

Rendu technique - S2.03

Mathys MULLIER, Giorgio UTZERI



Table des matières

0.1	Semaine 6	1
0.1.1	Description du travail de la semaine	1
0.1.2	Préparation des machines virtuelles	1
0.1.3	Installation manuelle de l'OS d'une machine virtuelle	2
0.1.4	Installation automatique de l'OS d'une machine virtuelle	2
0.1.5	Questions	4
0.2	Semaine 8	6
0.2.1	Description du travail de la semaine	6
0.2.2	Configuration de git	6
0.2.3	Questions	7
0.3	Semaine 10	7
0.3.1	Description du travail de la semaine	7
0.3.2	Configuration de la machine	8
0.3.3	Création du projet	8
0.3.4	Questions	8

0.1 Semaine 6

0.1.1 Description du travail de la semaine

Durant cette première semaine, l'objectif est de procéder à la configuration de deux machines virtuelles. Les deux machines virtuelles seront identiques mais le processus de configuration sera différent pour chacune d'entre elles. Ces machines virtuelles se conforment à certaines caractéristiques. Le système d'exploitation utilisé sera Linux avec l'utilisation de sa distribution Debian (ici Debian 12) et les machines seront dotées de 2048 Mo de mémoire vive (RAM) ainsi que de 20 Go de disque dur. Sur ces machines seront alors placés deux utilisateurs et quelques logiciels de départ seront également installés. Notons également qu'il est important pour mettre en place une telle virtualisation que la machine hôte doit disposer d'au moins 4 Go de mémoire vive, 20 Go d'espace disque, et doit avoir VirtualBox installé, accompagné de son pack d'extension (VBoxGuestAddition.iso). Les deux installations seront lancées par le biais de l'interface graphique de VirtualBox. La première se fera de manière manuelle tandis que la seconde sera automatisée. Pour chacune de ces installations, nous allons répondre aux questions du sujet en lien avec ces installations.

Concernant la première installation celle-ci sera une installation rapide sur VirtualBox . Pour cela nous nous sommes aidés de la documentation debian concernant l'installation rapide.

0.1.2 Préparation des machines virtuelles

Pour chacune des machines virtuelles, bien que leur configuration soit différente, elles commencent toutefois de la même manière. La première étape de ces configurations est de créer une nouvelle machine virtuelle via l'interface graphique. Une succession de questions relatives aux caractéristiques de la machine virtuelle sera alors posée. Ces questions concernent le nom de la machine virtuelle, la mémoire vive et l'espace disque qui lui sont alloués, son emplacement sur la machine physique et d'autres questions de ce genre. Les réponses à ces questions étant pour notre installation déjà fixées (et détaillées dans la

description du travail de la semaine), il suffit d'y répondre simplement par le biais de l'interface graphique. Une fois ces informations indiquées à notre logiciel de virtualisation, il faut donner à la machine l'image ISO de notre système d'exploitation (en "l'insérant" dans le lecteur de CD-ROM virtuel de notre machine) pour que la machine puisse procéder à son installation logicielle.

Nous avons trouvé ce fichier ISO sur le site officiel de debian.

C'est précisément à cette étape que les deux configurations diffèrent. Nous nous intéresserons premièrement à l'installation manuelle du système d'exploitation puis nous nous intéresserons dans un second temps à l'automatisation de cette installation.

