

Rapport Final SAE R2.03

- Equipe: Quentin GUIOT, Shems PETREMAND, Hugo JIMENEZ
- Moodle: <https://moodle.univ-lille.fr/course/view.php?id=30827§ionid=266881>
- Referent: Jean Carle - jean.carle@univ-lille.fr

Table des matieres

0. **Introduction**
1. **Preparation de l'environnement virtuel** 1.1 - Preparation d'une machine virtuel Debian 1.2 - Installation de l'OS 1.3 - Acces sudo pour user
2. **Installation de Git** 2.1 - Configuration globale de Git
3. **Installation de Gitea** 3.1 - Installation du binaire 3.2 - Preparation de l'environnement Git 3.3 - Lancement de Gitea 3.4 - Parametrage de Gitea 3.5 - Utilisation de Gitea
4. **Conclusion**

Introduction

Dans le cadre de la SAE-203 Réseau, notre équipe s'est exercée à l'installation et configuration d'un poste de travail type, incluant les outils de bases nécessaires ou favorables au confort d'utilisation, ainsi qu'à la préparation d'un environnement de versionning, ici Gitea. Au sein de ce rapport, nous reviendrons sur les différentes étapes de cette mise en œuvre, en appuyant sur des aspects et questions techniques ainsi que sur les éventuelles difficultés rencontrées. Le rapport est appuyé d'images, majoritairement des screenshots issus des diverses installations afin de démontrer le bon déroulé du processus.

1. Preparation de l'environnement virtuel

1.1 - Preparation d'une machine virtuel Debian

- *Question(s)* 1. Configuration matérielle dans VirtualBox

Que signifie "64-bit" dans "Debian 64-bit" ?

Dans "Debian 64-bit", le "64-bit" correspond à la taille des registres du processeur.

Un processeur 64 bits est favorisé par rapport à un processeur 32 bits car il permet de traiter de grandes quantités de mémoire plus efficacement avec une mémoire de 4 Go, 8 Go, 16 Go ou plus contrairement au processeur 32 bits qui lui a une mémoire vive de 2 Go et un espace disque de 20 Go maximum.

Quelle est la configuration réseau utilisée par défaut ?

Par défaut, lors de l'installation de la machine virtuelle nous avons configuré le miroir Debian de 'polytech-lille' et le proxy associé.

Quel est le nom du fichier XML contenant la configuration de votre machine ?

Le fichier contenant la configuration de la machine se nomme `SAE203.vbox`.

Sauriez-vous le modifier directement ce fichier pour mettre 2 processeurs à votre machine ?

Il est possible de modifier ce fichier en ajoutant `count='2'` dans `<Hardware>` à la suite de `CPU` → `<CPU count='2'>`

“—” ### 1.2 - Installation de l'OS

- **Question(s) 2. Installation OS de base**

Qu'est-ce qu'un fichier iso bootable ?

Un fichier ISO bootable est un fichier image qui peut être utilisé pour créer un disque de démarrage ou une clé USB de démarrage, qui est souvent utilisée pour installer un système d'exploitation, tels que Windows, Linux ou MacOS, sur un ordinateur ou pour exécuter des programmes de récupération système.

Qu'est-ce que MATE ? GNOME ?

GNOME est un environnement de bureau open-source pour les systèmes d'exploitation de type Unix, qui fournit une interface utilisateur graphique conviviale et des fonctionnalités utiles pour les utilisateurs de ces systèmes.

MATE est également un environnement de bureau open-source, basé sur GNOME. Il fournit une gamme complète de fonctionnalités, applications et options de personnalisation pour les utilisateurs de ces systèmes.

Qu'est-ce qu'un serveur web ?

Un serveur web est un logiciel qui stocke et distribue des fichiers sur Internet, comme des pages web, des images, des vidéos, etc. Il répond aux demandes des navigateurs web pour afficher ces fichiers en utilisant le protocole HTTP.

Qu'est-ce qu'un serveur ssh ?

Un serveur SSH est différent d'un serveur car le serveur web est conçu pour stocker et diffuser des fichiers sur Internet, tandis que le serveur SSH permet d'établir une connexion sécurisée à distance à un ordinateur ou un serveur.

