

Date limite du Compte Rendu : **dimanche 12 avril 2020** pour **TOUS**

H. TSOUNGUI

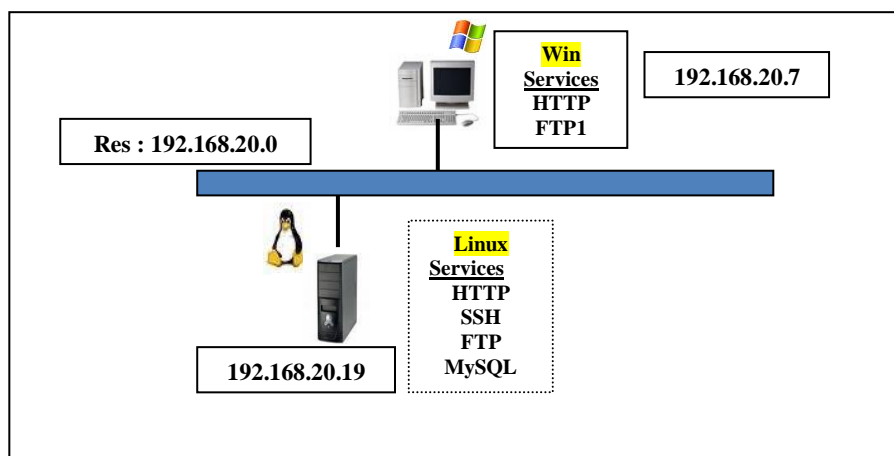
1)-Réalisation du réseau TCP/IP

Architecture : mettre en place l'architecture représentée ci-dessous

-Réseau:192.168.20.0/255.255.255.0 et

-Deux machines : **win (quelconque)**, et **linux (ubuntu18 ou 19 conseillé)**.

Chaque hôte du réseau aura **deux** cartes d'interface réseau à configurer correctement.



Etapes :

- Configuration TCP/IP de chacune des machines : **adresse IP et masque.**

-Test de communication (commande PING cible) entre les deux machines : de chaque machine, faire **ping adresse_ip** de l'autre machine Par ex : depuis win7, dans l'invite de commandes ping 192.168.20.19
Même chose à partir de linux. Si pas de réponse, désactiver le pare-feu de win7.

-Visualisez la configuration actuelle des machines **en utilisant ipconfig sous win7 et ifconfig sous linux**

2)-DNS minimal

Le service DNS sert à convertir les adresses IP de composants en noms plus faciles à retenir. Ainsi on peut faire des pings sur les noms ou accéder aux sites sans utiliser l'adresse IP.

Pour mettre en place le mini-DNS, il faut :

a)-sous windows ouvrir avec le bloc notes, en tant qu'administrateur, le fichier hosts qui se trouve dans **C:\windows\system32\drivers\etc** et le **modifier comme ci-dessous** :

```
# -----
# adresse IP      nom_court.DOMAINE      nom_court      alias
192.168.20.7     win7.LICENCE2.NET      win7           wincli          (1)
192.168.20.19   xubuntu19.LICENCE2.NET xubuntu19     servlinux       (2)
# Ne pas mettre (1) et (2) ci-dessus
```

```
127.0.0.1      localhost      # Adresse par défaut de la machine (boucle loopback)
```

```
# -----
```

Chaque ligne de ce fichier donne : l'adresse IP, le **nom_court** et le domaine séparés par **un point**, le **nom_court**, et des **alias** ou synonymes du nom de la machine.

-Ajoutez une ligne qui permettra de pinger google.fr avec l'alias **gog**. C'est-à-dire faire ping gog et avoir un résultat au lieu de faire ping google.fr.

b)-sous linux

* Lancez la commande qui permet de trouver le fichier appelé hosts **find ...**

Si vous l'avez trouvé, ouvrez-le avec un éditeur comme nano et ajoutez les deux lignes (1) et (2) ci-dessus.

Ces deux fichiers doivent avoir le même contenu sous linux et windows.