0.1.3 Installation manuelle de l'OS d'une machine virtuelle

L'installation commence au démarrage de la machine. Le BIOS¹, si ce dernier est bien configuré, va alors se démarrer depuis le support d'installation le plus approprié contenant une image ISO du système d'exploitation à installer. Sur une machine physique, il existe différents périphériques d'installation : un CD/DVD-ROM, une clé USB, une carte SD ou même depuis un environnement réseau PXE (Preboot Execution Environment) hébergé sur le réseau local de la machine. Pour notre machine virtuelle, nous nous contenterons d'utiliser un lecteur CD-ROM virtuel auquel nous attachons l'image ISO récupérée directement sur le site de Debian. Une fois le programme d'installation lancé, il faut, pour procéder à l'installation manuelle de l'OS, sélectionner le menu d'installation graphique. Une fois ce menu sélectionné, une succession de questions est alors posée à l'utilisateur, séparées pour la majeure partie des questions d'un temps d'installation. Ces questions concernent le choix de la langue, le pays, la disposition du clavier, la configuration réseau, le partitionnement du disque, la configuration du système et du gestionnaire de démarrage. C'est également lors de cette étape que l'on peut décider d'installer différents paquetages logiciels tels que l'environnement de bureau MATE (exigé pour notre projet), un serveur web, un serveur et autres logiciels utiles. Une fois toutes ces questions répondues et une fois les temps d'installations écoulés, le système est prêt à être utilisé. Le principal avantage de cette installation manuelle est l'aspect graphique facilitant grandement l'installation pour des utilisateurs novices. Le principal inconvénient de cette méthode est quant à lui le fait de devoir rester devant son ordinateur afin de pouvoir interagir avec la machine lorsque nécessaire et cela malgré les plus ou moins longs temps d'attente imposés par l'installation du système. Un autre inconvénient est que certaines autres tâches restent encore à réaliser une fois l'installation terminée. Dans notre exemple, à la fin de l'installation, seul l'administrateur du système (root) ne possède les droits sudo². Or, on souhaite que l'utilisateur principal les possède également. Nous devons donc encore procéder à l'ajout de l'utilisateur au groupe sudo.

Ajout de l'utilisateur au groupe sudo Pour ajouter un utilisateur au groupe sudo, il existe différentes méthodes. Nous allons vous en présenter une. La première étape est d'ouvrir un terminal en tant qu'administrateur. Pour cela, il suffit soit d'ouvrir simplement le terminal depuis la session de l'administrateur ou, le cas échéant, de saisir la commande « su - » avant de saisir le mot de passe de l'administrateur. Une fois le terminal ouvert en tant qu'administrateur, il suffit de saisir la commande `sudo visudo`. Cette commande ouvre alors un fichier dans lequel il faut repérer la ligne : `# User privilege specification` et d'ajouter en dessous de cette ligne la chaîne de caractères suivante : `<nom_utilisateur> ALL=(ALL:ALL) ALL`. Il faut alors sauvegarder les modifications du fichier avant de relancer la session de l'utilisateur.

Si on veut faire en sorte que les utilisateurs n'aient pas à saisir leurs mots de passe lors de l'utilisation de la commande sudo, il faut saisir, toujours dans le même fichier, la ligne : `%sudo ALL=(ALL) NOPASSWD: ALL`

0.1.4 Installation automatique de l'OS d'une machine virtuelle

Prérequis indispensables à l'installation La première étape indispensable à la configuration automatique d'une machine virtuelle est de fournir au logiciel de virtualisation (ici VirtualBox) des fichiers de préconfiguration utilisés lors de l'installation afin que l'installation automatique satisfasse nos attentes. Les fichiers indispensables à cette installation nous ont été fournis par le biais de Moodle. Parmi ces

1. **BIOS** : *Basic Input/Output System*. Le BIOS est un logiciel utilisé par le microprocesseur de la machine lors de sa mise sous tension pour permettre le démarrage du système de son système d'exploitation.

2. **Sudo** : *Substitute user do*. Cette commande permet à l'administrateur d'un système Linux d'accorder à certains utilisateurs ou groupes d'utilisateurs la possibilité de lancer la commande en tant qu'administrateur. Les utilisateurs ayant la possibilité d'utiliser cette commande doivent faire partie du groupe sudo du système.

fichiers, on retrouve une image ISO virtuelle associant (grossièrement) les fichiers utilisés lors de l'installation. C'est ce fichier qui doit être ajouté au lecteur virtuel de la machine virtuelle avant l'installation pour que l'installation automatique puisse se lancer. Ce fichier est un fichier d'extension `.viso`.

Pour que l'installation puisse se lancer, on doit ajouter au fichier `.viso` fourni sur Moodle un identifiant UUID permettant à VirtualBox d'identifier la machine virtuelle.

