Mise en place

Installation

Git

Installer :

- Git for Windows
- TortoiseSVN

tree

Utilisation commande Windows

Pour utiliser la commande tree Windows créer l'alias suivant dans le fichier /c/Users/<login>/.bashrc :

```
alias tree='cmd //c tree //a //f'
```

Installer la commande linux

- Aller sur le site gnuwin32
- Choisir le package tree
- Télécharger le zip correspondant au « binaries »
- Extraire le fichier tree.exe du répertoire bin et le mettre dans C:\Program Files\Git\usr\bin

zip

- Télécharger le fichier zip-3.0-bin.zip sur le site https://sourceforge.net/projects/gnuwin32/files/zip/3.0
- Copier le fichier bin/zip.exe dans le répertoire C:\Program Files\Git\mingw64\bin
- Télécharger le fichier bzip2-1.0.5-bin.zip sur le site https://sourceforge.net/projects/gnuwin32/files/bzip2/1.0.5
- Copier le fichier bin/bzip2.dll dans le répertoire C:\Program Files\Git\mingw64\bin
- Lancer la commande **`zip`** dans une console Git Bash pour vérifier que tout est bien installé

rsync

- Télécharger les dernières versions disponibles de rsync, libzstd et libxxhash sur le site https://repo.msys2.org/msys/x86_64/
- Décompresser les archives zst puis extraire le contenu

Utiliser zstd en suivant ce tuto https://shchae7.medium.com/how-to-use-rsync-on-git-bash-6c6bba6a03ca

```
zstd -d rsync-3.2.6-1-x86_64.pkg.tar.zst libzstd-1.5.2-2-x86_64.pkg.tar.zst
libxxhash-0.8.1-1-x86_64.pkg.tar.zst
tar xvf rsync-3.2.6-1-x86_64.pkg.tar
tar xvf libzstd-1.5.2-2-x86_64.pkg.tar
tar xvf libxxhash-0.8.1-1-x86_64.pkg.tar
```

- Copier le fichier usr/bin/rsync.exe sous C:\Program Files\Git\usr\bin
- Copier le fichier usr/bin/msys-xxhash-0

Configuration sur le PC

Personnalisation prompt

Modification de la variable PS1 dans le fichier C:\Program Files\Git\etc\profile.d\git-prompt.sh

Personnalisation vi

Créer le fichier ~/.vimrc ou bien directement dans le fichier C:\Program Files\Git\etc\vimrc pour indiquer les options.

Clef SSH

- Lancer une console GitBash
- Générer une clef SSH

```
ssh-keygen -t rsa -b 4096 -C "adresse-mail@altran.com"
```

Paramètres Git

Définition des paramètres

```
git config --global user.name "NOM prenom"
git config --global user.email prenom.nom@truc.fr
```

Format dos/unix

Pour ne pas changer le format LF en CR/LF lors du commit à partir d'une console Git/Bash

```
git config --global core.autocrlf false
```

Vérification des paramètres

```
git config --global -l
```

> Les paramètres sont stockés dans le fichier C:\Users\<login>\.gitconfig ou ~/.gitconfig sous Linux

Configuration difftool/mergetool

Il est possible d'utiliser un outil graphique pour faire la différence entre 2 fichiers. Outils possibles : opendiff kdiff3 tkdiff xxdiff meld kompare gvimdiff diffuse diffmerge ecmerge p4merge araxis bc codecompare smerge emerge vimdiff nvimdiff

vimdiff

```
git config --global diff.tool vimdiff
git config --global difftool.prompt false   # pour éviter un prompt à chaque fois
```

kdiff3

```
git config --global --add diff.guitool kdiff3
git config --global --add difftool.kdiff3.path "C:/Program Files/KDiff3/kdiff3.exe"
git config --global --add difftool.kdiff3.trustExitCode false
git config --global difftool.prompt false

git config --global --add merge.guitool kdiff3
git config --global --add mergetool.kdiff3.path "C:/Program Files/KDiff3/kdiff3.exe"
git config --global --add mergetool.kdiff3.trustExitCode false
git config --global mergetool.prompt false
```

examdiff

```
git config --global diff.tool=examdiff
git config --global difftool.examdiff.cmd="C:/Program Files (x86)/Launcher/Examdiff/examdiff.exe" "$LOCAL" "$REMOTE"
git config --global difftool.prompt=false
```

Avec examdiff, une fois la commande git difftool lancée, lancer à nouveau Examdiff à partir de Windows pour comparer les 2 fichiers mémorisés dans l'historique. Une fois terminé il faudra faire un ps -ef puis kill du process examdiff à partir d'un autre git bash

---

Configuration sur le serveur Git

Ajout de la clef ssh

- Se connecter sur le serveur Git
- Cliquer en haut à droite sur la roue pour accéder au menu « Edit Profile ».
- Choisir dans le menu à gauche « SSH Keys ».
- Copier/coller la clef à partir du fichier C:\Users\login\.ssh\id_rsa.pub
- Attention pas de retour à la ligne
- Le champ Title doit se remplir automatiquement
- Cliquer sur le bouton Add Key

Raccourcis

Il est possible de créer des raccourcis sur le bureau pour exécuter directement des commandes ou scripts linux. Pour cela :

- Créer un raccourci sur le bureau vers le programme C:\Program Files\Git\usr\bin\mintty.exe
- Modifier la cible de la façon suivante :

```
"C:\Program Files\Git\usr\bin\mintty.exe" --exec "/usr/bin/bash" --login -i
-c "echo 'Hello World!'; read"
```

Source

API Gitlab

Génération token

- Cliquer sur l'icône de son utilisateur
- Choisir « Edit Profile »
- Cliquer sur « Access token » dans le menu de gauche
- Cliquer sur le bouton « Add new token »
- Donner un nom au token puis choisir une date de validité pour le token
- cocher « read_api » et « read_repository »

- Cliquer sur le bouton « Create personnal access token »

Bien noter immédiatement le token qui ne pourra plus être visualisé ensuite.

Utilisation en lecture

Interrogation de la liste des projets

```
curl --header "PRIVATE-TOKEN: <token>"
"https://<url-git>/api/v4/projects?per_page=100" > liste-projet.json
```

Interrogation projet

A partir de l'id projet récupéré avant :

```
curl --header "PRIVATE-TOKEN: <token>"
"https://<url-git>/api/v4/projects/<id project>" > projet.json
```

Liste des commits du projets

```
curl --header "PRIVATE-TOKEN: <token>"
"https://<url-git>/api/v4/projects/<id project>/repository/commits" >
liste_commits.json
```

Accès complet par token

Penser à positionner **read_repository** et **write_repository** lors de la création du token

```
bash
```

Configurer l'url

```
git remote set-url origin https://<username>:<token>@<url git>
```

Si le username est une adresse mail il faut remplacer le caractère @ par son code **%40**

Vérification url

```
git remote -v
```

From:
https://wiki.iot-acs.fr/ - **Wiki**

Permanent link:
https://wiki.iot-acs.fr/doku.php?id=all:bibles:applications:git:installation

Last update: **2025/11/05 08:10**