Qu'est-ce qu'un serveur mandataire ?

Un serveur mandataire est un serveur informatique qui a pour fonction de relayer des requêtes entre un poste client et un serveur. Il agit comme un

tampon en améliorant les performances, la sécurité et la confidentialité des connexions réseau.

“__”

1.3 - Accès `sudo` pour user

- *Question(s) 3. sudo*

Comment peut-on savoir à quels groupes appartient l'utilisateur user ?

Pour savoir à quel groupe appartient l'utilisateur user, on peut utiliser la commande `sudo groups user`.

Quel est la version du noyau Linux utilisée par votre VM ?
N'oubliez pas, comme pour toutes les questions, de justifier votre réponse.

La version de notre noyau Linux utilisée est **Debian 11.6.0-amd64** comme indiqué dans les consignes lors de la mise en place de la VM. On peut connaître la version de notre noyau (aussi appelé Kernel) en utilisant la commande `uname -v` dans un terminal.

à quoi servent les suppléments invités ? Donner 2 principales raisons de les installer.

Les suppléments invités permettent d'améliorer l'expérience utilisateur et les performances de la machine virtuelle en lui fournissant des fonctionnalités supplémentaires et une meilleure intégration avec l'hôte. Les principales raisons de les installer sont :

1. *Meilleure résolution d'écran* : sans les suppléments invités, la machine virtuelle peut être limitée à une résolution d'écran basse, ce qui peut rendre l'utilisation de certains logiciels difficile. Les suppléments invités permettent de définir une résolution d'écran plus élevée, ce qui améliore l'affichage.
2. *Partage de fichiers et copier-coller entre la machine hôte et la machine virtuelle* : les suppléments invités permettent de partager des fichiers entre la machine hôte et la machine virtuelle, ce qui facilite le transfert de données. Mais il est également possible de copier-coller du texte et des fichiers entre la machine hôte et la machine virtuelle, ce qui facilite le travail entre les deux environnements.

à quoi sert la commande `mount` (dans notre cas de figure et dans le cas général) ?

La commande `mount` permet de demander au système d'exploitation de rendre un système de fichiers accessibles, à un emplacement spécifique (le point de montage).

Dans notre cas de figure, `mount` nous a servit pour installer “VirtualBox Guest Additions” qui est un ensemble de pilotes et de logiciels supplémentaires qui améliorent les performances et les fonctionnalités des systèmes d’exploitation invités dans VirtualBox.

2. Installation de Git

Dans cette section, nous verrons comment installer et configurer l’outil **Git**.

2.1 - Configuration globale de git

Installation de git :

```
sudo apt install git-gui
```

- **Questions 1. Préliminaire**

Qu’est-ce que le logiciel *git-gui* ? Comment se lance-t-il ?

Le logiciel *git-gui* est l’interface graphique de **Git** qui est en ligne de commande. Git-gui fournit une interface utilisateur graphique pour effectuer des opérations Git courantes telles que la création de commits, la fusion de branches et la gestion des conflits. mais toutes les possibilités offertes par Git ne sont pas disponibles dans Git Gui.

Pour le lancer il suffit simplement de faire la commande `git gui` dans un terminal.

- *Source* : <https://www.codeur-pro.fr/>

Qu’est-ce que le logiciel *gitk* ? Comment se lance-t-il ?

Gitk est un logiciel graphique de visualisation d’historique Git. Il permet de visualiser l’historique des commits et des branches, ainsi que les différences entre les versions d’un fichier.

Pour le lancer il suffit simplement de faire la commande `gitk` dans un terminal.

- *Source* : <https://www.atlassian.com/>

Quelle sera la ligne de commande *git* pour utiliser par défaut le proxy de l’université sur tous vos projets git ?

```
export http_proxy=http://cache.univ-lille.fr:3128
export https_proxy=http://cache.univ-lille.fr:3128
```

3. Installation de Gitea

- **Question(s) 2. a propos de Gitea**

Qu’est ce que Gitea ?

Gitea est une plateforme d'hébergement de code open source et legere pour Git, permettant de gerer des dépôts de code source et des collaborations de maniere simplifiée.

a quels logiciels bien connus dans ce domaine peut-on le comparer
(en citer au moins 2) ?

