

Rapport technique intermédiaire.

Sommaire :

1. Préparation de la machine virtuelle

- Préparation de la machine virtuelle. - Installation de l'OS.

2. Préparation système.

- Accès sudo pour *user*. - Installation des suppléments invités.

3. À propos de la distribution Debian.

- Documentations. - Quelques questions.

4. Installation préconfigurée.

- Récupérer et préparer les fichiers nécessaires.

5. Analyse préliminaire de git et des outils graphiques associés.

- Configuration globale de *git*. - Interfaces graphiques pour *git*. - Installons autre chose et comparons. - Essais à faire. - Comparaisons.

6. Redirection de port.

- Redirection de port.

7. Installation de *Gitea*.

- Quelques questions. - Installation “from binary” de *Gitea* - Mise à jour du binaire du service *Gitea*

Préparation de la machine Virtuelle :

Préparation de la machine virtuelle :

1. Que signifie “64-bit” dans “Debian 64-bit” ? > “64-bits” signifie que Debian est conçu pour fonctionner sur des processeurs capables de traiter des instructions en 64 bits.

2. Quelle est la configuration réseau utilisée par défaut ?

La configuration réseau par défaut est la configuration NAT (Network Address Translation).

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

Il s'agit du fichier : sae203.vbox-prev

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

Il doit y avoir une balise qui sert dans ce but, mais ce n'est pas recommandé.

Installation de l'OS :

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

Oui, il est important de savoir ce qu'est un fichier ISO bootable pour l'installation de Debian. C'est une image disque complète d'un CD ou DVD amorçable qui peut être gravée sur un support physique (comme une clé USB ou un DVD) et utilisée pour démarrer un ordinateur et installer un système d'exploitation.

2. Qu'est-ce que MATE ? GNOME ?

Oui, il est important de connaître les différences entre les environnements de bureau MATE et GNOME pour choisir celui qui convient le mieux à vos besoins. - MATE: MATE est un environnement de bureau stable, léger et personnalisable, avec une interface utilisateur traditionnelle. - GNOME: GNOME est un environnement de bureau moderne et convivial, avec une interface utilisateur intuitive et des fonctionnalités avancées.

3. Qu'est-ce qu'un serveur web ?

Oui, il est important de comprendre le fonctionnement d'un serveur web pour configurer et administrer votre propre serveur. Un serveur web est un logiciel qui distribue des fichiers web aux clients qui en font la demande via un navigateur web.

4. Qu'est-ce qu'un serveur ssh ?

Oui, il est important de connaître les avantages d'utiliser un serveur ssh pour sécuriser vos connexions à distance. Un serveur ssh permet de se connecter à un ordinateur distant de manière sécurisée en utilisant le protocole SSH.

5. Qu'est-ce qu'un serveur *mandataire* ?

Oui, il est important de comprendre les avantages et les inconvénients d'utiliser un serveur mandataire pour améliorer la sécurité et la confidentialité de votre navigation web. Un serveur mandataire agit comme un pont entre votre ordinateur et Internet et peut être utilisé pour filtrer les contenus web, masquer votre adresse IP et améliorer la performance de votre navigation.

Préparation du système :

Accès sudo pour *user*:

1. Ajout du groupe sudo au groupe principal :
adduser user sudo ajoute user au groupe sudo
2. Comment peut-on savoir à quels groupe appartient l'utilisateur *user* ?
usermod -aG sudo pareil mais à partir du compte root
4. **Comment peut-on savoir à quels groupes appartient l'utilisateur *user*?**
groups user

Installation des suppléments invités :

1. Quelle est la version du noyau de Linux utilisé par votre VM ? > Commande utilisée : `uname -r` > Ici c'est la version 6.1.0-17-amd64
2. À quoi servent les suppléments invités ? Donner 2 principales raisons de les installer.
 1. Améliorer les performances: Les suppléments invités installent des pilotes de périphériques et d'autres logiciels qui peuvent améliorer les performances de votre machine virtuelle, notamment la vitesse de la souris, le partage du presse-papiers et la gestion de la résolution d'écran.
 2. Accéder à des fonctionnalités supplémentaires: Les suppléments invités peuvent vous donner accès à des fonctionnalités supplémentaires, telles que la possibilité de partager des dossiers entre votre machine virtuelle et votre ordinateur hôte, ou de synchroniser l'heure de la machine virtuelle avec l'heure de votre ordinateur hôte.
3. À quoi sert la commande *mount* ? (Dans notre cas de figure et dans le cas général). >1. Dans notre cas de figure : La commande `mount` est utilisée pour monter le CD des suppléments invités sur votre machine virtuelle. Cela permet d'installer les suppléments invités.
 2. Cas général : La commande `mount` est utilisée pour associer un périphérique de stockage à un point de montage dans le système de fichiers. Cela permet d'accéder aux fichiers et dossiers du périphérique de stockage.

Exemples :

- Monter un CD-ROM :

```
sudo mount /dev/cdrom /mnt/cdrom
```

- Monter une partition USB :

```
sudo mount /dev/sdb1 /mnt/usb
```

À propos de la distribution Debian

Documentation :

- debian.org/doc/ : Le site officiel de Debian propose une documentation complète et traduite dans de nombreuses langues, y compris le français.
- debian-fr.org : Portail francophone avec des guides, des tutoriels et des forums pour vous aider à utiliser Debian.
- forum-debian.fr : Forum francophone où vous pouvez poser vos questions à propos de Debian et obtenir des réponses d'autres utilisateurs.
- debian-facile.org : Site avec de la documentation en français sous forme de wiki et un forum.

Quelques questions :

1. Qu'est-ce que le Projet Debian ? D'où vient le nom Debian ?

Le Projet Debian est une communauté mondiale de développeurs bénévoles qui s'occupent de la distribution GNU/Linux Debian. Le nom "Debian" est un mot-valise composé des prénoms de Ian Murdock, le fondateur du projet, et de sa femme Debra.

2. Quelle sont les durées des 3 prises en charge de ces versions ?

- Durée minimale : 1 an
- Durée en support long terme (LTS) : 5 ans
- Durée en support long terme étendue (ELTS) : 5 années supplémentaires après la fin du support LTS (soit 10 ans)

3. Pendant combien de temps les mises à jour de sécurité seront-elles fournies ?

Les mises à jour de sécurité sont fournies pendant la durée de prise en charge de la version. Pour les versions LTS et ELTS, les mises à jour de sécurité sont disponibles pendant 10 ans après la date de publication de la version.

