**TP ASR2**

**Partie 1: Installation d’Ubuntu 18**

1. **Notion de distribution**: une distribution linux est un système (d’exploitation) conçu à partir du noyau Linux. Par exemple: Kali, Debian ou Ubuntu (qui est le système avec lequel nous allons travailler)
2. **Notion de netinstall**: pour installer ubuntu, nous devons télécharger un fichier image (extension *.iso*), qui contient tous les paquets (ensemble de fichiers) nécessaires à l’installation du système d’exploitation sur notre machine. Ce fichier image est généralement très lourd (~2Go) en fonction de la version du SE et du type d’installation souhaitée (bureau ou serveur).

Cependant, il existe des images plus légères, dites *mini.iso*: on parle alors de *netinstall*. Netinstall est un type d’installation utilisant une image minimale qui va télécharger tous les paquets nécessaires aucours de l’installation. Ceci nécessite d’avoir une connexion haut débit pour permettre le téléchargement de ces paquets. L’utilisateur est donc libre de choisir les paquets qu’il souhaite installer, et de configurer le réseau à sa guise au cours de la procédure d’installation.

1. **Notion de partition**: une partie est juste une partie du disque. Lorsqu’on installe un système d’exploitation on doit définir des régions précises dont la plus importante est celle sur laquelle sera installé le système lui même (partition C pour Windows par exemple). Les partitions importantes à définir lors de l’installation d’Ubuntu sont:
   1. Le **swap**: c’est un espace d’échange qui per met d’étendre la mémoire (RAM) de notre machine. Afin d'éviter un blocage de votre ordinateur lorsque sa RAM est pleine, Ubuntu se sert de cette partition pour décharger temporairement la RAM.

La taille du swap se calcule en fonction de celle de la RAM: si nous avons moins d’1 Go de RAM, entre 1,5× et 2× la taille de la RAM; si nous avons plus d’1 Go de RAM, de 1× à 1,5× la taille de la RAM. Cette partition est de type *SWAP*.

* 1. La **racine (/):** la partition racine est la base de l'arborescence du système Ubuntu. Par défaut, si aucun réglage n'est changé, c'est dans celle-ci que tous les fichiers vont être placés : fichiers de configuration, programmes, documents personnels, etc.

Cette partition est de type *EXT4* et sa taille doit être au minimum de 8Go.

1. **Installation proprement dite** (durant la séance)
2. Une fois l’installation terminée:
   1. La première chose à faire est de mettre à jour les paquets. Pour cela:
      1. Ouvrir le fichier *sources.list* et décommenter certaines lignes (voir en TP):

vi_sources_list

* + 1. Faire la mise à jour:

update

* + 1. **APT** signifie *Advanced Package Tool.* Cette interface fonctionne avec les dépôts (bibliothèques) de base pour gérer l'installation et la suppression de logiciels sur Debian, Ubuntu et les distributions Linux associées. APT simplifie le processus de gestion des logiciels sur les systèmes informatiques de type Unix en automatisant la récupération, la configuration et l'installation des progiciels, à partir de fichiers précompilés ou en compilant le code source.

Pour rechercher un paquet dans la liste des paquets connus par APT, on utilise la commande *apt-cache search*.

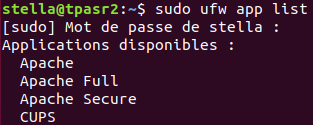
1. **Installation d’Apache**: Apache est un serveur HTTP, i.e. il permet à un site web de communiquer avec un navigateur en utilisant le protocole HyperText Transfer Protocol (HTTP).
   1. Installation d’apache2:

install_apache

* 1. Vérification de l’installation:

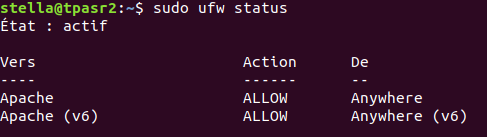
apache_version

* 1. Configuration du firewall: afin de configurer Apache, nous devons d'abord autoriser l'accès extérieur à certains ports Web de notre système et autoriser Apache sur notre pare-feu UFW (Uncomplicated FireWall).
     1. Lister les profiles d’application ufw:



* + 1. Autoriser apache sur le pare-feu et vérifier le status:

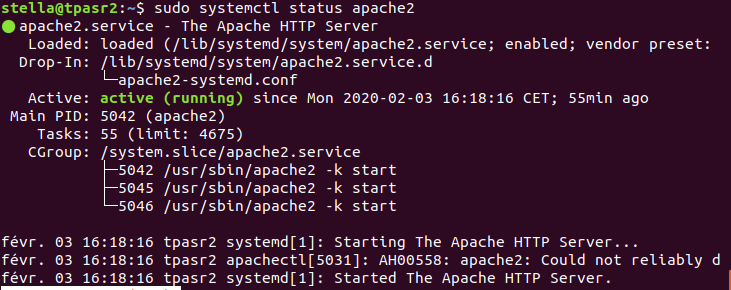
ufw_allow



Si le résultat de *status* vous renvoie *Etat: Inactif*, alors tapez ceci:

ufw_enable

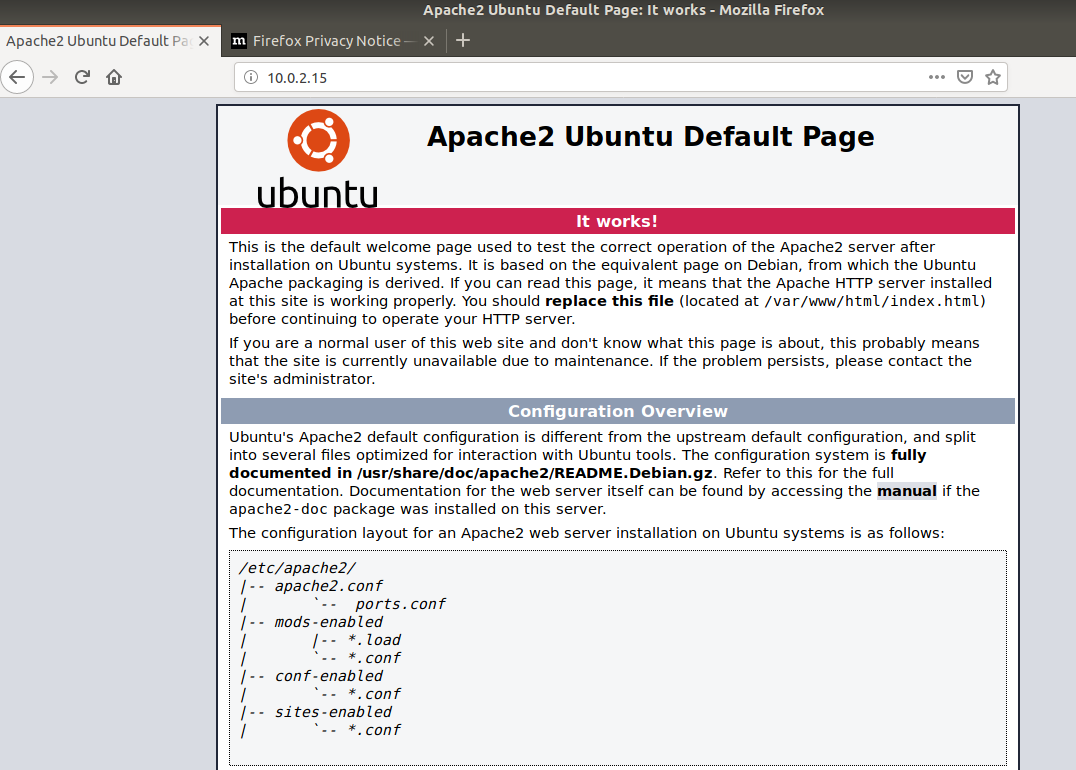
* 1. Configuration du serveur Apache:
     1. Vérifions que le service apache est actif et en train de s’exécuter:



* + 1. Vérifions que le server écoute bien sur notre adresse IP. Pour connaître son adresse IP faire:

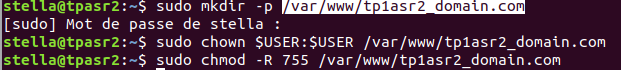
hostname

Ensuite depuis un navigateur rentrer cette adresse comme ceci: *<http://ip_address>*



* 1. Création de notre site:
     1. Installation d’un nom de domaine: Apache est installé avec un site par défaut (voir le dossier /var/www/html/index.html: c’est le code de la page par défaut qui s’est affichée précédemment). Nous allons créer un nouveau dossier dans /var/www/html pour installer notre domaine:

domain_com



* + 1. Création de la page d’accueil de notre site (index.html):

vim_domain_index

<html>

<head>

<title>Welcome to the page tp1asr2\_domain.com!</title>

</head>

<body>

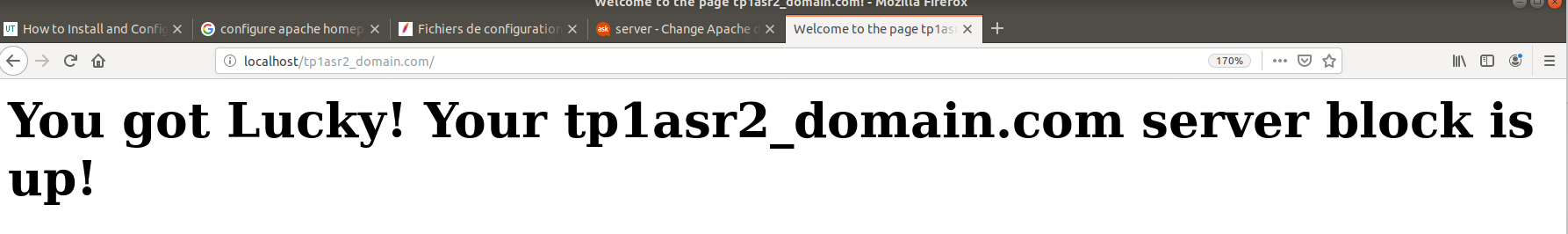
<h1>You got Lucky! Your tp1asr2\_domain.com server block is up!</h1>

</body>

</html>

* + 1. Redémarrage du serveur et accès à notre site depuis le navigateur:

restart_server



**Partie 2: Exploration du disque**

1. **Explorer les partitions avec fdisk** 
   1. **fdisk**  est un utilitaire permettant de gérer les partitions disques. Tout d’abord visualiser les différent périphériques de stockage présents (ce sont les devices):



b/ Les fonctionnalités de fdisk s’appliquent sur un disque en mode interactif, par exemple créons une partition sur un disque ( par exemple /dev/sdb).

NB: Nous pouvons le faire dans notre machine virtuelle , pour ce faire il est préférable de lui associer un nouveau disque virtuel via l’interface de virtualbox. Ce nouveau disque servira de cobaye pour la suite. Suivre les instructions suivantes.

i/ Arrêtez votre machine virtuelle

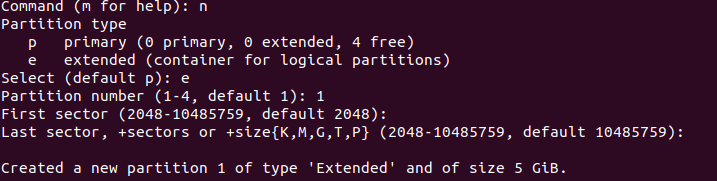
ii/ Allez dans le panneau de configuration de votre machine virtuelle pour y ajouter un disque virtuel de 5 GB.