Pour obtenir cet UUID, il existe (sûrement parmi tant d'autres) deux solutions :

- La première est d'exécuter dans un terminal la commande : `cat /proc/sys/kernel/random/uuid` puis de copier le résultat de la commande dans le fichier `S203-Debian12.viso` à côté du champ `--iprt-iso-maker-file-marker-bourne-sh=`
- La deuxième solution, plus compliquée à mémoriser mais plus efficace, est d'utiliser directement la commande `sed` afin que le résultat de la commande précédente soit directement ajouté au bon endroit. Pour faciliter les différentes installations (une installation à chaque fois que nous souhaitons tester une fonctionnalité de l'installation automatique), nous avons décidé d'écrire un script `addUUID.sh` exécutable reprenant la commande `sed` donnée par le sujet :

```
#!/bin/bash
sed -i -E "s/(--iprt-iso-maker-file-marker-bourne-sh).*$/\1=$(cat
/proc/sys/kernel/random/uuid)/" ./S203-Debian12.viso
```

En plus de l'image ISO virtuelle, nous devons, évidemment, fournir à la machine virtuelle l'image ISO de Debian, le fichier de configuration pour l'automatisation (`preseed.cfg`) ainsi que d'autres fichiers utiles à la configuration de l'installation. Une fois tous les fichiers réunis, nous pouvons les déposer dans le répertoire de notre machine virtuelle. C'est donc l'image ISO virtuelle qui va permettre à l'installation d'identifier et de retrouver les différents fichiers énumérés précédemment. Nous avons donc listé les différents prérequis de l'installation automatisée de la machine virtuelle. Nous allons donc maintenant voir comment configurer cette installation.

Configuration de l'installation automatique Les diverses informations et actions à faire automatiquement lors de l'installation sont écrites dans le fichier de préconfiguration nommé `preseed.cfg`. Nous avons donc récupéré une première version de ce fichier. Cette version du fichier contient un peu moins d'une centaine de lignes de configuration fournissant au système des informations relatives au partitionnement du disque, à la langue utilisée, à la disposition du clavier, à la gestion des utilisateurs, à sa configuration réseau et autres. En l'état, ce fichier permet déjà de lancer une installation automatique. Toutefois, le système installé sera assez léger en termes de fonctionnalités utilisables pour nos besoins dans cette SAé, les besoins d'un client ou pour un usage qui nous est propre. Nous avons donc le loisir d'ajouter les fonctionnalités que l'on désire en ajoutant simplement au fichier des lignes de configuration. Pour cette SAé, il nous est demandé d'ajouter à l'utilisateur les droits `sudo` et d'installer un certain nombre de paquets. Pour trouver les différentes lignes de configuration que nous allons détailler dans la suite de ce rapport, nous nous sommes servis de la documentation de Debian relative à son installation.

Ajout des droits `sudo` à l'utilisateur standard Pour ce qui est de l'ajout des droits `sudo` à l'utilisateur nous avons ajouté dans le fichier la ligne : `d-i passwd/user-default-groups string sudo`.

Nous allons décortiquer cette ligne de commande afin de mieux comment elle fonctionne :

- **d-i** : cette valeur est commune à tout les lignes de préconfiguration d'un système Debian.
- **passwd/user-defaults-groups** : ce paramètre spécifie les groupes par défaut pour l'utilisateur.
- **string** : ce paramètre permet d'indiquer la valeur suivante est une chaîne de caractères.
- **sudo** : ce paramètre permet naturellement de demander à la machine d'ajouter l'utilisateur au groupe `sudo`, ce qui octroi de ce fait les droits `sudo` à l'utilisateur (à condition bien sûr que le paquet de gestion des droits `sudo` soit installé).

Il ne nous était pas demandé d'ajouter l'utilisateur à d'autres groupes par défaut toutefois cette manipulation est possible en ajoutant simplement le nom des autres groupes à la fin de cette ligne.