4. Combien de version au minimum sont activement maintenues par Debian ?

Au minimum 3 versions sont activement maintenues : - La version stable actuelle (Bullseye) - La version “oldstable” (Buster) - La version “oldoldstable” (Stretch)

5. D'où viennent les noms de code donnés aux distributions ?

Les noms de code des distributions Debian sont tirés des personnages de la série de livres et films Toy Story.

6. Combien et quelles d'architectures sont prises en charge par la version Bullseye ?

Bullseye prend en charge 11 architectures, dont amd64, i386, arm64, armel, powerpc64el, s390x et sparc64.

7. Première version avec un nom de code :

- Premier nom de code : > Buzz
- Date d'annonce : >16 août 1996
- Numéro de version de cette distribution : > 1.1

8. Dernier nom de code attribué :

- Dernier nom de code : > Bookworm
- Date d'annonce : > 30 juillet 2022
- Numéro de version : > 12

Installation préconfigurée

Récupérer et préparer les fichiers nécessaires :

1. Ajout du droit d'utilisation de *sudo* par l'utilisateur principal

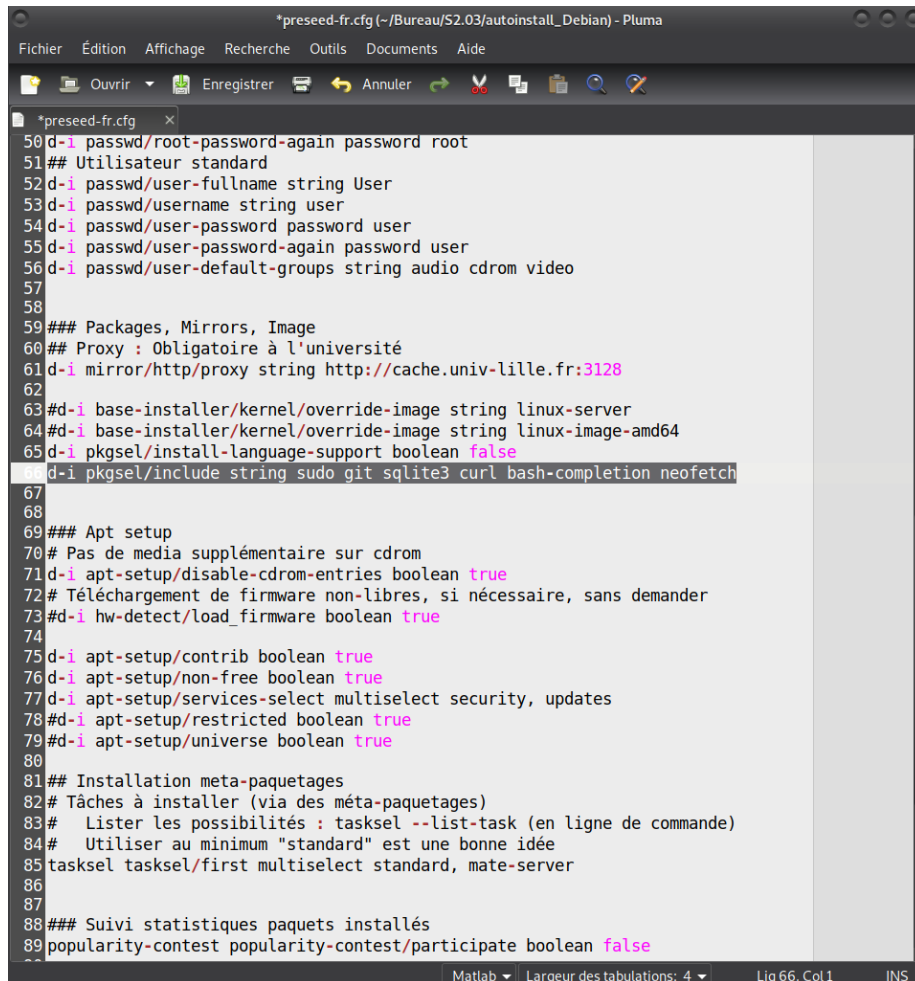
#d-i passwd/user-default-groups string audio cdrom video sudo (juste ajouter sudo a la liste des groupes)

2. Installer l'environnement mate

Modification du fichier preseed-fr.cfg :

```
61 d-i mirror/http/proxy string http://cache.univ-tlse.fr:3128
62
63 #d-i base-installer/kernel/override-image string linux-server
64 #d-i base-installer/kernel/override-image string linux-image-amd64
65 d-i pkgsel/install-language-support boolean false
66 d-i pkgsel/include string sudo git sqlite3 curl bash-completion neofetch
67
68
69 ### Apt setup
70 # Pas de media supplémentaire sur cdrom
71 d-i apt-setup/disable-cdrom-entries boolean true
72 # Téléchargement de firmware non-libres, si nécessaire, sans demander
73 #d-i hw-detect/load_firmware boolean true
74
75 d-i apt-setup/contrib boolean true
76 d-i apt-setup/non-free boolean true
77 d-i apt-setup/services-select multiselect security, updates
78 #d-i apt-setup/restricted boolean true
79 #d-i apt-setup/universe boolean true
80
81 ## Installation meta-paquetages
82 # Tâches à installer (via des méta-paquetages)
83 # Lister les possibilités : taskel --list-task (en ligne de commande)
84 # Utiliser au minimum "standard" est une bonne idée
85 taskel taskel/first multiselect standard, mate-desktop
86
87
88 ### Suivi statistiques paquets installés
89 popularity-contest popularity-contest/participate boolean false
90
91
92 ### Grub
93 d-i grub-installer/grub2_instead_of_grub_legacy boolean true
94 d-i grub-installer/only_debian boolean true
95 d-i grub-installer/timeout string 2
96 # To install to the first device (assuming it is not a USB stick):
97 d-i grub-installer/bootdev string default
98
99 d-i finish-install/reboot_in_progress note
100
```

