**ACADEMIA TEHNICĂ MILITARĂ “FERDINAND I”**

**FACULTATEA DE SISTEME INFORMATICE ŞI SECURITATE CIBERNETICĂ**

**Specializarea: Calculatoare şi sisteme informatice pentru apărare şi securitate naţională**

A yellow eagle with a crown and a shield

Description automatically generated

Proiectarea Sistemelor de Operare

Server DHCP

Huțuleac Andreea

Stancu David

C-113A

Cuprins

[1. Introducere 3](#_Toc181182653)

[2. Structura Serverului DHCP 4](#_Toc181182654)

[2.1. Functionalitati de baza 4](#_Toc181182655)

[3. Fisier de configurare 4](#_Toc181182656)

[4. Initializarea Serverului DHCP 5](#_Toc181182657)

[4.1. Configurarea Socketului UDP 5](#_Toc181182658)

[4.2. Asocierea Socketului la Portul UDP 5](#_Toc181182659)

[5. Gestionarea mesajelor DHCP 5](#_Toc181182660)

[6. Managementul Lease-urilor 8](#_Toc181182661)

[6.1. Alocarea adreselor IP 9](#_Toc181182662)

[6.2. Reinnoirea Lease-urilor 9](#_Toc181182663)

[6.3. Expirarea Lease-urilor 9](#_Toc181182664)

[7. Logging si Monitorizare 9](#_Toc181182665)

[8. Tratarea erorilor 9](#_Toc181182666)

[9. Concurenta si Multi-threading 9](#_Toc181182667)

[10. Testare si depanare 10](#_Toc181182668)

[11. Concepte de sisteme de operare 10](#_Toc181182669)

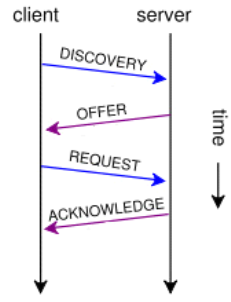
[12. Referinte si bibliografie 11](#_Toc181182670)

# Introducere

Serverul DHCP, avand la baza protocolul DHCP (Dynamic Host Configuration Protocol), automatizeaza procesul de alocare a adreselor IP si a altor parametrii de retea pentru dispozitive dintr-o retea locala. Serverul raspunde cererilor dispozitivelor (clientilor) pentru configurarea retelei, utilizand urmatoarele mesaje:

1. **DHCPDISCOVER**
2. **DHCPOFFER**
3. **DHCPREQUEST**
4. **DHCPDECLINE**
5. **DHCPACK**
6. **DHCPNAK**
7. **DHCPRELEASE**
8. **DHCPINFORM**

Mod de operare al protocolului DHCP cuprinde 4 faze distincte cu 4 tipuri de mesaje schimbate intre client si server:



1. Clientul trimite un mesaj **DHCPDISVCOVER** cu adresa MAC a clientului

* pentru a gasi un server DHCP care sa ii ofere o adresa IP
* mesaj broadcast la nivelul retelei: 255.255.255.255 pe portul UDP 67(server); clientul pe UDP 68

2. Serverul raspunde cu un mesaj **DHCPOFFER**

* mesajul contine o oferta pentru adresa IP, timpul de LEASE (perioada pentru care adresa IP este rezervata), id-ul serverului DHCP care trimite oferta + alte informatii de configurare
* serverul aloca temporar o adresa IP din pool-ul sau isi pastreaza alocarea pana cand clientul confirma oferta
* mesajul este transmis ca broadcast pentru a fi vazut de celelalte dispozitive, desi acesta este destinat doar clientului care a trimis DISCOVER(adresa MAC)

3. Clientul trimite un **DHCPREQUEST** cu adresa IP solicitata de la server, id-ul serverului DHCP care a facut oferta

* dupa ce primeste mesajul OFFER, clientul trimite un mesaj pentru a cere oficial una dintre ofertele primite
* mesaj tot broadcast pentru ca celelalte servere sa cunoasca faptul ca nu au fost selectate

4. Serverul trimite **DHCPACK** cu IP confirmat, durata LEASE + alte informatii pentru configurare

* + dupa ce primeste REQUEST, serverul verifica daca adresa IP solicitata este disponibila
  + in caz afirmativ, se raspunde cu un mesaj ACK ca adresa IP a fost alocata cu succes clientului

Mesaje optionale:

5. Clientul trimite un mesaj de tip **DHCPDECLINE** catre server indicand ca adresa oferita de acesta este deja folosita

6. Serverul trimite catre client un mesaj de tipul **DHCPNAK**, in cazul in care ACK este negativ

7. Clientul trimite catre server un mesaj de tip **DHCPRELEASE** prin care acesta specifica serverului ca doreste eliberarea parametrilor de retea primiti anterior de la acesta

8. Clientul care are deja o adresa IP alocata trimite un mesaj de tipul **DHCPINFORM** serverului prin care ii solicita ceilalti parametrii de retea.