Installation des paquets Pour ce qui est de l'installation automatique des paquets demandés, une solution est d'ajouter les lignes suivantes au fichier de préconfiguration :

```
tasksel tasksel/first multiselect mate-desktop d-i pkgsel/include string sudo git sqlite3
curl bash-completion neofetch
```

Comme pour l'ajout des droits `sudo`, nous allons décortiquer ces deux lignes de configuration :

- **tasksel** : cette valeur fait appel à l’outil de gestion des tâches de Debian permettant notamment d’installer des paquets.
- **tasksel/first** : ce paramètre permet de spécifier au système que cette tâche doit être la première tâche prise en charge par l’outil tasksel.
- **multiselect** : ce paramètre permet de sélectionner plusieurs option.
- **mate-desktop** : nous souhaitons installer l’environnement de bureau Mate, c’est donc ici que nous le précisons au système.
- **pkgssel/include** : cette option permet d’installer d’autres packages de manière individuelle en plus des paquets installés par l’outil tasksel.
- **sudo git sqlite3 ... neofetch** : ces différents paramètres sont les différents paquets installés. La liste nous a été précisée dans le sujet.

Une fois ces différentes configurations mises en place, l’installation correspondante aux attentes du sujet peut-être lancée. Une fois cette installation terminée, la machine est prête à être utilisée pour la suite du travail.

0.1.5 Questions

1. Que signifie “64-bit” dans “Debian 64-bit” ?

“64-bit” fait référence à l’architecture des processeurs et du système d’exploitation qui peuvent gérer des données en unités de 64 bits à la fois. Dans le cas de “Debian 64-bit”, cela signifie que la version de Debian est conçue pour fonctionner sur des processeurs compatibles 64 bits et peut ainsi tirer parti des avantages des capacités de traitement accrues et de la mémoire disponible sur ces systèmes. En effet, un processeur “64-bit”. (PcAstuces)

2. Quelle est la configuration réseau utilisée par défaut ?

Par défaut, Debian utilise un service de gestion réseau appelé “systemd-networkd” cela se trouve alors dans le fichier /etc/network/interfaces avec comme IP statique l’interface Ethernet. (Debian)

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

Le fichier XML se trouve dans /etc/libvirt/qemu/ et se nomme alors : serveur.xml car c’est le nom de notre machine. (Linux.developpez)

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

Oui en allant dans le fichier et en modifiant la ligne avec .

5. __ Qu’est-ce qu’un fichier iso bootable? __ 3. Comparons.

Un fichier ISO bootable est un fichier d’une image d’un disque qui contient une copie des données d’un disque optique comme un CD. Ce fichier permet alors l’utilisation et l’installation du système d’exploitation. (WinZip)

6. Qu’est-ce que MATE ? Gnome ?

MATE est un environnement de bureau assez intuitif cela permet d’avoir un gestionnaire de fichiers des navigateurs web, des lecteurs multimédias etc .. GNOME est un environnement plus moderne qui est beaucoup plus utilisé pour Linux. (Mate-Desktop)

7. Qu’est-ce qu’un serveur web ?

Un serveur web est un logiciel conçu pour répondre aux requêtes HTTP principalement sur un navigateur web. (Developer Mozilla)

8. Qu’est-ce qu’un serveur ssh ?

Un serveur SSH est un logiciel qui permet aux utilisateurs de se connecter de manière sécurisé à distance. (ITConnect)

9. Qu’est-ce qu’un serveur mandataire ?

Un serveur mandataire est un proxy qui permet un relais entre les utilisateurs et les serveurs auxquels ils accèdent sur internet. (Techno Science)

10. Comment peux-t’on savoir à quels groupes appartient l’utilisateur user ?

Pour savoir à quels groupes appartient user on fait groups user (Wiki Debian)

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

Cela sert à améliorer les performances et les fonctionnalités des machines virtuelles :

- Une meilleure intégration hôte-invité
- L'amélioration des performances

12. À quoi sert la commande mount (dans notre cas de figure et dans le cas général) ?