3. Ajout des package sudo, git, sqlite3, curl, bash-completion et neofetch au fichier de configuration automatique (preseed-fr) > Modification du fichier preseed-fr.cfg



```
*preseed-fr.cfg
50 d-i passwd/root-password-alternative password root
51 ## Utilisateur standard
52 d-i passwd/user-fullname string User
53 d-i passwd/username string user
54 d-i passwd/user-password password user
55 d-i passwd/user-password-alternative password user
56 d-i passwd/user-default-groups string audio cdrom video
57
58
59 ### Packages, Mirrors, Image
60 ## Proxy : Obligatoire à l'université
61 d-i mirror/http/proxy string http://cache.univ-lille.fr:3128
62
63 #d-i base-installer/kernel/override-image string linux-server
64 #d-i base-installer/kernel/override-image string linux-image-amd64
65 d-i pkgsel/install-language-support boolean false
66 d-i pkgsel/include string sudo git sqlite3 curl bash-completion neofetch
67
68
69 ### Apt setup
70 # Pas de media supplémentaire sur cdrom
71 d-i apt-setup/disable-cdrom-entries boolean true
72 # Téléchargement de firmware non-libres, si nécessaire, sans demander
73 #d-i hw-detect/load_firmware boolean true
74
75 d-i apt-setup/contrib boolean true
76 d-i apt-setup/non-free boolean true
77 d-i apt-setup/services-select multiselect security, updates
78 #d-i apt-setup/restricted boolean true
79 #d-i apt-setup/universe boolean true
80
81 ## Installation meta-paquetages
82 # Tâches à installer (via des méta-paquetages)
83 # Lister les possibilités : tasksel --list-task (en ligne de commande)
84 # Utiliser au minimum "standard" est une bonne idée
85 tasksel tasksel/first multiselect standard, mate-server
86
87
88 ### Suivi statistiques paquets installés
89 popularity-contest popularity-contest/participate boolean false
90
```

Analyse préliminaire de git et des outils graphiques associés :

Configuration globale de *git* :

1. Explication des lignes suivantes :

- `git config --global user.name "Prénom Nom"`
- `git config --global user.email "votre@email"`
- `git config --global init.defaultBranch "master"`

Code permettant de configurer notre nom d'utilisateur, notre adresse email et la branche par défaut pour Git pour que nos commits soient correctement identifiés :

Les interfaces graphiques pour *git* :

1. Que sont les logiciels gitk et git-gui ?

Gitk et git-gui sont deux outils graphiques puissants pour visualiser et gérer l'historique de votre dépôt Git.

2. Comment les installe-t-on ?

```
sudo apt-get install gitk
sudo apt-get install git-gui
```

3. Comment les lance-t-on ?

```
gitk
git-gui
```

4. Que permet *gitk* ?

- Explorer l'arborescence des commits de votre projet.
- Visualiser les modifications apportées aux fichiers à chaque commit.
- Comparer différentes versions d'un même fichier.
- Filtrer les commits par auteur, date, message, etc.
- Accéder aux informations détaillées de chaque commit.

5. Que permet *git-gui* ?

- Créer des commits en sélectionnant les fichiers modifiés.
- Ajouter un message de commit clair et précis.
- Annuler les modifications apportées aux fichiers.
- Créer et gérer des branches.
- Fusionner des branches entre elles.

6. Quels sont les avantages des outils graphiques utilisés ?

- Facilité d'utilisation : Ils offrent une interface intuitive et conviviale, accessible aux utilisateurs débutants.
- Visualisation claire : Ils permettent de visualiser l'historique du projet de manière graphique, ce qui facilite la compréhension des modifications apportées.
- Gain de temps : Ils automatisent certaines tâches répétitives, comme la création de commits ou la fusion de branches.

7. Quels sont leurs inconvénients ?

- Moins puissants que les commandes en ligne de commande : Ils ne proposent pas toutes les fonctionnalités disponibles avec Git.
- Dépendance d'une interface graphique : Ils ne peuvent pas être utilisés sur des serveurs sans interface graphique.

Installons autre chose et comparons :

1. Sur quel logiciel s'est porté votre choix ?

Notre choix s'est porté sur le logiciel Magit.

2. Pourquoi ce choix ?

Il est l'une des interfaces gratuites les plus utilisées, de ce fait la communauté est active ce qui nous facilitera si besoin. De plus, l'interface est puissante, flexible et personnalisable : Magit permet de gérer facilement les remotes, en les clonant, les synchronisant et en y "poussant" les modifications. On peut également y voir l'historique complet de nos dépôts avec des options et filtres de recherche.

3. Comment l'avez vous installé ?

```
sudo apt update
```

```
sudo apt-get install magit
```

4. Comment l'avez vous lancé ?

```
emacs
```

une fois dans l'application, il faut déclencher l'apparition du terminal avec la combinaison de touches : Alt + x.

```
magit
```

Essais à faire :

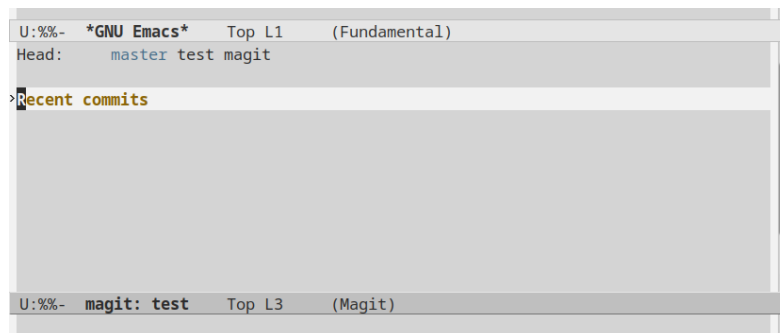
Prérequis : dépôt git local

Essai avec des documents personnels : emacs

1. Comment avez vous fait pour accéder directement à un répertoire git en ouvrant Magit ?

- Étape 1: Après la création d'un répertoire « test » contenant les fichier test.txt et textdiff.txt, on lance emacs avec la commande « emacs ».
- Étape 2: Activer Magit (Avec Alt + x) :

