

Compte rendu d'installation et de configuration des services

1. Objectifs

- Fournir un espace de stockage personnel aux utilisateurs.
- Restreindre l'accès au shell Bash aux membres du service "Administration Informatique".
- Automatiser la création de comptes utilisateurs à partir de listes fournies, avec des mots de passe générés automatiquement.
- Configurer et tester les services suivants sur un serveur Debian :
 - Cache DNS (Bind9)
 - Partage de fichiers SMB
 - Service web avec Nginx
 - Service TFTP

2. Préparation des comptes utilisateurs

2.1 Création manuelle des utilisateurs

Exemple pour un utilisateur :

```
useradd -m -g AdministrationInformatique -s /bin/bash -c "Bernard Léo" bernard
```

2.2 Définition des mots de passe

Utiliser pwgen pour générer des mots de passe alphanumériques de 8 caractères :

```
echo "bernard:$(pwgen -cnsB 8 2)" | chpasswd
```

2.3 Restrictions d'accès au shell

Assigner `/usr/sbin/nologin` comme shell pour les utilisateurs ne faisant pas partie du service "Administration Informatique" :

```
usermod -s /usr/sbin/nologin boyer
```

3. Configuration des services

3.1 Cache DNS (Bind9)

Installation de Bind9

```
apt update
apt install bind9 bind9-utils
```

Configuration de Bind9

Modifier `/etc/bind/named.conf.options` :

```
options {
    directory "/var/cache/bind";
    forwarders { 8.8.8.8; };
    allow-query { 192.168.0.0/24; };
    recursion yes;
    dnssec-validation auto;
};
```

Redémarrer Bind9 :

```
sudo systemctl restart bind9
```

Test du service Utiliser dig pour tester :

```
dig @localhost google.com
```

3.2 Partage de fichiers SMB

Installation de Samba

```
apt update
apt install samba
```

Configuration de Samba Modifier `/etc/samba/smb.conf` :

```
[global]
workgroup = WORKGROUP
server string = Serveur Samba
netbios name = debian
security = user
map to guest = bad user
dns proxy = no

[Public]
path = /srv/samba/Public
```

```
browseable = yes
read only = yes
guest ok = yes
create mask = 0664
directory mask = 0775

[Private]
path = /srv/samba/Private
browseable = yes
read only = no
guest ok = no
valid users = @Administratif, @AdministrationInformatique
create mask = 0660
directory mask = 0770
```

Créer les dossiers nécessaires :

```
mkdir -p /srv/samba/Public
mkdir -p /srv/samba/Private
chmod -R 0770 /srv/samba/Private
chmod -R 0755 /srv/samba/Public
```

Ajout des utilisateurs à Samba

```
smbpasswd -a bernard
```

Redémarrer Samba

```
systemctl restart smbd
```

Tester l'accès

```
smbclient //192.168.111.20/public -U bernard
```

3.3 Service Web avec Nginx

Installation de Nginx

```
apt update
apt install nginx
```

Configuration du site web

Créer un fichier de configuration dans `/etc/nginx/sites-available/mon_site` :

```
server {  
    listen 80 default_server;  
    listen [::]:80 default_server;  
  
    root /var/www/mon_site;  
    index index.html index.htm;  
  
    server_name _;  
  
    location / {  
        try_files $uri $uri/ =404;  
    }  
}
```

Activer la configuration :

```
sudo ln -s /etc/nginx/sites-available/mon_site /etc/nginx/sites-enabled/
```

Déployer le site Importer les fichiers depuis le dépôt Git :

```
git clone https://github.com/McMt27/SAE14_projet_final.git /var/www/mon_site
```

Redémarrer Nginx

```
sudo systemctl restart nginx
```

Tester le site

```
curl http://localhost
```

3.4 Service TFTP

Installation de TFTP

```
sudo apt update  
sudo apt install tftpd-hpa
```

Configuration du service

Modifier `/etc/default/tftpd-hpa` :

```
TFTP_USERNAME="tftp"  
TFTP_DIRECTORY="/srv/tftp"  
TFTP_ADDRESS="0.0.0.0:69"  
TFTP_OPTIONS="--secure"
```

Créer le dossier et ajuster les permissions :

```
mkdir -p /srv/tftp  
chmod -R 0777 /srv/tftp  
  
Redémarrer TFTP  
  
sudo systemctl restart tftpd-hpa  
  
Tester TFTP  
Utiliser :  
  
tftp 192.168.111.20
```

4. Problèmes rencontrés

Création des utilisateurs : La commande `newusers` ne fonctionnait pas correctement, ce qui a nécessité une création manuelle.

Conflit Apache2/Nginx : Apache2 occupait déjà le port 80, ce qui a retardé la configuration.

Installation hors-ligne : Nous avons dû utiliser des archives pour installer certains packages, ce qui était nouveau pour moi et a causé un retard.

5. Tests finaux et validation des services

5.1 Cache DNS (Bind9)

Vérification de la résolution DNS pour un site internet :

```
dig @192.168.111.20 example.com
```

5.2 Partage de fichiers SMB

Accéder au partage public en tant qu'utilisateur invité :

```
smbclient //192.168.111.20/Public -U bernard
```

Vérifier également l'accès au partage privé avec authentification :

```
smbclient //192.168.111.20/Private -U bernard
```

5.3 Service Web avec Nginx

Ouvrir le site sur un navigateur ou utiliser curl :

```
curl http://192.168.111.20
```

5.4 Service TFTP

Charger un fichier de test sur le serveur TFTP :

```
echo "test file" > test.txt
tftp 192.168.111.20
tftp> put test.txt
tftp> get test.txt
```

6. Résumé des configurations

Service	Statut	Remarques
Cache DNS	<input checked="" type="checkbox"/> Fonctionnel	Résolution des noms réussie pour les adresses testées.
Partage SMB	<input checked="" type="checkbox"/> Fonctionnel	Accès conforme aux droits d'utilisateur configurés.
Service Web	<input checked="" type="checkbox"/> Fonctionnel	Site web accessible via navigateur et outil de ligne de commande.
Service TFTP	<input checked="" type="checkbox"/> Fonctionnel	Transferts de fichiers confirmés sans erreur.