

# Utilisation du systeme de rendus du Koalab - Manuel de l'etudiant

## Notes importantes

Ce guide presume que vous connaissez le nom du depot que vous cherchez a utiliser. Dans tous les cas, la forme du nom d'un depot est <nom du projet>-<promo>-<login du leader>. Le nom du depot est suffixe de ".git" dans le cas d'un depot Git.

Ce guide presume egalement, dans TOUS LES CAS sous BSD/Linux, que votre cle privee SSH se trouve dans ~/.ssh/id\_rsa, et est donc la cle utilisee par default par SSH.

Ce guide presume enfin que vous savez vous servir de git, svn, et mercurial, et que vous savez utiliser WinSCP ou autre utilitaire de SCP sur Windows (Pour recuperer sur votre windows vos cles SSH du PIE). Dans le cas contraire, ce guide presume que vous savez lire une documentation ...

Ce guide explique, pour chaque SCM propose par le Koalab, la marche a suivre pour se connecter a un depot a l'aide des outils svn, hg et git en ligne de commande sous BSD/Linux, ainsi que TortoiseSVN, TortoiseGit et TortoiseHg sous Windows. Tout autre utilitaire ne fera pas l'objet d'une documentation et son utilisation se fera a VOS risques et perils. Veuillez donc a vous conformer a ce guide si vous n'avez pas envie d'experimenter. Vous etes prevenus.

Ce guide et tous les outils qu'il utilise ont ete testes avec succes sur Windows XP/Vista/Seven (32 et 64 bits), FreeBSD/Debian/Archlinux.

## Enregistrement de votre cle SSH

Rendez vous sur <http://koala-scm.epitech.net/scm-register> . Authentifiez vous a l'aide de votre mot de passe PPP, et utilisez le formulaire qui vous est presente pour uploader votre cle SSH publique.

Pour creer une cle SSH (sur le PIE) : Se loguer, taper ssh-keygen, choisir un emplacement de sauvegarde (Par default ~/.ssh), et une passphrase (eventuellement). La cle privee est ~/.ssh/id\_rsa, la cle publique est ~/.ssh/id\_rsa.pub. C'est cette derniere que vous devez uploader.

Note importante : Si le systeme d'enregistrement vous dit que votre cle est invalide, ouvrez votre cle sous vim ou autre et verifiez : Si elle ne commence pas par "ssh-rsa" ou "ssh-dss", elle est invalide. Je pense notemment a ceux qui ont uploadé une cle au format .ppk (Cle PuTTY) : Ces dernieres NE MARCHERONT PAS. Le systeme refusera egalement les cles RSA d'une longueur inferieure a 2048 bits (Longueur par default sur le PIE. En cas de probleme, forcer la longueur avec "ssh-keygen -b 2048")

Egalement, veuillez noter que vous ne pouvez avoir qu'une cle SSH a la fois sur notre systeme. Si vous en enregistrez une autre, elle remplacera l'ancienne. Ainsi, si vous voulez avoir acces au systeme depuis plusieurs machines, copiez votre paire de cles d'une machine a l'autre.

# Mise en place de votre cle SSH pour l'utilisation avec les outils SCM

## BSD/Linux

Assurez vous simplement que la cle SSH privee associee a la cle publique que vous avez enregistre sur notre serveur soit : `~/.ssh/id_rsa` (C'est le cas par default.)

## Windows

Vous aurez besoin des outils suivants:

- WinSCP
- putty
- plink
- pageant
- puttygen

Commencez par utiliser WinSCP ou assimile pour recuperer votre cle SSH privee. Ouvrez ensuite puttygen, allez dans le menu Conversions, et selectionnez votre cle privee. Cliquez ensuite sur "Save private key", et sauvegardez la sur votre disque. A partir de maintenant, chaque reference a votre "cle privee" sera une reference a la cle que vous venez de sauver (format .ppk).

A partir de maintenant, deux solutions:

### Avec Pageant

Lancez pageant. Double cliquez sur l'icone de pageant dans le systray pour ouvrir l'interface, cliquez sur Add key, et selectionnez votre cle privee. Le soft se servira automatiquement de cette cle quand vous interagirez avec le depot. Par contre, si vous partagez le PC avec un binome, ca devient plus problematique, car Pageant utilisera la premiere cle qui marche avec le depot selectionne, et vous serez donc toujours authentifie en tant que la personne proprietaire de ladite cle. A n'utiliser donc que si vous etes seul sur la machine.

### Avec PuTTY

Lancez putty, puis remplissez les champs suivants

- Session > Host Name : `koala-scm.epitech.net`
- Session > Connection Type : SSH
- Connection > Data > Auto-login username : `svn/git/hg` (Selon le SCM souhaite)
- Connection > SSH > Auth > Private key file for authentication : `<votre cle privee>`
- Retournez dans la page Session, entrez un nom dans le champ sous Saved Sessions (par exemple `svn/git/hg_koala_login_x`), puis cliquez sur Save.

Ensuite, au lieu d'utiliser `svn/hg/git@koala-scm.epitech.net` pour vous connecter, vous utiliserez le nom de la connection que vous venez de creer, et le soft utilisera les parametres que vous avez choisi.