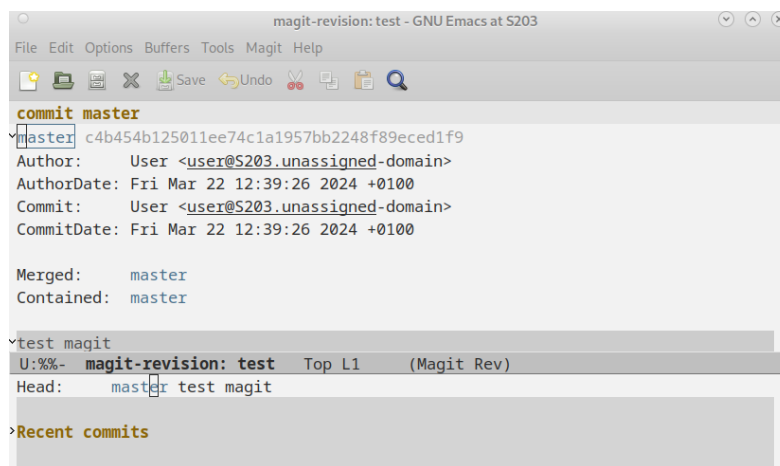
```
M-x magit
```



Accès au répertoire “test” dont le message de commit était test magit.

2. Comment avez vous fait pour accéder à l’historique du répertoire git ?

Étape 1: En s’aidant des flèches droites et gauches, déplacer le curseur sur la branche qu’on veut explorer et appuyer sur entrée :



Étape 2: A l’aide des flèches haut et bas, naviguer dans l’historique des versions et créations de fichier :



The screenshot shows the Emacs editor with the Magit package installed. The main window displays a commit diff for a commit on the 'master' branch. The diff shows two files changed: 'testdiff.txt' and 'text.txt'. 'testdiff.txt' has 2 insertions, and 'text.txt' has 2 insertions. The diff content is highlighted in green. The status bar at the bottom shows 'U: % - magit-revision: test Bot 1.22 (Magit Rev)' and 'Head: master test magit'. The 'Recent commits' panel at the bottom right shows the current commit as 'U: % - magit: test Top 1.1 (Magit)'.

3. Comment peut on comparer deux versions d'un fichier ?

Étape 1: Modifier le fichier

Étape 2: Ouvrir le fichier dans Emacs

`emacs <nom_fichier>`

Étape 3: Activer Magit (Avec Alt + x) :

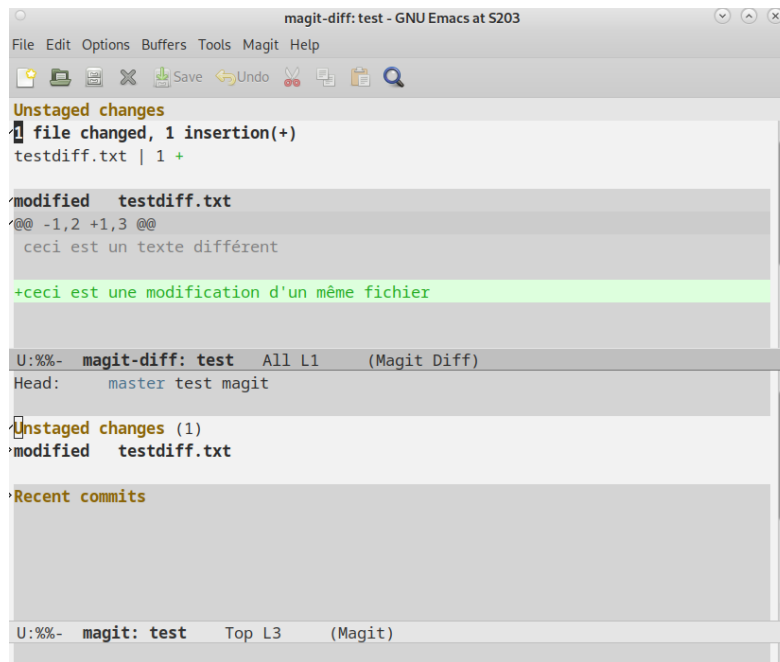
`M-x magit`

Étape 4: Afficher les différences (Avec Alt + x) :

`M-x magit-diff`

Étape 5: Sélectionner les versions à comparer : - Utilisation des touches fléchées - Commande : `M-x magit-log`

Avec ces deux méthodes, nous pouvons afficher l'historique complet du fichier :



4. Comment vous êtes vous pris pour créer et supprimer des branches ?

Étape 1: Créer une branche (Avec Alt + x) :

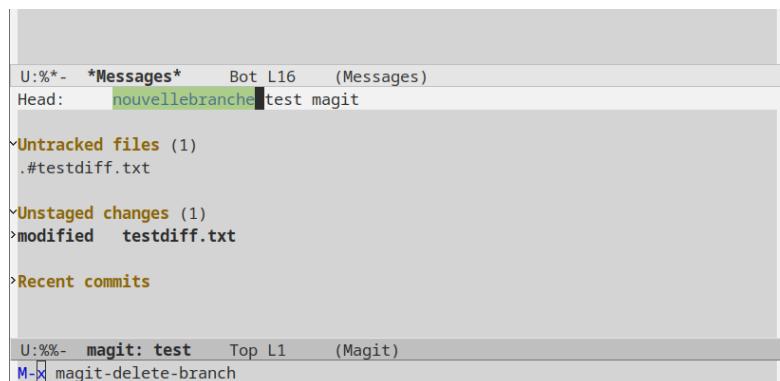
M-x magit-branch-create

On l'appelle 'nouvelle branche'.