-Faites des **commandes ping sur les noms des différentes machines**. Capturez les résultats.

Ex : ping xubuntu ping win7

3)-Installation du service FTP et test

a)-sur la machine win7

-Installez le petit programme serveur FTP appelé **TypSoftFTP**. Créez les utilisateurs dumont et alain avec **TypSoftFTP** dans le menu Server et –User –new (voir <http://tsoungui.fr/cours> mot clé FTP/win). En vérifiant le serveur FTP, s'il ne tourne pas, changez le port par défaut (mettez 210 au lieu de 21).
-Sur C:\, créez un répertoire appelé FTP et comportant deux sous-répertoires dumont et alain.
Créez ces deux utilisateurs sous linux et donnez-leur les mots de passe dumont2020 et alain2020.

b)-sur la machine linux, vous avez installé le paquet **vsftpd** dans le TD 1.

-Installez maintenant le paquet **gftp**. C'est un clientftp linux qui permet de se connecter à un serveur ftp à l'image de filezilla que vous installez sur la machine win7 (ou sur votre machine réelle si vous n'utilisez pas deux machines virtuelles).

c)-A partir de windows, connectez-vous en FTP avec Filezilla. Pour cela vous aurez besoin d'un nom d'utilisateur qui a un compte sur linux et de son mot de passe linux. Vous pouvez créer les mêmes utilisateurs que sous windows ci-dessus. Le port par défaut est le port TCP numéro 21.

Envoyez quelques fichiers de windows à linux et **capturez les résultats**.

-Téléchargez aussi quelques fichiers à partir de votre répertoire (/home/vous) linux avec l'outil graphique gftp.

-A partir de linux : essayez de faire des transferts de fichiers avec les **commandes FTP** en ligne : **pwd, ls, dir, mdir, get, mget, put, mput, send, status, delete**, etc. Tapez ftp + ENTREE puis **help** pour avoir la liste des commandes FTP.

4)-Linux serveur : installation du service HTTP

-Sur la machine linux, installez le paquet **apache2** qui rendra votre machine serveur HTTP.

-après l'installation du paquet précédent, le serveur tourne.

Pour le vérifier, lancez un navigateur comme firefox sur <http://localhost> ou bien <http://127.0.0.1> ou encore <http://192.168.20.19> vous devriez avoir la page par défaut « It works ! ».

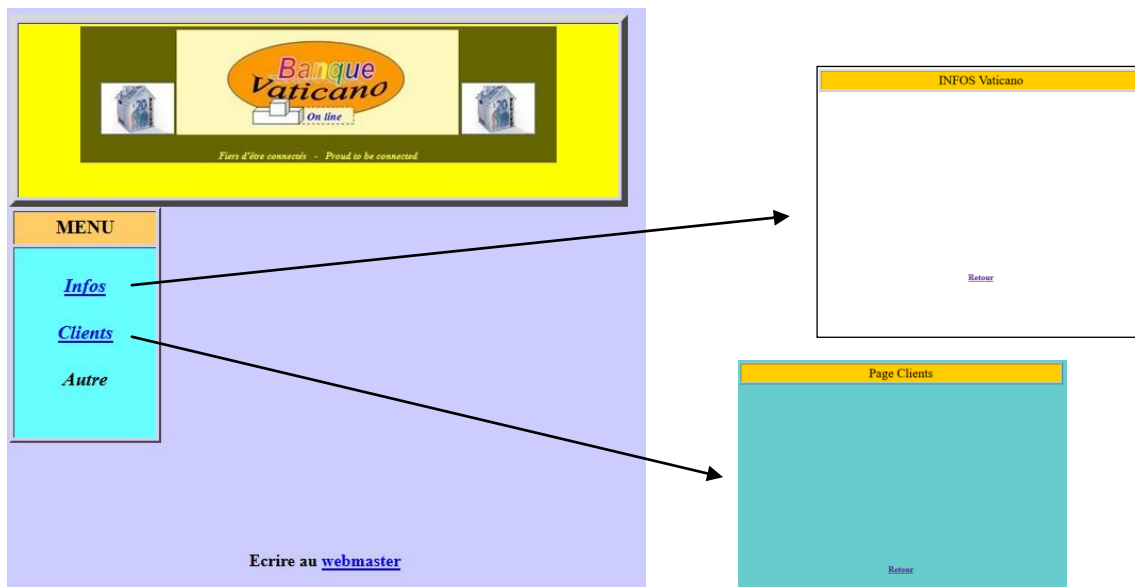
La page par défaut se trouve dans le répertoire /var/www/html.