# Structura Serverului DHCP

Serverul DHCP functioneaza prin crearea si gestionarea unui pool de adrese IP, din care aloca adrese dispozitivelor care solicita acest lucru. Se utilizeaza un fisier de lease-uri pentru a pastra asocierile **MAC-IP** si durata **lease-ului** pentru fiecare client conectat.

## Functionalitati de baza:

* 1. **Alocarea IP dintr-un pool de adrese**: Cand un client solicita o adresa IP, serverul trebuie sa ofere o adresa disponibila.
  2. **Reinnoirea lease-urilor**: După expirarea unui lease, serverul trebuie sa poata reinnoi adresa IP sau sa ofere una noua.
  3. **Reasocierea adresei IP**: Daca un client cu un MAC specific solicita din nou o adresa IP, serverul poate sa-i reasocieze aceleasi IP, daca este disponibil.
  4. **Timpul de expirare a lease-urilor**: Adresele IP trebuie sa fie eliberate automat dupa expirarea lease-ului daca clientul nu mai solicită o reinnoire.

# Fisier de configurare

Se utilizeaza un fisier de configurare pentru a defini:

* Intervalele de adrese IP disponibile pentru alocare.
* Asocieri MAC-IP statice.
* Parametrii de retea: durata lease-ului, gateway-ul, DNS-ul.

**Functia de citire a fisierului de configurare** este implementata pentru a interpreta continutul fisierului. Formatul utilizat este de tip „cheie-valoare” pentru a facilita parsarea si utilizarea acestor informatii de catre server.

**Exemplu structura fisier de configurare:**

# Configuratie server DHCP

#Optiuni generale

option domain-name "localdomain";

option domain-name-servers 8.8.8.8, 8.8.4.4;

#Timpi de lease

default-lease-time 600;

max-lease-time 7200;

# ddns-update-style none;

# Subnet si setari IP

subnet 192.168.1.0 netmask 255.255.255.0 {

range 192.168.1.10 192.168.1.20;

option routers 192.168.1.1;

option subnet-mask 255.255.255.0 ;

option broadcast-address 192.168.1.255;

}

authoritative;

# Binding PENDING time

#pending-time 30;

# Interfata de retea pentru serverul DHCP

INTERFACES="enp0s3";

# Initializarea Serverului DHCP

## Configurarea Socketului UDP

Serverul creeaza un socket UDP pe portul 67, folosit doar de serverele DHCP pentru a primi pachete broadcast de la clienti. Setarile socketului permit trimiterea de pachete broadcast catre toate dispozitivele din retea, asigurand ca toate cererile clientului vor fi procesate.

## Asocierea Socketului la Portul UDP

Functia **bind()** asociaza socketul creat la portul 67 al serverul DHCP astfel incat sa poata primi mesajele DHCP de la clienti. Aceasta permite monitorizarea continua a portului 67 pentru cereri noi de configurare DHCP.

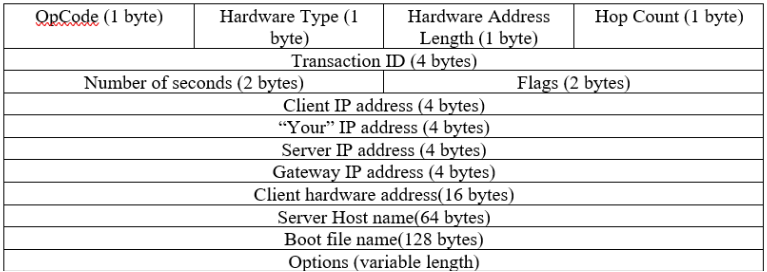
# Gestionarea mesajelor DHCP

Serverul interpreteaza pachetele DHCP primite de la clienti si actioneaza corespunzator in functie de tipul mesajului:

* **send\_dhcp\_offer():** trimite o oferta de adresa IP atunci cand primeste un mesaj DHCPDISCOVER
* **send\_dhcp\_ack()**: confirma alocarea adresei IP la primirea unui mesaj DHCPREQUEST de la client

Implementarea functiilor de mai sus are rolul de a asigura ca fiecare client primeste o adresa IP unica.

**Formatul pachetelor DHCP:**



**Structura pachetelor implementate in cod C:**

struct dhcp\_packet {

uint8\_t op;

uint8\_t htype;

uint8\_t hlen;

uint8\_t hops;

uint32\_t xid;

uint16\_t secs;

uint16\_t flags;

uint32\_t ciaddr;

uint32\_t yiaddr;

uint32\_t siaddr;

uint32\_t giaddr;

uint8\_t chaddr[16];

uint8\_t sname[64];

uint8\_t file[128];

uint8\_t options[312];

};

**Descrierea campurilor:**

* op: Codul de operatiune.