Étape 2: Basculer vers une branche (Avec Alt + x) :

M-x magit-checkout

nouvellebranche



Comparaison :

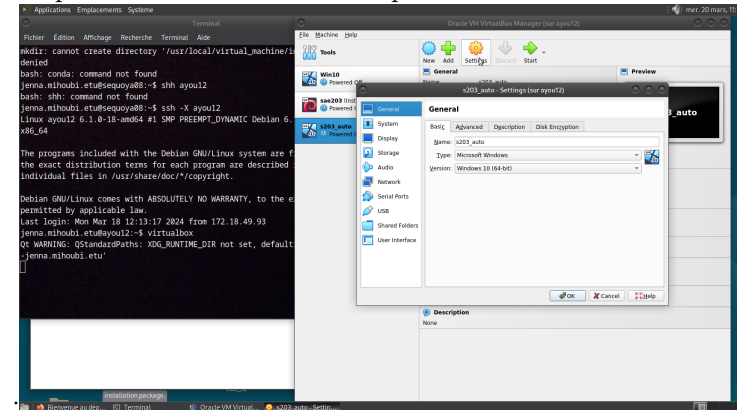
Fonctionnalité	Gitk	Git-gui	Magit
Avantages	Possède une interface graphique simple et intuitive. Permet de visualiser l'historique du projet. Offre des options de filtrage et de recherche.	Possède une interface graphique simple et intuitive. Permet de créer et de modifier des commits. Facilite la gestion des branches. Offre des options de fusion et de rebase.	Possède une intégration native avec Emacs. Offre un large éventail de fonctionnalités pour visualiser et gérer l'historique du projet. Permet d'effectuer des actions complexes sur les commits et les branches. Offre une interface personnalisable.
Inconvénients	Possède des fonctionnalités limitées. Ne permet pas de créer ou de modifier des commits. N'offre pas d'options de collaboration.	Possède des fonctionnalités limitées comparées à Magit. Ne s'intègre pas avec d'autres outils de développement. Possède une interface moins personnalisable que Magit.	Possède une courbe d'apprentissage plus raide. Nécessite une configuration d'Emacs. Ne fonctionne pas sur tous les systèmes d'exploitation.
Collaboration	Non	Non	Oui
Intégration avec d'autres outils	Non	Non	Oui
Coût	Gratuit	Gratuit	Gratuit
Licence	Open-source	Open-source	Open-source

Redirection de port :

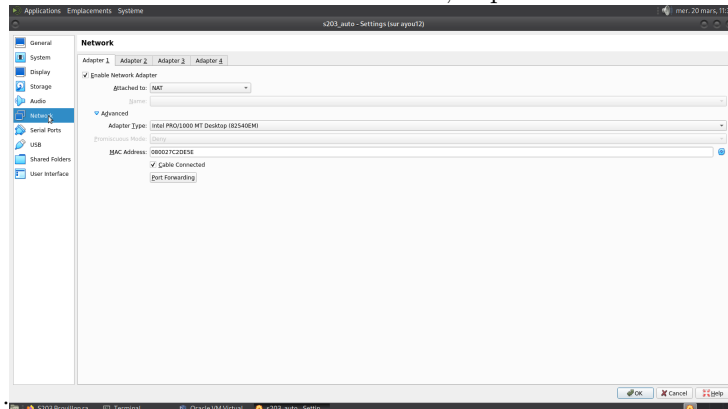
Redirection de port :

Montrez comment vous êtes parvenus à rediriger les ports :

- Étape 1: Se rendre dans les paramètres de la virtualbox

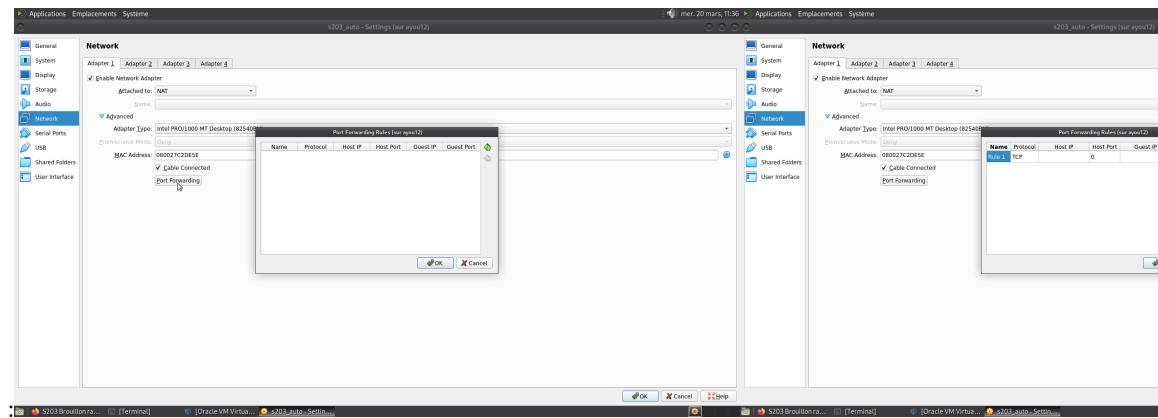


- Étape 2: Se rendre dans la section Network, déplier le menu



avancé :

- Étape 3: Sélectionner port forwarding et remplir les champs Host Port et Guest Port par 3000 puis valider



Le port 3000 est choisi car Gitea utilise par défaut le port 3000 pour se lancer. Une fois Gitea installé, nous n'aurons plus besoin de systématiquement lancer la machine virtuelle. Nous pourrions y accéder, par exemple, grâce à l'url : <http://localhost:3000>.

Il ne nous reste donc plus maintenant qu'à installer le service Gitea sur notre machine virtuelle.

Installation de *Gitea* :

Quelques questions :

1. Qu'est-ce que Gitea ?

Gitea est un logiciel de gestion libre de dépôt Git (versions de code source) auto-hébergé, il permet d'héberger de gérer les dépôts git en plus de présenter des fonctionnalités telles que le suivi des problèmes, la gestion des versions et l'intégration continue.

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

Nous pouvons comparer Gitea à GitHub ou encore GitLab. GitHub: Le service de gestion de versions de code source le plus populaire. GitLab : Un autre service de gestion de versions de code source open source (tout comme gitea).

