

RAPPORT DE PROJET

PROJET d'UF

Projet réalisé par :

MAHAMAT NAIM
&
KAFI AYMAN

Projet encadré par :

Pr. AARAM Anas

SOMMAIRE

I- LES OBJECTIFS DU PROJET

- a) Objectifs Techniques
- b) Timeframes du projet

II- PROJECT MANAGEMENT

- a) Planification du projet
- b) Objectifs du projet

III- CONFIGURATION RÉSEAUX ET SYSTÈMES

- a) Configuration réseaux
- b) Configuration système

I-OBJECTIFS ET BESOINS

a) Les objectifs techniques demandés

La société LIPS Taiwanese demandant pour nous mettre en place un système de sauvegarde qui serait mis en exécution dans les salles de serveurs de production.

b) Le Timeframe

Ce projet débuta le jeudi 17 Mars et se termina le vendredi 15 Avril 2022, soit le jour de la présentation ; une durée de 28 jours. Afin de terminer ce projet ambitieux à temps, il est important de correctement le gérer et de le tenir à jour grâce aux outils de gestion adéquats.

II- GESTION DE PROJET

a- La planification du projet

Pour suivre le développement du projet dès sa première étape, des séances sur l'Infrastructure et Système d'information nous ont été enseignées par notre professeur.

Ces cours ont pu bien sûr nous guider pour établir les méthodologies à suivre durant ce projet.

b- Les objectifs du projet

Après une analyse approfondie du projet nous avons pu fixer les objectifs du projet qui sont cités ci-dessous :

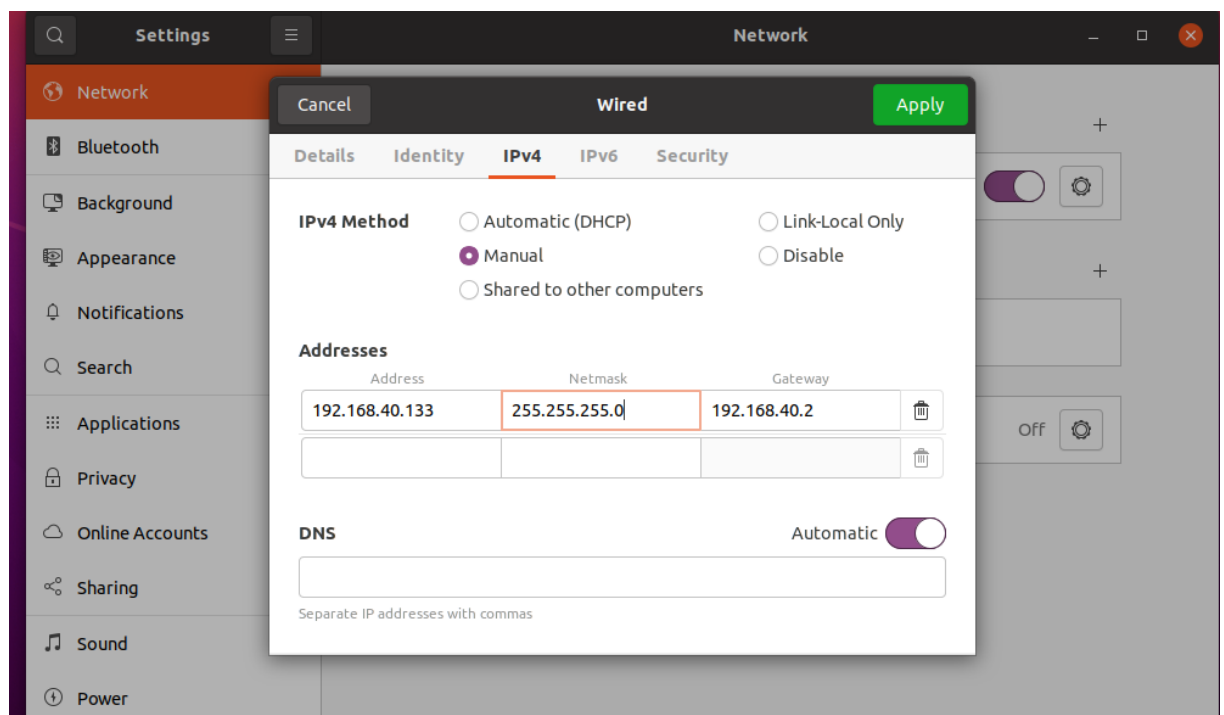
- ADMINISTRER NOTRE POSTE.
- CONFIGURER UN RÉSEAU SIMPLE.
- GÉRER UN ENVIRONNEMENT VIRTUEL.
- METTRE EN PLACE UNE INTERACTION CLIENT-SERVEUR.
- APPRÉHENDER LA SÉCURITÉ.