On peut le comparer a **GitLab** et **GitHub**.

3.1 - Installation du binaire

Telechargement du binaire :

```
wget -O gitea https://dl.gitea.com/gitea/1.18.5/gitea-1.18.5-linux-amd64
chmod +x gitea
```

Verification de la signature avec la cle GPG (pour eviter toutes modifications du binaire) :

```
gpg --keyserver keys.openpgp.org --recv 7C9E68152594688862D62AF62D9AE806EC1592E2
gpg --verify gitea-1.18.5-linux-amd64.asc gitea-1.18.5-linux-amd64
```

“__”

3.2 - Preparation de l'environnement Git

Verification de la version de Git (doit etre inférieure ou égale à 2.0 pour fonctionner) : `git --version`

Creation de l'utilisateur pour demarrer Git :

```
adduser \
  --system \
  --shell /bin/bash \
  --gecos 'Git Version Control' \
  --group \
  --disabled-password \
  --home /home/git \
  git
```

Creation de la structure du repertoire necessaire :

```
mkdir -p /var/lib/gitea/{custom,data,log}
chown -R git:git /var/lib/gitea/
chmod -R 750 /var/lib/gitea/
mkdir /etc/gitea
chown root:git /etc/gitea
chmod 770 /etc/gitea
```

Configuration du repertoire :

```
export GITEA_WORK_DIR=/var/lib/gitea/
```

```
#Copie du binaire Gitea dans un emplacement global
```

```
cp gitea /usr/local/bin/gitea
```

```
“__”
```

3.3 - Lancement de Gitea

Demarrage automatique de Gitea au lancement de la machine :

- Créer le fichier /etc/systemd/system/gitea.service et enregistrer le contenu suivant à l'intérieur :

```
[Unit]
Description=Gitea (Git with a cup of tea)
After=syslog.target
After=network.target
###
# Don't forget to add the database service dependencies
###
#
#Wants=mysql.service
#After=mysql.service
#
#Wants=mariadb.service
#After=mariadb.service
#
#Wants=postgresql.service
#After=postgresql.service
#
#Wants=memcached.service
#After=memcached.service
#
#Wants=redis.service
#After=redis.service
#
###
# If using socket activation for main http/s
###
#
#After=gitea.main.socket
#Requires=gitea.main.socket
#
###
# (You can also provide gitea an http fallback and/or ssh socket too)
#
```

```

# An example of /etc/systemd/system/gitea.main.socket
###
##
## [Unit]
## Description=Gitea Web Socket
## PartOf=gitea.service
##
## [Socket]
## Service=gitea.service
## ListenStream=<some_port>
## NoDelay=true
##
## [Install]
## WantedBy=sockets.target
##
###

[Service]
# Uncomment the next line if you have repos with lots of files and get a HTTP 500 error
# LimitNOFILE=524288:524288
RestartSec=2s
Type=simple
User=git
Group=git
WorkingDirectory=/var/lib/gitea/
# If using Unix socket: tells systemd to create the /run/gitea folder, which will contain
# (manually creating /run/gitea doesn't work, because it would not persist across reboots)
#RuntimeDirectory=gitea
ExecStart=/usr/local/bin/gitea web --config /etc/gitea/app.ini
Restart=always
Environment=USER=git HOME=/home/git GITEA_WORK_DIR=/var/lib/gitea
# If you install Git to directory prefix other than default PATH (which happens
# for example if you install other versions of Git side-to-side with
# distribution version), uncomment below line and add that prefix to PATH
# Don't forget to place git-lfs binary on the PATH below if you want to enable
# Git LFS support
#Environment=PATH=/path/to/git/bin:/bin:/sbin:/usr/bin:/usr/sbin
# If you want to bind Gitea to a port below 1024, uncomment
# the two values below, or use socket activation to pass Gitea its ports as above
###
#CapabilityBoundingSet=CAP_NET_BIND_SERVICE
#AmbientCapabilities=CAP_NET_BIND_SERVICE
###
# In some cases, when using CapabilityBoundingSet and AmbientCapabilities option, you may
# set the following value to false to allow capabilities to be applied on gitea processes
# value if set to true sandboxes gitea service and prevent any processes from running u