[Source sur la définition de Gitea][Gitea - Bienvenue sur le site consacré à l'utilisation de Linux ! - Wiki (univers-linux.fr)](<https://www.univers-linux.fr/wiki/gitea#:~:text=Gitea%20est%20un%20logiciel%20libre%20de%20code%20source>)

[Comparaison entre Gitea et GitHub][Gitea vs GitHub | What are the differences? (stackshare.io)](<https://stackshare.io/stackups/gitea-vs-github>)

[Comparaison entre Gitea et GitLab][Gitea VS Gitlab (hostedgitea.com)](<https://hostedgitea.com/article/g-vs-gitlab>)

Installation “from binary” de *Gitea* :

- Étape 1: Télécharger le fichier avec wget :

Commandes pour télécharger Gitea 1.21.8 pour 64-bit Linux :

```
wget -O gitea https://dl.gitea.com/gitea/1.21.8/gitea-1.21.8-linux-amd64
```

```
chmod +x gitea
```

<https://dl.gitea.com/gitea/1.21.7/gitea-1.21.7-linux-amd64.xz>

Téléchargement réussi :



```
Terminal
Fichier Édition Affichage Recherche Terminal Aide
gpg: verify signatures failed: Aucun fichier ou dossier de ce type
root@S203:~# wget -O https://dl.gitea.com/1.21.8/gitea-1.21.8-linux-amd64 .sig
https://dl.gitea.com/1.21.8/gitea-1.21.8-linux-amd64: Aucun fichier ou dossier
de ce type
root@S203:~# wget -O gitea https://dl.gitea.com/gitea/1.21.8/gitea-1.21.8-linu
x-amd64 .sig
--2024-03-18 12:47:21-- https://dl.gitea.com/gitea/1.21.8/gitea-1.21.8-linux-
amd64
Résolution de dl.gitea.com (dl.gitea.com)... 52.84.174.33, 52.84.174.60, 52.84.1
74.6, ...
Connexion à dl.gitea.com (dl.gitea.com)[52.84.174.33]:443... connecté.
requête HTTP transmise, en attente de la réponse... 200 OK
Taille : 138110128 (132M) [binary/octet-stream]
Sauvegarde en : « gitea »

gitea          100%[=====>] 131,71M  91,7MB/s   ds 1,4s

2024-03-18 12:47:23 (91,7 MB/s) - « gitea » sauvegardé [138110128/138110128]

--2024-03-18 12:47:23-- http://.sig/
Résolution de .sig (.sig)... échec : Nom ou service inconnu.
wget : impossible de résoudre l'adresse de l'hôte « .sig »
Terminé - 2024-03-18 12:47:23 -
Temps total effectif : 1,5s
Téléchargés : 1 fichiers, 132M en 1,4s (91,7 MB/s)
root@S203:~#
```

- Étape 2: Vérifier la signature GPG (facultatif) :

Gitea signe ses binaires, pour vérifier la signature nous pouvons télécharger le fichier de signature (finissant par .asc) et utiliser l'outil en ligne de commande gpg.

Pour ce faire, nous pouvons installer gpg :

```
sudo apt install gpg
```


Puis utiliser la commande :

```
gpg --verify fichier.sig
```

Si la phrase : Good signature from “Teabot <teabot@gitea.io>” s’affiche alors la signature est bonne.

[*Nous n’avons pas réussi cette étape car échec lors de l’ouverture du fichier : .asc*]

- Étape 3 : Vérifier la version :

Grâce à la commande : `bash git --version`

Il faut que la version soit la 2.0 ou supérieure.

```
root@S203:~# git --version
git version 2.39.2
```

Nous constatons que nous sommes sur la version 2.39.2 de git sur notre machine virtuelle.

- Étape 4: Créer un utilisateur (user) :

```
sudo adduser --system --no-create-home --shell /bin/bash gitea
```

```
root@S203:~# sudo adduser --system --no-create-home --shell /bin/bash gitea
```

```
root@S203:~# grep gitea /etc/passwd
gitea:x:110:65534::/nonexistent:/bin/bash
```

- Étape 5: Créer une structure de répertoire requise :

```
mkdir -p /var/lib/gitea/{custom,data,log}
```

```
chown -R gitea /var/lib/gitea/
```

```
chmod -R 750 /var/lib/gitea/
```

```
mkdir /etc/gitea
```

```
chown gitea /etc/gitea
```

```
chown root /etc/gitea
```

```
chmod 770 /etc/gitea
```

- Étape 6: Copier le Gitea binaire dans une location globale

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

- Étape 7: Lancer Gitea manuellement :

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

[Nouvelle difficulté rencontrée : erreur inconnue.]

```
root@S203:~# GITEA_WORK_DIR=/var/lib/gitea/ /usr/local/bin/gitea web -c /etc/gitea/app.ini
2024/03/20 12:11:46 ...s/setting/setting.go:180:loadRunModeFrom() [ER] Gitea is not supposed to be run as root. Sorry. If you need to use privileged TCP ports please instead use setcap and the `cap_net_bind_service` permission
root@S203:~#
```

D'après la documentation officielle de gitea, /etc/gitea est temporairement installé avec les droits d'écriture sur l'utilisateur que nous avons créé (ici gitea et root). Il est recommandé de retirer ce droit et de ne le laisser qu'en lecture seule après l'installation :

```
chmod 750 /etc/gitea
```

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

[Ça ne fonctionne toujours pas, en effet, il ne faut pas lancer git à partir de l'interface root. Nous allons donc essayer de la lancer à partir de la session utilisateur.]



The screenshot shows a terminal window titled 'Applications Emplacements système' with a menu bar containing 'Fichier', 'Édition', 'Affichage', 'Recherche', 'Terminal', and 'Aide'. The prompt is 'user@S203: ~'. The user has entered the command 'GITEA_WORK_DIR=/var/lib/gitea/ /usr/local/bin/gitea web -c /etc/gitea/app.ini'. The output shows a timestamp '2024/03/20 12:33:38' followed by an error message: '...s/setting/setting.go:96:InitCfgProvider() [ER] Unable to init config provider from "/etc/gitea/app.ini": unable to check if "/etc/gitea/app.ini" is a file. Error: stat /etc/gitea/app.ini: permission denied'. The prompt returns to 'user@S203: ~\$'.

