

Documentation : Attribution d'IP fixes via DHCP avec ISC sous Linux

Documentation de Valentin BAILLEUX

1. Introduction

La Maison des Ligues de Lorraine (M2L) nécessite une gestion efficace des adresses IP pour ses postes. Certaines applications requièrent des adresses IP fixes, mais la configuration manuelle de chaque machine est difficilement gérable.

La solution retenue consiste à utiliser un serveur **ISC DHCP** sous Linux, configuré pour attribuer des adresses IP statiques en fonction des adresses MAC des machines clientes. Une automatisation permet de récupérer les associations MAC/Nom d'hôte et de mettre à jour la configuration du serveur DHCP dynamiquement.

2. Présentation de la solution

2.1 Objectifs

- Assurer que chaque poste reçoive la même adresse IP via DHCP.
- Automatiser la récupération des adresses MAC et des noms d'hôte.
- Générer dynamiquement les réservations DHCP.

2.2 Technologies utilisées

- **Système d'exploitation** : Linux (Debian/Ubuntu)
 - **Serveur DHCP** : ISC DHCP Server
 - **Outils de scan réseau** : `nmap` , `arp-scan`
 - **Langage de script** : Bash / Python
 - **Base de données** : MySQL / SQLite (optionnel)
-

3. Installation et Configuration du serveur DHCP

3.1 Installation d'ISC DHCP

```
sudo apt update
sudo apt install isc-dhcp-server -y
```

3.2 Configuration du fichier DHCP

Modifier le fichier `/etc/dhcp/dhcpd.conf` :

```
default-lease-time 600;
max-lease-time 7200;
authoritative;

subnet 192.168.1.0 netmask 255.255.255.0 {
    range 192.168.1.100 192.168.1.200;
    option routers 192.168.1.1;
    option domain-name-servers 8.8.8.8, 8.8.4.4;
}
```

3.3 Configuration des baux statiques (exemple manuel)

```
host poste1 {
    hardware ethernet AA:BB:CC:DD:EE:FF;
    fixed-address 192.168.1.10;
}
```

4. Automatisation de l'ajout des réservations DHCP

4.1 Scan du réseau pour récupérer les adresses MAC

Utilisation de `nmap` pour détecter les machines actives :

```
sudo nmap -sn 192.168.1.0/24 | grep MAC
```

4.2 Génération automatique des réservations DHCP

Exemple de script en Python :

```
import os

def generate_dhcp_config(mac_ip_list, output_file="/etc/dhcp/dhcpd.conf"):
    with open(output_file, "a") as f:
        for mac, ip in mac_ip_list.items():
            f.write(f"host client_{ip.replace('.', '_')} {{\n")
            f.write(f"    hardware ethernet {mac};\n")
            f.write(f"    fixed-address {ip};\n")
            f.write("}\n\n")
    os.system("systemctl restart isc-dhcp-server")
```

4.3 Ajout automatique à la configuration DHCP

Un script peut être exécuté périodiquement via un cron job pour maintenir à jour la liste des baux statiques.

5. Déploiement et tests

5.1 Vérification du service DHCP

```
sudo systemctl status isc-dhcp-server
```

5.2 Test d'attribution IP

Sur un client Linux :

```
ip a | grep eth0
```

Ou relancer la requête DHCP :

```
sudo dhclient -r && sudo dhclient
```

5.3 Logs et diagnostics

Consulter les logs DHCP pour vérifier les attributions :

```
tail -f /var/log/syslog | grep dhcpd
```

6. Conclusion

Cette solution permet d'attribuer automatiquement des adresses IP fixes aux machines en fonction de leur adresse MAC, simplifiant ainsi la gestion réseau de la M2L. L'automatisation via des scripts Python et Bash assure une mise à jour dynamique des configurations DHCP.

Améliorations possibles :

- Intégration avec une base de données centralisée.
- Interface web pour gérer les baux statiques.
- Sécurisation du DHCP avec des VLANs ou des ACLs.