Cette solution permet de sauvegarder autant de reglages que vous voulez, si vous etes plusieurs a partager la machine

# SVN

## BSD/Linux

Checkout le depot :

```
zaz@blackjack:~/scm/$ svn checkout svn+ssh://svn@koala-scm.epitech.net/<NOM DU REPO>
Checked out revision 0.
zaz@blackjack:~/scm/$ cd test-2011-montin_m
zaz@blackjack:~/scm/test-2011-montin_m$
```

Travaillez ensuite comme d'habitude sur votre depot.

## Windows

Vous aurez besoin de TortoiseSVN.

Pour checkout votre depot SVN, cliquez droit dans un dossier, puis SVN Checkout. Dans URL of repository, entrez : svn+ssh://svn@koala-scm.epitech.net/<NOM DU REPO> (ou svn+ssh://nom\_de\_la\_connection/<NOM DU REPO> si vous n'utilisez pas Pageant), puis cliquez sur OK.

Travaillez ensuite comme d'habitude sur votre depot.

# Mercurial

## BSD/Linux

Clonez le depot:

```
zaz@blackjack:~/scm/$ hg clone ssh://hg@koala-scm.epitech.net/test-2011-montin_m
destination directory: test-2011-montin_m
requesting all changes
adding changesets
adding manifests
adding file changes
added 1 changesets with 1 changes to 1 files
1 files updated, 0 files merged, 0 files removed, 0 files unresolved
zaz@blackjack:~/scm/$ cd test-2011-montin_m
zaz@blackjack:~/scm/test-2011-montin_m$
```

Travaillez ensuite comme d'habitude sur votre depot.

## Windows

Vous aurez besoin de TortoiseHg.

Pour cloner votre depot Mercurial, cliquez droit dans un dossier, puis TortoiseHg > Clone a repository.

Dans "Source path", entrez : ssh://hg@koala-scm.epitech.net/<NOM DU REPO>, puis

cliquez sur Clone.

Travaillez ensuite comme d'habitude sur votre depot.

Note IMPORTANTE: N'oubliez surtout pas, pour synchroniser VOTRE depot avec NOTRE depot, de pusher vos changements sur le serveur.

Pour ce faire, clic droit dans votre depot, puis TortoiseHg > Synchronize, puis cliquez sur Push.

Lors de votre soutenance, c'est NOTRE depot qui fait foi : Si vous n'avez pas push votre travail, tant pis pour vous.

## Git

### BSD/Linux

Votre depot n'est pas cree directement. Vous devez le creer vous-memes, puis l'uploader sur le serveur.

Procedez comme suit :

Creez le depot:

```
zaz@blackjack:~$ mkdir test-2011-montin_m
zaz@blackjack:~$ cd test-2011-montin_m
zaz@blackjack:~/test-2011-montin_m$ git init
Initialized empty Git repository in /home/zaz/test-2011-montin_m/.git/
zaz@blackjack:~/test-2011-montin_m$ git remote add origin git@koala-scm.epitech.net:test-2011-montin_m.git
```

Faites un commit sur le depot, puis :

```
zaz@blackjack:~/test-2011-montin_m$ git push origin master
```

Votre depot est maintenant cree. Vous pouvez le cloner et travailler normalement :

```
zaz@blackjack:~$ git clone git@koala-scm.epitech.net:test-2011-montin_m.git
Initialized empty Git repository in /home/zaz/test-2011-montin_m/.git/
warning: You appear to have cloned an empty repository.
zaz@blackjack:~$ cd test-2011-montin_m/
zaz@blackjack:test-2011-montin_m$
```

Veuillez noter qu'a la soutenance, nous ramasserons la branche "master", et pas une autre.

### Windows

Vous aurez besoin de TortoiseGit, qui depend de MSysGit.

Vous pouvez trouver MSysGit sur <http://code.google.com/p/msysgit/downloads/list>

(Utilisez le premier download, "Official Git", et non pas le download nomme MSysGit ...)

Faites bien attention lors de l'installation a choisir l'option "Run Git from Command line" et non pas "Use Git Bash only", sinon TortoiseGit ne fonctionnera pas. Si vous avez envie d'utiliser Git en ligne de commande et non pas forcément depuis Tortoise, choisissez également d'utiliser PLink et non OpenSSH lors de l'installation.

Une fois TortoiseGit installe et configure (Clic droit > TortoiseGit > Settings, dans MSysGit path entrer C:\Program Files\Git\bin, ou le dossier bin du repertoire d'install de Git), vous pouvez commencer.

Comme sous Linux, vous devez creer votre depot avant de travailler. Creez un repertoire

vide, clic droit, puis Git Create repository here.

Dans les propriétés de votre depot, ajoutez git@koala-scm.epitech.net:<NOM DU REPO>.git dans les remotes (Clic droit, TortoiseGit > Settings, puis Git > Remote a gauche. Puis clic sur Add new, mettre l'URL, puis entrée.)

Faites ensuite un commit, puis push sur le serveur (clic droit dans votre depot, puis TortoiseGit > Push).

Pour cloner le depot, clic droit > Git Clone, et dans Url, entrez : git@koala-scm.epitech.net:<NOM DU REPO>.

Note IMPORTANTE: N'oubliez surtout pas, pour synchroniser VOTRE depot avec NOTRE depot, de pusher vos changements sur le serveur.

Pour ce faire, clic droit dans votre depot, puis TortoiseGit > Push.

Lors de votre soutenance, c'est NOTRE depot qui fait foi : Si vous n'avez pas push votre travail, tant pis pour vous.

Veuillez noter qu'à la soutenance, nous ramasserons la branche "master", et pas une autre.