La commande mount est utilisée pour monter un système de fichiers dans l'arborescence de répertoires du système d'exploitation. Dans notre cas, elle est souvent utilisée pour monter des systèmes de fichiers supplémentaires ou des dispositifs de stockage amovibles. (Debian FR)

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

Le projet debian est une organisation communautaire qui développe le système d'exploitation debian GNU/Linux et est connue pour sa stabilité sa sécurité et sa diversité de logiciels. (Debian)

14. Il existe 3 durées de prise en charge (support) de ces versions : la durée minimale, la durée en support long terme (LTS) et la durée en support long terme étendue (ELTS). Quelle sont les durées de ces prises en charge ?

- Durée minimale de prise en charge : environ 5 ans
- Support à long terme : 5 ans supplémentaires
- Support à long terme étendu : 5 années supplémentaires

(Wiki Debian)

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

Environ 5 ans. (Debian)

16. Combien de version au minimum sont activement maintenues par Debian ? Donnez leur nom générique (= les types de distribution).

3 versions : stable, testing et unstable. (Debian)

17. Chaque distribution majeure possède un nom de code différent. Par exemple, la version majeure actuelle (Debian 12) se nomme bookworm. D'où viennent les noms de code données aux distributions ?

Cela vient des personnages de Toy Story. (Debian)

Stabilité	Version	Personnages
Stable	Debian 12	Bookworm
Oldstable	Debian 11	Bullseyes
	Debian 10	Buster
	Debian 9	Stretch
	Debian 8	Jessie
	Debian 7	Wheezy
Obsolète	Debian 6	Squeeze
	Debian 5	Lenny
	Debian 4	Etch
	Debian 3,1	Sarge
	Debian 3	Woody
	Debian 2,2	Potato

Stabilité	Version	Personnages
	Debian 2,1	Slink
	Debian 2,0	Hamm
Inconnue	Debian X	Trixie

18. **L'un des atouts de Debian fut le nombre d'architecture (processeurs) officiellement prises en charge. Combien et lesquelles sont prises en charge par la version Bullseye ?**

Il y en a actuellement environ 13. (Debian)

19. **Première version avec un nom de code**

- **Quelle a été le premier nom de code utilisé ?**
- **Quand a-t-il été annoncé ?**
- **Quelle était le numéro de version de cette distribution ?**

La première était Buzz annoncé le 17 juin 1996 avec comme numéro 1.1 (Wikipédia)

20. **Dernier nom de code attribué**

- **Quel est le dernier nom de code annoncée à ce jour ?**
- **Quand a-t-il été annoncé ?**
- **Quelle est la version de cette distribution ?**

La dernière est Bookworm annoncé le 14 août 2023 avec comme numéro 12. (Wikipédia)

0.2 Semaine 8

0.2.1 Description du travail de la semaine

Pour ce qui est maintenant de la semaine 8, nous allons procéder à la recherche et à l'analyse d'applications clientes en gérant et en installant des dépôts git par le biais d'une interface graphique. Il est donc attendu de nous d'apprendre à utiliser git. Pour cela, il est possible de lire des documentations qui ne sont toutefois pas forcément toujours fournies. Nous allons évidemment également utiliser nos compétences acquises à l'aide de la ressource R2.03 de qualité de développement. Une autre solution est de tester différentes commandes et techniques sur nos propres fichiers en créant des dépôts git.

Nous pouvons donc installer l'outil git, qui est libre de droits, ainsi que les applications graphiques officielles associées et les comparer ensuite une à une.

0.2.2 Configuration de git

La première étape, une fois git installé, consiste à le configurer. Ces configurations sont légères et permettent de fournir des données sur l'utilisateur lors des différents commits³ et push⁴. Cette étape se fait avec les exécutions des deux commandes suivantes :

```
git config --global user.name "<username>"
git config --global user.email "<useremail>"
```

La seconde étape permet, elle, d'installer les logiciels **gitk** et **git-gui**. Cette étape consiste elle aussi à la simple exécution des commandes :

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

3. **Commit** : Les commits constituent les piliers d'une chronologie de projet Git. Les commits peuvent être considérés comme des instantanés ou des étapes importantes dans la chronologie d'un projet git. Ils sont créés grâce à la commande `git commit` pour capturer l'état d'un projet à un point dans le temps. (Atlassian)

4. **Push** : Concrètement un push permet d'uploader les différents commits réalisés de manière locale sur le dépôt distant.