-Dans un terminal où vous êtes sudo, allez dans ce répertoire et copiez le seul fichier qui s'y trouve et qui s'appelle **index.html** en le nommant la copie **index.html.old**. Tapez **ls -l**, vous devriez avoir DEUX fichiers de noms index mais avec des extensions différentes.

- créez un « site » de 3 pages similaire aux images ci-dessous.

-Le click sur **webmaster** doit déclencher le démarrage d'un logiciel client de mail.

-Chaque page annexe propose un lien **retour** à la page de menu. Accédez à ce s « site » à partir de win7



-Téléchargez le soft **seamonkey** pour linux (format .bz2) et installez-le sous linux.

-Comment installer un paquet *.bz2 ... cherchez et vous trouverez ! Je mets une vidéo sur l'utilisation de seamonkey sur mon site. Noms des pages annexes **infos.html** et **clients.html**

5)-Installation du service SSH

Le service SSH peut être installé avec le paquet **openssh-server**. Il permet de se connecter au serveur SSH de manière sécurisée en déclarant son login et son mot de passe. Il existe plusieurs clients SSH parmi lesquels **secure-shell-Client** sous win et **putty**.

-Sur le serveur linux, installez le paquet **openssh-server**.

-Sur la machine win7 installez **secure-shell-client** ou **putty** pour win.

-Lancez votre client ssh et connectez-vous au serveur linux. **capturez les résultats**.

-Si vous n'y arrivez pas, allez sur un terminal linux, passez **sudo** et tapez **xhost +** puis **ENTREE**. Vous devriez avoir un message vous indiquant que les utilisateurs peuvent désormais se connecter à distance.

-Trouvez le fichier de configuration de **openssh-server** (**ssh*...conf**) et modifiez-le pour empêcher toute connexion distante d'un utilisateur **root**.

6)-Linux comme serveur de bases de données MySQL

-Sur le serveur installez le paquet **mysql-server** avec la commande adéquate ...

-Installez aussi un client MySQL **emma** pour linux et sur win7, installez le client MySQL appelé **SQLyog**. Son utilisation est traitée sur mon site.

-Pour commencer à travailler sous mysql, tapez dans un terminal **service mysql status** pour visualiser l'état du serveur. Il est **Running**, c'est parfait.

Il faut savoir qu'après l'installation du serveur, l'administrateur **root** n'a pas de mot de passe !

-Connectez-vous avec la commande

mysql -u root +ENTREE **capturez les résultats**.

Ensuite donnez un mot de passe avec

>GRANT ALL ON *.* to 'root'@'%' IDENTIFIED BY 'password' ;

Créez aussi un simple utilisateur appelé **dupont** par exemple avec la commande :

>GRANT ALL ON *.* to 'dupont'@'%' IDENTIFIED BY 'dupont2020' ; N'oubliez pas de vous créer vous-mêmes !

Votre serveur est prêt et vous allez vous entraîner en créant la BDD suivante

<http://tsoungui.fr/cours/scripts-mysql-dbcomptes.pdf>

-Essayez de vous connecter à distance par SSH pour écrire et exécuter quelques requêtes SQL sur cette base.

Travail à rendre seul ou en binômes Groupe 1 et Groupe 2 :

dimanche 12 avril 2020

Bon courage !

H. T.