```

```
# in the host user namespace.
###
#PrivateUsers=false
###
```

```
[Install]
WantedBy=multi-user.target
```

Effectuer ces successions de commande dans le terminal, afin d'activer et demarrer Gitea a chaque demarrage de la machine :

```
# Enable and start Gitea at boot
sudo systemctl enable gitea
sudo systemctl start gitea
```

```
# If you have systemd version 220 or later, you can enable and immediately start Gitea at on
sudo systemctl enable gitea --now
```

“—” -> **ETAPE OPTIONNELLE** Realiser cette etape si le demarrage de Gitea echoue.

Installation de *supervisor* :

```
# Install supervisor by running below command in terminal
sudo apt install supervisor
```

```
# Create a log dir for the supervisor logs
# /\ En supposant que Gitea est installe dans "/home/git/gitea/"
mkdir /home/git/gitea/log/supervisor
```

Creation du fichier configuration de *supervisor* :

- Creer le fichier /etc/supervisor/supervisord.conf et enregistrer le contenu suivant a l'interieur :

Modifiez les parametres d'utilisateur (*git*) et d'accueil (*/home/git*) pour qu'ils correspondent a l'environnement de deploiement. Modifiez le *PORT* ou supprimez l'indicateur *-p* si le port par default est utilise.

```
[program:gitea]
directory=/home/git/go/src/github.com/go-gitea/gitea/
command=/home/git/go/src/github.com/go-gitea/gitea/gitea web
autostart=true
autorestart=true
startsecs=10
stdout_logfile=/var/log/gitea/stdout.log
stdout_logfile_maxbytes=1MB
stdout_logfile_backups=10
stdout_capture_maxbytes=1MB
stderr_logfile=/var/log/gitea/stderr.log
```



```
stderr_logfile_maxbytes=1MB
stderr_logfile_backups=10
stderr_capture_maxbytes=1MB
user = git
environment = HOME="/home/git", USER="git"
```

Activation et demarrage automatique de *supervisor* :

```
sudo systemctl enable supervisor
sudo systemctl start supervisor
```

Si vous avez systemd version 220+, vous pouvez activer et demarrer immédiatement le superviseur avec :

```
sudo systemctl enable supervisor --now
```

“—” Demarrage via le terminal :

```
GITEA_WORK_DIR=/var/lib/gitea/ /usr/local/bin/gitea web -c /etc/gitea/app.ini
```

Redemarrage de Gitea avec **systemd** (recommande) : `systemctl restart gitea`

Si le demarrage echoue, realiser l'etape optionnelle precedant celle-ci.

“—”

3.4 - Parametrage de Gitea

Vérifier que le service est bien démarré avec : `systemctl status gitea.service`

Utiliser le navigateur physique de la machine et se rendre sur l'URL suivant : `http://localhost:3000`

Arrivée sur l'interface de Gitea

Début du paramétrage avec les informations nécessaires :

- La base de données sera SQLite3 ;
- Le compte administrateur web sera :
 - Nom : gitea
 - Password : gitea
 - Email : git@localhost

→ A la fin de l'installation, ne pas oublier de protéger `/etc/gitea` et `/etc/gitea/app.ini` :

```
chmod 750 /etc/gitea
chmod 640 /etc/gitea/app.ini
```

- *Question(s) 3. Mise à jour*

Installation - Gitea: Git with a cup of tea - Mozilla Firefox

Installation - Gitea: Git w | x

localhost:3000

Configuration initiale

Si vous exécutez Gitea dans Docker, veuillez lire la [documentation](#) avant de modifier les paramètres.

Paramètres de la base de données

Gitea nécessite MySQL, PostgreSQL, MSSQL, SQLite3 ou TiDB (avec le protocole MySQL).

Type de base de données *

Hôte *

Nom d'utilisateur *

Mot de passe *

Figure 1: Illustration configuration Gitea

▼ Paramètres de compte administrateur

La création d'un compte administrateur est facultative. Le premier utilisateur enregistré deviendra automatiquement un administrateur le cas échéant.

