existe plusieurs services en lignes qui peuvent héberger des repo (GitHub, GitLab etc...). Dans votre cas c'est EPITA qui héberge les différents repo dont vous vous servirez tout au long de l'année.
- Je vais vous présenter une solution simple (si si, je vous le jure) pour utiliser Git, via un terminal, sous Windows. Il en existe plusieurs autres que vous pouvez découvrir en quelques recherches internet.
- Petites options à activer pour vous aider : dans l'explorateur de fichier, cliquez sur l'onglet *Affichage*. Dans la section *Afficher/Masquer*, cochez *Extensions de noms de fichiers* et *Éléments masqués*.
- Ne recopiez pas les \$ avant les commandes.
- Remplacez toute occurrence de NOM par le nom de votre session Windows.

2 Git Bash

2.1 Télécharger Git pour Windows

Rendez vous à l'adresse : <https://git-scm.com/downloads>, cliquez sur *Windows*, le téléchargement débute automatiquement. Il s'agit d'un installateur classique, exécutez le et cliquez sur *Suivant* jusqu'à pouvoir *Installer*. (Ne changez les paramètres que si vous savez ce que vous faites !)

2.2 Présentation de Git Bash :

Trois nouveaux logiciels sont disponibles dans votre menu Démarrer (touche Windows) sous le dossier Git. Celui qui vous servira se nomme Git Bash. Il s'agit d'un terminal, fonctionnant à peu près comme un terminal Linux.

2.3 Configurer Git :

Avant de pouvoir utiliser git, vous allez devoir vous présenter.

1. Lancez Git Bash
2. Entrez la commande :

```
$ git config --global user.name "Prénom Nom"
```

3. Entrez la commande :

```
$ git config --global user.email "prénom.nom@epita.fr"
```

2.4 Relier votre ordinateur au cri :

Pour pouvoir éditer vos repo, vous devez lier votre ordinateur au serveur d'EPITA (cri.epita.fr).

1. Lancez/Retournez sur Git Bash
2. Entrez les commandes suivantes :

```
$ cd /c/Users/NOM/  
$ mkdir .ssh
```

3. Entrez la commande :

```
$ ssh-keygen -f "/c/Users/NOM/.ssh/id_rsa" -b 4096
```

4. Entrez une "passphrase" (il s'agit du mot de passe que vous entrez à chaque 'push')

Dans votre dossier *.ssh* ce trouve maintenant deux fichiers : *id_rsa* et *id_rsa.pub*. Ils contiennent tous les deux une clef, le fichier ".pub" contient la clef publique. Vous ne devez jamais communiquer celle qui se trouve dans le fichier sans extension ! Cela compromettrait l'intégrité de vos repo.

Vous devez maintenant enregistrer votre clef publique sur le serveur d'EPITA.

1. Ouvrez le fichier *id_rsa.pub* avec le bloc-note, emacs, ou autre
2. Copiez le gros pavé de texte (Ctrl + C, Ctrl + A)
3. Connectez vous sur le site <https://cri.epita.fr/>

4. Allez dans la section *My SSH-Keys* et cliquez sur le + (à droite)
5. Collez votre clef (Ctrl + V) et cliquez sur *Add Key*

C'est fini ! Vous pouvez maintenant communiquer avec les serveurs d'EPITA via Git.

3 Conclusion

Voici quelques liens utiles pour en apprendre toujours plus :

- Le site de Mr. Roussille avec de nombreux tutos qui pourront vous sauver la vie <http://prof.roussille.io/>
- Littéralement ce tuto, en anglais, pour Linux : <https://doc.cri.epita.fr/git/>
- RTFM pour moins cher que gratuit : <https://git-scm.com/book/en/v2>
- Un guide en français des commandes basiques d'un terminal Linux : <http://juliend.github.io/linux-cheatsheet/>
- Une liste non exhaustive de problèmes et de leur solution : <https://ohshitgit.com/>