L'utilisateur doit évidemment posséder les droits sudo ou être administrateur du système pour exécuter ces commandes

Une fois les différentes installations et configurations accomplies nous pouvons les tester afin de constater de la bonne réalisation, ou non, des précédentes étapes. Pour cela, nous allons tout simplement créer un dépôt git et y ajouter une multitude de fichiers et de documents. A l'aide de l'interface graphique, nous pouvons par la suite constater les différents commits réalisés dans ce dépôt.



Pour bien tenir à jour son dépôt git, il est conseillé après l'implémentation de fonctionnalités dans un projet ou bien après une avancé significative de faire un commit, puis d'uploader les fichiers sur le dépôt existant (si ce dernier existe).

```
git add . #Pour ajouter les fichiers au commit
git commit -m "message associé au commit" #Pour réaliser le commit
git pull #Permet de récupérer la dernière version sur le dépôt distant et d'éviter les conflits
git push #Permet d'uploader les différents commits sur le dépôt distant
```

Concernant l'interface graphique utilisé pour le git nous avons choisi Magit qui est un outil gratuit et assez simple d'utilisation permettant d'avoir des informations détaillées et une belle image d'ensemble du dépôt git. Après des tests sur nos fichiers en créant des dépôts git et en comparant l'interface graphique de base et git et celle que nous avons choisi nous avons repéré les avantages suivants : - D'abord git Magic est basé sur des lignes de commande qui offrent plus de flexibilité que l'interface graphique de base. - Un des autres avantages est la personnalisation disponible qui permet de modifier l'environnement avec des fichiers de script de git magics - On peut également automatiser des tâches avec git magics ce que l'on ne peut pas faire avec l'interface de base Les inconvénients que nous avons pu trouver sont que les lignes de commande à écrire pour effectuer des tâches peuvent être plus complexes à comprendre grâce à la grande disposition de personnalisation disponible mais après des recherches et des tests effectués il est plus simple à comprendre.

0.2.3 Questions

1. Qu'est-ce que le logiciel gitk ? Comment se lance-t-il ?

Gitk est une interface graphique pour git cela permet de visualiser l'historique des commits, les branches tags etc.. Il est utile pour explorer l'historique d'un projet. Et cela se lance avec la commande "gitk". (Codeur-Pro)

2. Qu'est-ce que le logiciel git-gui ? Comment se lance-t-il ?

Git gui est un système de controle de version contrairement à gitk qui se concentre sur la visualisation il offre une interface graphique pour effectuer différentes variété d'opérations Git et cela se lance avec la commande "git gui". (Codeur-Pro)

3. Pourquoi avez-vous choisi ce logiciel ?

Nous avons choisi le logiciel Magic car il est tout d'abord gratuit sur l'utilisation courante que l'on souhaite mais aussi simple d'utilisation tout en fournissant toutes les informations nécessaire sur ce que nous avons besoin. (Git-scm)

4. Comment l'avez vous installé ?

Pour installer cela on fait ces commandes suivantes :

```
git clone https://github.com/blynn/gitmagic.git
cd gitmagic
./install.sh
```

0.3 Semaine 10

0.3.1 Description du travail de la semaine

Cette semaine nous allons découvrir l'installation du service gitea. Gitea est d'abord une plateforme de gestion de code il fonctionne sur le même principe que github et gitlab avec de la collaboration. Il s'agit alors d'un logiciel open-source qui permet à des équipes de travailler ensemble efficacement sur des projets. Gitea est populaire sur les petites équipes de développement pour des projets open sources car il est léger et facile de configuration.

0.3.2 Configuration de la machine

Pour cela nous allons d'abord commencer par faire la redirection de port de notre machine virtuelle cela permet d'éviter des problèmes sur l'utilisation plus tard de tests que l'on souhaiterait faire. Nous allons alors aller dans la configuration de la VM puis aller dans réseau puis cliquer sur redirection de port ensuite il faut cliquer en haut à droite pour ajouter une redirection. On nomme alors notre redirection gitea et on lui met un port hôte ainsi qu'un port invité à 3000. On peut ensuite redémarrer la machine virtuelle.