[Nous essayons de remettre les droits sur app.ini à l'utilisateur en question :]

```
sudo chmod 770 /etc/gitea/
```

Remettre les droits d'exécution :

```
sudo chmod +x /etc/gitea/
```

```

user@S203:~$ sudo chmod +x /etc/gitea/app.ini
chmod: impossible d'accéder à '/etc/gitea/app.ini': Aucun fichier ou dossier de ce type
user@S203:~$ sudo chmod +x /etc/gitea/
user@S203:~$ GITEA_WORK_DIR=/var/lib/gitea/ /usr/local/bin/gitea web -c /etc/gitea/app.ini
2024/03/20 12:38:10 cmd/web.go:242:runWeb() [I] Starting Gitea on PID: 226853
2024/03/20 12:38:10 cmd/web.go:111:showWebStartupMessage() [I] Gitea version: 1.21.8 built with GNU Make 4.3, go1.21.8 : bindata, sqlite, sqlite_unlock_notify
2024/03/20 12:38:10 cmd/web.go:112:showWebStartupMessage() [I] * RunMode: prod
2024/03/20 12:38:10 cmd/web.go:113:showWebStartupMessage() [I] * AppPath: /usr/local/bin/gitea
2024/03/20 12:38:10 cmd/web.go:114:showWebStartupMessage() [I] * WorkPath: /var/lib/gitea/
2024/03/20 12:38:10 cmd/web.go:115:showWebStartupMessage() [I] * CustomPath: /var/lib/gitea/custom
2024/03/20 12:38:10 cmd/web.go:116:showWebStartupMessage() [I] * ConfigFile: /etc/gitea/app.ini
2024/03/20 12:38:10 cmd/web.go:117:showWebStartupMessage() [I] Prepare to run install page
2024/03/20 12:38:10 ...ation/translation.go:68:func1() [E] Failed to list locale files: open /var/lib/gitea/custom/options/locale: permission denied
user@S203:~$

```

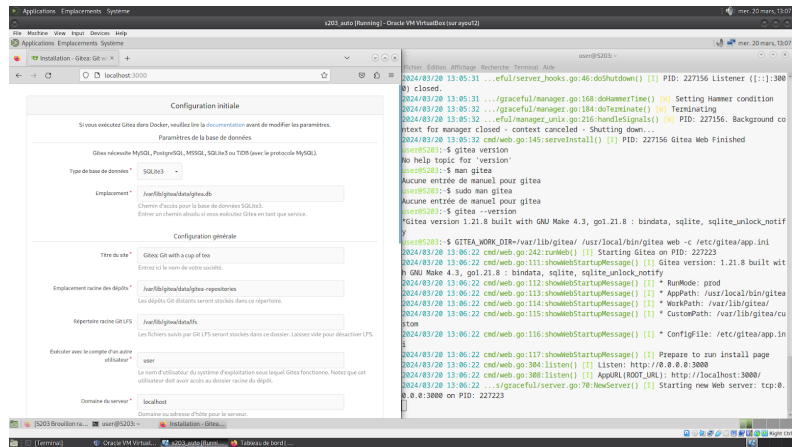
[Après plusieurs manipulation des droits, la page reste figée comme suit :]

```

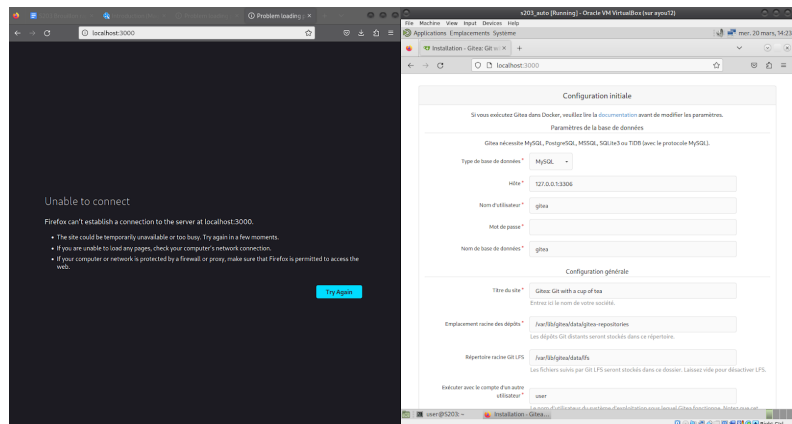
Fichier Edition Affichage Recherche Terminal Aide
user@S203:~$
2024/03/20 12:40:27 cmd/web.go:117:showWebStartupMessage() [I] Prepare to run install page
2024/03/20 12:40:27 ...ation/translation.go:68:func1() [E] Failed to list locale files: open /var/lib/gitea/custom/options/locale: permission denied
user@S203:~$ cat /var/lib/gitea/custom/options/locale
cat: /var/lib/gitea/custom/options/locale: Permission non accordée
user@S203:~$ sudo chmod +r /var/lib/gitea/custom/options/locale
chmod: impossible d'accéder à '/var/lib/gitea/custom/options/locale': Aucun fichier ou dossier de ce type
user@S203:~$ sudo chmod +w /var/lib/gitea/custom/options/locale
chmod: impossible d'accéder à '/var/lib/gitea/custom/options/locale': Aucun fichier ou dossier de ce type
user@S203:~$ sudo chmod +x /var/lib/gitea/custom/options/locale
chmod: impossible d'accéder à '/var/lib/gitea/custom/options/locale': Aucun fichier ou dossier de ce type
user@S203:~$ sudo chmod +w /var/lib/gitea/custom/
user@S203:~$ sudo chmod +r /var/lib/gitea/custom/
user@S203:~$ GITEA_WORK_DIR=/var/lib/gitea/ /usr/local/bin/gitea web -c /etc/gitea/app.ini
2024/03/20 12:42:13 cmd/web.go:242:runWeb() [I] Starting Gitea on PID: 226918
2024/03/20 12:42:13 cmd/web.go:111:showWebStartupMessage() [I] Gitea version: 1.21.8 built with GNU Make 4.3, go1.21.8 : bindata, sqlite, sqlite_unlock_notify
2024/03/20 12:42:13 cmd/web.go:112:showWebStartupMessage() [I] * RunMode: prod
2024/03/20 12:42:13 cmd/web.go:113:showWebStartupMessage() [I] * AppPath: /usr/local/bin/gitea
2024/03/20 12:42:13 cmd/web.go:114:showWebStartupMessage() [I] * WorkPath: /var/lib/gitea/
2024/03/20 12:42:13 cmd/web.go:115:showWebStartupMessage() [I] * CustomPath: /var/lib/gitea/custom
2024/03/20 12:42:13 cmd/web.go:116:showWebStartupMessage() [I] * ConfigFile: /etc/gitea/app.ini
2024/03/20 12:42:13 cmd/web.go:117:showWebStartupMessage() [I] Prepare to run install page
2024/03/20 12:42:13 cmd/web.go:304:listen() [I] Listen: http://0.0.0.0:3000
2024/03/20 12:42:13 cmd/web.go:308:listen() [I] AppURL(ROOT_URL): http://localhost:3000/
2024/03/20 12:42:13 ...s/graceful/server.go:70:NewServer() [I] Starting new Web server: tcp:0.0.0.0:3000 on PID: 226918