1: BOOTREQUEST (folosit de client) 2: BOOTREPLY (folosit de server)

* htype: Tipul de hardware.

Pentru Ethernet (10Mb), valoarea este 1.

* hlen: Lungimea adresei hardware (MAC address).

Pentru Ethernet, lungimea este 6.

* hops: Numarul de hopuri, folosit de agenti de retransmitere (relays) pentru a urmari de cate ori un pachet a trecut printr-un agent.
* xid: ID-ul tranzactiei, un identificator unic utilizat pentru a lega cererile si raspunsurile intre client si server.
* secs: Numarul de secunde de la initierea cererii clientului pana la trimiterea pachetului. Este utilizat pentru a prioritiza cererile vechi.
* flags:

Primul bit: Broadcast Flag. Daca este 1, clientul cere ca raspunsul sa fie difuzat.

Restul sunt rezervate si nefolosite.

* ciaddr: Client IP address.

Utilizat daca clientul are deja o adresa IP si cere o reinnoire a lease-ului. Daca nu are o adresa, este setat la 0.0.0.0.

* yiaddr: 'Your' IP address.

Este adresa IP oferita de server catre client. Serverul completeaza acest camp in mesajul DHCPOFFER.

* siaddr: Server IP address.

Serverul poate completa acest camp cu adresa sa IP.

* giaddr: Gateway IP address.

Utilizat de agenti de retransmitere pentru a transmite cererea la alte servere DHCP.

* chaddr: Client hardware address.

Adresa MAC a clientului care trimite cererea. Serverul foloseste acest camp pentru a identifica clientul.

* sname: Server name.

Nume optional al serverului care raspunde.

* file: Boot file name.

Numele fisierului pe care clientul il va descarca daca foloseste BOOTP pentru bootare.

* options: Campul optiuni este cel mai variabil si contine informatiile specifice despre tipul mesajului DHCP si alti parametri de retea (de exemplu, gateway, DNS, durata lease-ului, tipul mesajului DHCP etc.).

**Detalii despre campul options:**

Acest camp este folosit pentru extensii DHCP si contine diferite optiuni. Structura tipica a unei optiuni DHCP include:

* Codul optiunii: Un octet care identifica tipul optiunii.
* Lungimea optiunii: Un octet care indica lungimea valorii optiunii.
* Valoarea optiunii: Octetii care contin valoarea propriu-zisa a optiunii.
* Exemple de optiuni comune:
  + Option 53: Tipul mesajului DHCP (un octet care indică tipul mesajului: DHCPDISCOVER, DHCPOFFER, DHCPREQUEST, DHCPACK etc.).

1 = DHCPDISCOVER

2 = DHCPOFFER

3 = DHCPREQUEST

5 = DHCPACK

* Option 54: IP-ul serverului DHCP.
* Option 1: Mască de subrețea (Subnet Mask).
* Option 3: Gateway implicit (Router).
* Option 6: Servere DNS.

**Exemplu de pachet DHCPDISCOVER:**

Pentru a construi un pachet DHCPDISCOVER, trebuie completate campurile astfel:

op: 1 (BOOTREQUEST)

htype: 1 (Ethernet)

hlen: 6 (lungimea adresei MAC)

hops: 0 (nu folosești agenti de retransmitere in acest caz)

xid: un numar aleator care identifica sesiunea

secs: 0 sau numarul de secunde trecute de la inceputul cererii

flags: 0 sau 1 pentru broadcast

ciaddr: 0.0.0.0 (clientul nu are încă o adresă IP)

yiaddr, siaddr, giaddr: 0.0.0.0

chaddr: adresa MAC a clientului

sname si file: campuri lasate goale (zero).

options:

Option 53 (DHCP Message Type): 1 (DHCPDISCOVER)

Option 255 (End option)

Ordinea campurilor in pachetul DHCP:

DHCP Header (primele 236 octeti, fara campul options)

DHCP Options

# 

# Managementul Lease-urilor

Lease-urile includ informatii precum:

1. Adresa MAC a dispozitivului client
2. Adresa IP alocata
3. Timpul de expirare a lease-ului

## Alocarea adreselor IP

Functia de alocare a adreselor IP verifica pool-ul disponibil si atribuie o adresa clientului, mentinand o evidenta a duratei lease-urilor. Coliziunile trebuie evitate prin verificarea fiecarei adrese IP inainte de alocare, astfel incat fiecare client sa primeasca un IP unic.

## Reinnoirea Lease-urilor

Functia de reinnoire permite extinderea lease-ului pentru adresa IP deja atribuita unui client, atunci cand acesta solicita o reinnoire. Aceasta este activata la jumatatea perioadei lease-ului si asigura ca dispozitivul client poate pastra aceeasi adresa IP atata timp cat este activ.