Installation de gitea sur la machine

Ensuite nous devons procéder à l'installation du binaire pour cela nous allons sur la documentation des étapes d'installation pour récupérer le fichier contenant l'extension `.linux-amd64` dans la dernière version disponible ici dans notre cas il y en a un seul étant `gitea-1.21.7-linux-amd64.xz` on le télécharge alors sur notre machine virtuelle puis on suit les instructions.

A partir du fichier téléchargé on peut alors décompresser l'archive en le mettant dans un dossier créé pour gitea puis on donne la permission avec :

```
sudo chmod +x gitea-1.21.7-linux-amd64
```

Il faut aussi vérifier que notre version de git est compatible avec celle de gitea pour cela nous faisons `git --version` et il faut que la version soit supérieur à 2. On trouve toutes ces instructions sur la documentation d'installation par fichier binaire de gitea : (Doc-Gitea)

Une fois le fichier installé et les permissions mises il suffit alors de le lancer en suivant les instructions de la documentation, on arrive alors ensuite lors du premier lancement sur le paramétrage de gitea grâce à la partie web il faut alors mettre comme base de données `sqlite3` et comme compte administrateur : nom `gitea`, password `gitea` et comme email `git@localhost`

La documentation de gitea nous informe ensuite qu'il faut protéger les fichiers `/etc/gitea` et `/etc/gitea/app.ini` avec les commandes suivantes :

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

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

0.3.3 Création du projet

Pour créer un projet il suffit sur la partie web de se connecter au compte gitea et d'aller sur l'option nouveau projet et de remplir les détails du projet comme le nom la description et les options de visibilité. On peut ensuite configurer comme paramètre de visibilité comme lecture et ajoutant dans le projet les sujets de TP nous avons alors ensuite tenter de visualiser le projet pour confirmer qu'il était accessible. Nous avons effectué une série de tests pour vérifier que les utilisateurs avaient les droits appropriés sur nos projets. Ces tests comprenaient la modification des autorisations pour différents utilisateurs et la vérification des actions possibles pour chacun. Nous avons documenté chaque test, en notant les résultats réussis et les problèmes rencontrés. Par exemple, nous avons constaté qu'un utilisateur pouvait voir le projet mais ne pouvait pas le modifier, ce qui indiquait une erreur dans la configuration des autorisations. Nous avons ensuite vérifié la configuration de la redirection de port pour l'accès à gitea depuis d'autres interfaces ce que nous avons mal fait après une erreur d'écriture sur la redirection pendant la configuration de la machine virtuelle.

0.3.4 Questions

1. Qu'est-ce que Gitea ?

Gitea est d'abord une plateforme de gestion de code il fonctionne sur le même principe que github et gitlab avec de la collaboration. Il s'agit alors d'un logiciel open-source qui permet à des équipes de travailler ensemble efficacement sur des projets. Gitea est populaire sur les petites équipes de développement pour des projets open sources car il est léger et facile de configuration. (Doc-Gitea)

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

Comme dit précédemment on peut comparer gitea à gitlab et github qui sont également des outils de développeurs qui permettent d'héberger des codes sources et de travailler en équipe. (Doc-Gitea)

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

Nous avons pris la version 1.21.7 et le fichier gitea-1.21.7-linux-amd64.xz pour voir la version du fichier binaire installé sur notre VM il suffit de faire la commande `gitea -version` (Doc-Gitea)

4. Comment faire pour mettre à jour le binaire de votre service sans devoir tout reconfigurer ? Essayez en mettant à jour vers la version 1.22-dev.

Pour cela il suffit de télécharger la nouvelle version du binaire Gitea à partir du site officiel de gitea puis de le remplacez pas l'ancien et lui donner les permissions nécessaires. Il faut ensuite redémarrez le service Gitea pour prendre en compte les changements de version du fichier binaire. (Doc-Gitea)