```

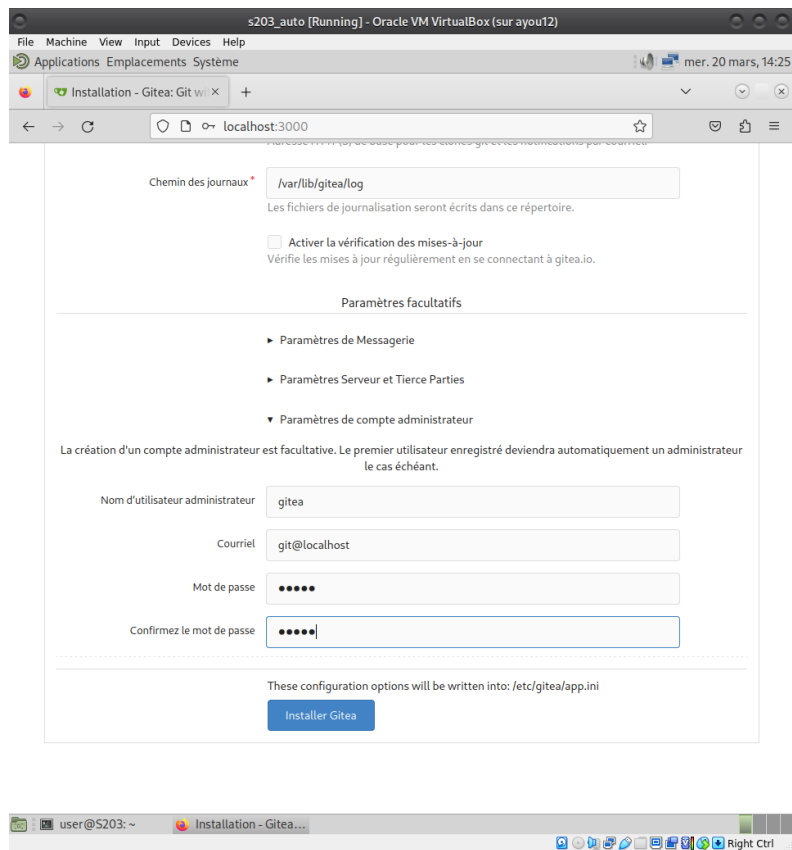
[Cela signifie que la commande a fonctionné, on peut désormais accéder via le port 3000 de la machine virtuelle :]



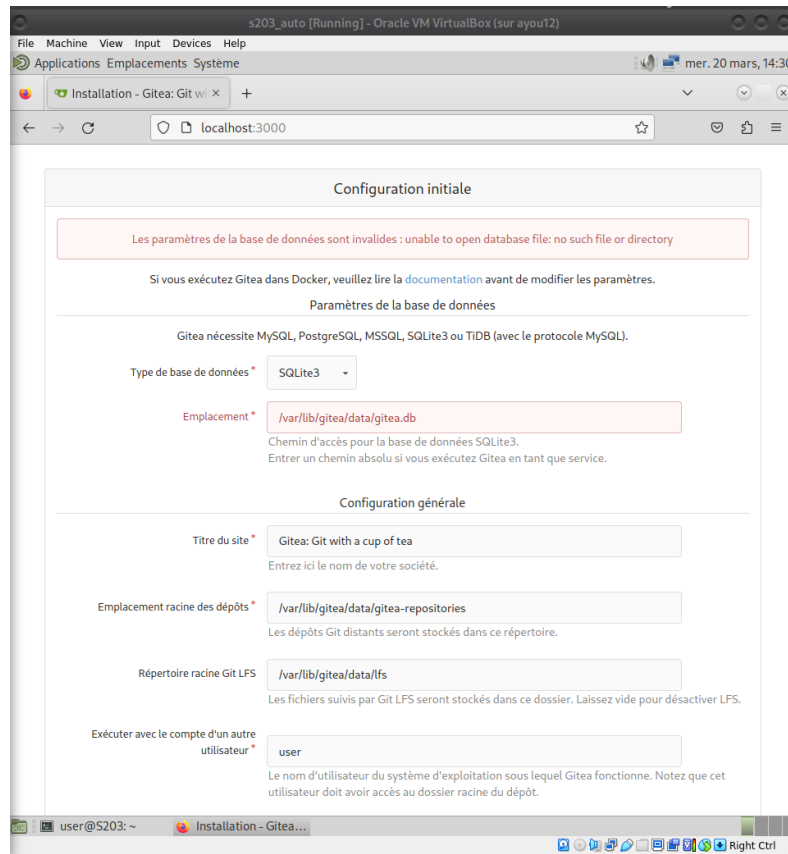
[La redirection du port 3000 de la vm vers le port 3000 de la machine hôte ne fonctionne pas :]



[Création du compte administrateur dont le mot de passe est gitea :]



[Lancement l'installation à partir de sqlite3, mais cela ne fonctionne



pas :

Source des commandes utilisées :

- Site officiel de Gitea, download from binary : <https://docs.gitea.com>

Mise à jour du binaire du service *Gitea* :

1. Quelle version du binaire avez-vous installée ? Donnez la version et la commande permettant d'obtenir cette information.

Nous avons installé la version 1.21.8. Pour le savoir, la commande est la suivante :

gitea --version.

```
user@S203:~$ gitea --version
Gitea version 1.21.8 built with GNU Make 4.3, go1.21.8 : bindata, sqlite, sqlite_unlock_notify
user@S203:~$
```

2. Comment faire pour mettre à jour le binaire de votre service sans devoir tout reconfigurer ?

Pour mettre Gitea à jour sans reconfiguration, on peut utiliser la commande : `wget` [lien vers le fichier de la version souhaitée]

```
sudo [lien vers le fichier de la version souhaitée] /usr/local/bin/gitea
```

```
sudo service gitea start
```

3. Essai en mettant à jour vers la version 1.22-dev :

```
wget https://dl.gitea.com/gitea/1.22-dev/gitea-1.22-dev-linux-amd64
```

```
sudo mv gitea-1.22-dev-linux-amd64 /usr/local/bin/gitea
```

```
sudo service gitea start
```