Nom d'utilisateur administrateur

Mot de passe

Confirmez le mot de passe

Adresse e-mail

Figure 2: Illustration creation utilisateur Gitea

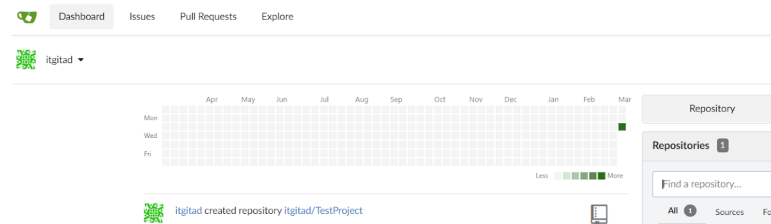
Comment faire pour la mettre a jour sans devoir tout reconfigurer ?
Essayez en mettant a jour vers la version 1.19.

Pour mettre a jour Gitea tout en conservant la configuration existante, vous pouvez suivre les etapes suivantes :

1. Sauvegardez vos donnees : **Avant toute mise a jour, il est recommande de sauvegarder vos donnees Gitea, y compris la base de donnees et les fichiers de configuration.**
2. Telechargez la nouvelle version de Gitea : Rendez-vous sur le site officiel de Gitea pour telecharger la derniere version stable de Gitea.
3. Arretez le service Gitea : Arretez le service Gitea sudo avec la commande suivante : `sudo systemctl stop gitea`
4. Faites une copie des fichiers de configuration : Faites une copie des fichiers de configuration de Gitea (generalement situes dans le repertoire `/etc/gitea`) (et des fichiers de log s'ils existent).
5. Installez la nouvelle version de Gitea : Installez la nouvelle version de Gitea en suivant les instructions.
6. Copiez les fichiers de configuration : Copiez les fichiers de configuration de l'etape 4 dans le repertoire de configuration de la nouvelle installation. Assurez-vous de remplacer les fichiers de configuration de la nouvelle installation par ceux que vous avez sauvegardes.
7. Redemarrez le service Gitea : Redemarrez le service Gitea en utilisant la commande suivante : `sudo systemctl start gitea`
8. Verifiez que tout fonctionne correctement : Ouvrez votre navigateur et accédez a votre instance Gitea pour verifier que tout fonctionne correctement.

“__”

3.5 - Utilisation de Gitea




Creation d'un projet depuis l'interface Gitea :

- **Question(s) 4. Projets existans**

Que se passe-t-il ? Qu'elle semble en etre la cause ? Corrigez ce probleme.

Nous avons remarquer aucun probleme lors de la realisation des etapes finales.

New Repository

Owner *  itgitad

Repository Name * TestProject

Good repository names use short, memorable and unique keywords.

Visibility

☐ Make Repository Private

Description

Test Git Project with Gitea

.gitignore

Select .gitignore templates.

License

Select a license file.

README

Default

☒ Initialize Repository (Adds .gitignore, License and README)

Create Repository

Cancel

Figure 3: Illustration creation projet Gitea 2

Dashboard Issues Pull Requests Explore

itgitad / TestProject

Watch 1

Star 0

Fork 0

Code Issues Pull Requests Releases Wiki Activity

Test Git Project with Gitea

1 Commit

1 Branch

Branch: master

New Pull Request

New File Upload File

[http](http://localhost:3000/itgitad/)
[SSH](http://localhost:3000/itgitad/)

itgitad

a9d5085542

Initial commit

34 minutes ago

README.md

Initial commit

34 minutes ago

README.md

TestProject

Test Git Project with Gitea

Figure 4: Illustration creation projet Gitea 3

12

Conclusion

Au cours de cette SAe l'ensemble de notre equipe a pu beneficier d'un apprentissage theorique, au travers de diverses questions techniques, mais surtout pratiques, en s'exerçant directement, en manipulant les premiers outils et technologies, le tout en apprenant a faire face aux difficultees et a les surmonter, ce en respectant des deadlines mais aussi une methode de travail encadree, via notamment la redaction du rapport precedent ainsi que de celui-ci. Ce savoir-faire nous est par ailleurs directement utile pour la gestion de nos projets et travaux au sein de l'IUT, et ceux a venir.