III – CONFIGURATION RESEAUX ET SYSTEMES

a) Configuration réseaux

En cette partie, nous vous présentons les étapes nécessaires qui nous ont aidé à configurer notre réseau.

Mettre en place un réseau privé NAT et l' allocation de l'adresse IP



La configuration du serveur

```
GNU nano 4.8 /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    ens33:
      addresses: [192.168.40.130/24]
      gateway4: 192.168.40.129
      nameservers:
        addresses: [4.2.2.2, 8.8.8.8]
  version: 2
```

La mise en liaison entre nos deux machines virtuelles,
côté serveur et côté client.

```
client@ubuntu:~$ ping 192.168.40.130
PING 192.168.40.130 (192.168.40.130) 56(84) bytes of data.
64 bytes from 192.168.40.130: icmp_seq=1 ttl=64 time=1.02 ms
64 bytes from 192.168.40.130: icmp_seq=2 ttl=64 time=1.03 ms
64 bytes from 192.168.40.130: icmp_seq=3 ttl=64 time=0.970 ms
64 bytes from 192.168.40.130: icmp_seq=4 ttl=64 time=1.25 ms
64 bytes from 192.168.40.130: icmp_seq=5 ttl=64 time=0.976 ms
64 bytes from 192.168.40.130: icmp_seq=6 ttl=64 time=0.946 ms
64 bytes from 192.168.40.130: icmp_seq=7 ttl=64 time=1.04 ms
^C
```

b) La configuration réseau

Automatisation des CRON JOBS

```
GNU nano 4.8 /tmp/crontab.G0utdE/crontab Modified
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
0 6 * * 1,5 /home/client/make_backup.sh
```

Rédiger le script backup de sauvegarde

```
client@ubuntu: ~
GNU nano 4.8 make_backup.sh
function perform_backup {
    #get backup location
    backup_path=$(cat /backup_locc.conf)

    echo "Starting backup..." > $LOG_LOC
    #for each dir , archive and compress to backup location
    while read dir_path
    do
        #get backup dir name
        dir_name=$(basename $dir_path)

        #create filename for compressed backup
        filename=$backup_path$dir_name.tar.gz

        #archive dirs and compress archive
        tar -zcf $filename $dir_path 2>> $LOG_LOC

        #change ownership of backup files
        chown client:client $filename

        echo "Backing up of $dir_name completed." >> $LOG_LOC
    done < /backup_dirs.conf

    echo "Backup completed at :" >> $LOG_LOC
    date >> $LOG_LOC
```

Etablissement de la sauvegarde entre les machines

```
client@ubuntu:~$ rsync -av -e ssh /make_backup/ server@192.168.40.130:/home/client
The authenticity of host '192.168.40.130 (192.168.40.130)' can't be established.
ECDSA key fingerprint is SHA256:nIlQmPjpCcISr4TF4qgBGGeR80Dcis5sJbW0th5AOZQ.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.40.130' (ECDSA) to the list of known hosts.
server@192.168.40.130's password:
sending incremental file list
rsync: change_dir "/make_backup" failed: No such file or directory (2)

sent 20 bytes  received 12 bytes  1.49 bytes/sec
total size is 0  speedup is 0.00
rsync error: some files/attrs were not transferred (see previous errors) (code 23) at main.c(1207) [sender=3.1.3]
client@ubuntu:~$
```