iii/ Une fois le disque virtuel ajouté, démarrer votre machine

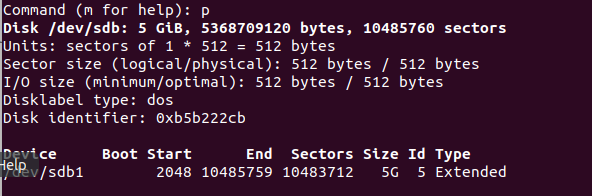
iv/ sélectionner le nouveau disque créé via fdisk



iii/ Ensuite répondre par un n pour créer la nouvelle partition, puis par e pour spécifier qu’il s’agit d’une partition étendue:



iv/ faire enfin un p pour afficher la nouvelle partition créée puis un q pour quitter .



c) Formater la nouvelle partition à l’aide de la commande mkfs.ext3



d) cette nouvelle partition sur le disque /dev/sdb est prête à l’emploi. On peut donc la monter sur un dossier du système de fichier pour l’utiliser.

i) créer le dossier sur lequel monter la partition



ii) monter la partition sur ce dossier



2- Explications /etc/fstab : Il s’agit d’un fichier contenant tous les montages nécessaire au bon fonctionnement du système. Il est lu à chaque que fois que la commande mount est utilisé par exemple lors du démarrage. D’autres informations comme le format se diront séance tenante.

3 - Utilisation de dd pour la sauvegarde du disque dur de la vm.

i/ Arreter la VM

ii/ se déplacer dans le dossier où se trouve le disque de la vm (il est préférable d’utiliser le disque virtuel créé plus haut)

iii/ sauvegarder le disque de la vm



iv/ Utiliser dd pour écrire des zéros sur ce disque



v/ Essayer de redémarrer la VM (Vous devez avoir une erreur liée à ce disque)

vi) Recouvrer le disque altéré



vii/ Redémarrer la VM. Tout fonctionne :)

**Partie 3: Compilation du noyau linux**

1. **Compilation du noyau linux :**  Il s’agit de compiler un nouveau noyau en ayant sous la main le code source. C’est très utile pour modifier certaines fonctionnalités du noyau linux. Nous installeront la version 5.4.1 disponible en téléchargement via le lien <https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.4.1.tar.xz>

**i / Télécharger le code source**

****

**ii/** décompresser l’archive téléchargée





iii/ se déplacer dans le dossier décompressé qui contient le code source de linux



iv/importer le fichier de configuration par défaut de votre noyau courant



v/ installer les utilitaires de compilation



vi) réaliser la configuration



vii) compiler le noyau (avec éventuellement un certain nombre de processeurs, option -j). Pour accélérer la compilation, arrêter la VM et configurer le nombre de processeurs.



NB : Avant d’installer le nouveau noyau, s’assurer

* que le fichier /boot/grub/grub.cfg ne contient aucune référence à la version du noyau que nous voulons installer.
* que le dossier /boot ne contient ni le binaire de notre nouveau noyau, ni son système de fichier.
* de noter la version actuelle du noyau installé.

viii) Installer les modules puis le noyau compilé 



mettre à jour le grub pour qu’il prenne en compte le nouveau noyau lors du démarrage de la machine virtuelle





ix) redémarrer la VM puis vérifier que le nouveau noyau a bien été pris en compte. (Revoir tous le points du NB plus haut. )



x) supprimer le vmlinuz de l’ancien noyau (Désormais, il sera impossible de booter dessus à nouveau)



1. Dézipper le kernel installé avec l’outil extract-vmlinux let retrouver dans le binaire obtenu à l’aide de l’outil objdump une fonction contenue dans le code source du noyau, par exemple “start\_readonly”.

i/ installation de l’outil kernel-extract



ii/ copier le fichier vmlinux du répertoire /boot vers le répertoire courant



iii/ Extraire le binaire du noyau linux



iv) rechercher l’adresse de la fonction start\_readonly dans le fichier /boot/System.map-xxxxx



v) Afficher son code assembleur dans le binaire à cette adresse.



1. Explorer le mapping mémoire système-périphériques dans le fichier /proc/iomem