## Expirarea Lease-urilor

Dupa expirarea lease-ului, serverul elibereaza adresa IP pentru a o reutiliza. Aceasta functie verifica periodic duratele lease-urilor si elibereaza adresele neutilizate, astfel incat sa fie disponibile pentru clientii noi.

# Logging si Monitorizare

Serverul genereaza un fisier de log pentru a inregistra toate evenimentele DHCP, incluzand:

* Alocarile si reinnoirile de IP
* Expirarea lease-urilor
* Erorile de alocare si alte probleme de retea.

Fisierul de log este utilizat pentru monitorizarea starii serverului si diagnosticarea eventualelor probleme.

# Tratarea erorilor

Serverul gestioneaza erorile de tipul: imposibilitatea deschiderii unui socket sau de a aloca o adresa IP, prin logarea acestora si transmiterea unor mesaje de eroare clare.

# Concurenta si Multi-threading

Pentru a permite procesarea mai multor cereri simultan, serverul poate utiliza multi-threading sau multi-processing. Fiecare cerere DHCP poate declansa un nou thread sau un proces separat, care gestioneaza cererea specifica fara a bloca alte cereri.

Thread-urile partajeaza aceeasi memorie, ceea ce inseamna ca toate firele de executie pot accesa aceeasi baza de date sau fisiere de configurare, fara a fi necesare mecanisme complicate de comunicare intre procese.

Din punct de vedere al utilizarii resurselor, multi-threading este mai eficient decat multi-processing, deoarece firele de executie folosesc mai putina memorie si pot impartasi datele mai rapid intre ele.

**Principiu de functionare in practica:**

* **Fire de executie pentru cererile clientilor**: Fiecare cerere DHCP de la un client (DHCPDISCOVER) poate declansa crearea unui nou thread care se ocupa de acea cerere. Acest thread va prelua cererea, va fisierul de configurare cu lease-uri si va raspunde cu un DHCPOFFER.
* **Sincronizare**: Daca mai multe fire executie acceaseaza aceleasi fisiere (lease-uri sau configurari), este nevoie de mecanisme de sincronizare (de exemplu, mutex-uri) pentru a preveni accesul concurent care poate duce la coruperea datelor.

**Pasi de implementare multi-threading pentru serverul DHCP:**

1. **Crearea unui thread pentru fiecare cerere**:
   1. La fiecare cerere DHCP primita (de exemplu, un DHCPDISCOVER), serverul poate crea un thread nou care va trata aceasta cerere.
   2. Dupa ce cererea este procesata si clientul a primit un raspuns (DHCPOFFER și DHCPACK), thread-ul respectiv poate fi terminat.
2. **Sincronizarea accesului la lease-uri si configurari**:
   1. Daca se foloseste un fisier de lease-uri, trebuie implementate **mutex-uri** sau alte mecanisme de sincronizare pentru a evita ca mai multe thread-uri sa modifice lease-urile in acelasi timp.

Pentru implementarea multi-threading se poate folosi biblioteca **pthread** in C sau echivalentul în C++ pentru a crea fire de execuție care gestioneaza cererile DHCP.

# Testare si depanare

Serverul se va testa intr-o retea locala pentru a se asigura ca alocarea adreselor IP si gestionarea lease-urilor functioneaza corect. Scenariile de test presupun:

* Cererile initiale de IP
* Reinnoirea lease-urilor
* Gestionarea expirarii lease-urilor si eliberarea adreselor IP
* Trimiterea corecta a mesajelor DHCP la fiecare etapa a procesului de alocare

# Concepte de sisteme de operare

* 1. **Multi-threading sau Multi-processing**: Serverul va trebui sa fie capabil sa gestioneze mai multe cereri DHCP simultan, fie folosind fire de executie (threads) sau procese separate
  2. **Gestionarea fisierelor**: Implementarea mecanismelor de citire si scriere in fisierele de configurare si lease-uri.
  3. **Sincronizare**: Utilizarea multi-threading → implementarea mecanismelor de sincronizare pentru accesul concurent la lease-uri si fisierele de configurare.

# Referinte si bibliografie

1. **RFC 2131 - DHCP** (Documentul oficial pentru DHCP):
   1. RFC 2131 - Dynamic Host Configuration Protocol
2. **Programarea cu socket-uri în C/C++**:
   1. Beej's Guide to Network Programming - Un ghid pentru socket-uri în C.
3. **Concurența în C (pthread)**:
   1. pthread Documentation - Documentatia pentru fire de executie (threads) în C.
4. **Utilizarea mutex-urilor si conditiilor de sincronizare**:
   1. POSIX Threads Programming
5. **Exemple de servere DHCP** (pentru a vedea cum sunt construite):
   1. DHCP Server on Linux