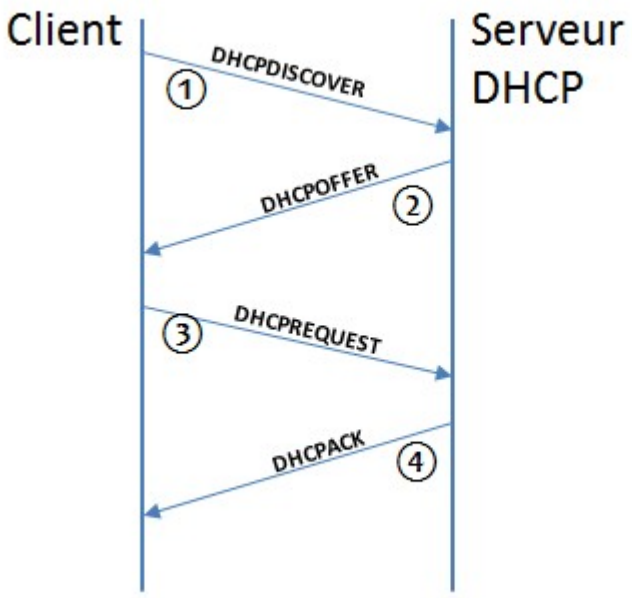
Dans un premier temps je me rappelle les trames DHCP , les échanges DHCP sont les suivantes

```
1179 42.735773012  0.0.0.0          255.255.255.255    DHCP     373 DHCP Discover - Transaction ID 0x68edfc09
1197 43.741742571  10.202.255.254   255.255.255.255    DHCP     342 DHCP Offer    - Transaction ID 0x68edfc09
1198 43.742864655  0.0.0.0          255.255.255.255    DHCP     379 DHCP Request  - Transaction ID 0x68edfc09
1199 43.751447045  10.202.255.254   255.255.255.255    DHCP     342 DHCP ACK      - Transaction ID 0x68edfc09
```

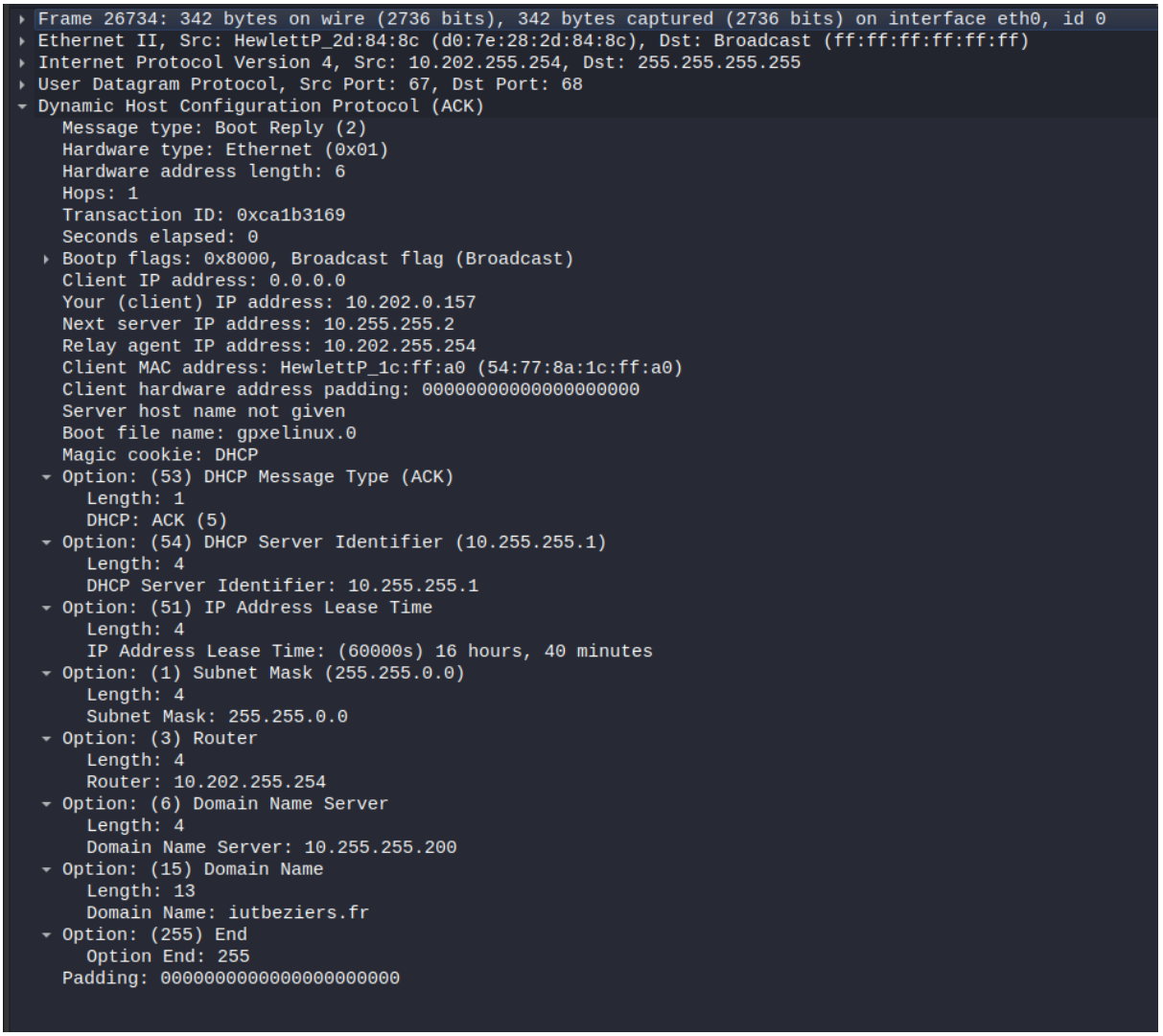
On émet un DHCP Discover en broadcast, le serveur DHCP répond avec une DHCP offer (ip potentielles), On demande à avoir cette “offre” avec un DHCP Request et le DHCP ACK fait de l'accusé la réception en fixant l’adresse IP et son masque de sous-réseau au client ainsi que la durée du bail de cette adresse.

On a au total 4 trames à chaque fois si tout se passe bien.

**Illustration :**



On peut observer tous les informations relatives au baux dhcp depuis la trame DHCP ACK :




À récupérer depuis la TRAME ACK, mac de destination, address ip , mask, address route par défaut, domain name, ip dns

Après quelque recherche sur la façon de récupérer les trames avec python,
je suis tomber sur scapy, il est très complet et il correspond bien à ce qu'on cherche.

La documentation scapy :

https://scapy.readthedocs.io/en/latest/index.html

Je prend le reste de mon temps à lire la documentation pour cette journée en préparation pour la prochaine

**Bout de code exemple utilisable trouver sur internet (code à retravailler/façonner) :**

```
from scapy.all import sniff, Ether, IP, UDP, DHCP

# Créer une fonction de traitement des paquets DHCP
def dhcp_packet_handler(packet):
	if DHCP in packet:
	print("DHCP Packet Received:")
	print("Source IP:", packet[IP].src)
	print("Destination IP:", packet[IP].dst)
	print("Source MAC:", packet[Ether].src)
	print("Destination MAC:", packet[Ether].dst)
	print("Message Type:", packet[DHCP].options[0][1])  # Assuming the first option is the DHCP message type
	print("Transaction ID:", packet[DHCP].xid)
	print("-----")

# Filtrer et capturer les trames DHCP
sniff(filter="udp and (port 67 or port 68)", prn=dhcp_packet_handler)
```