**Client DHCP**

Zară Cătălin-Ionuț

Toșcariu Alberto-Ionuț

**CUPRINS**

1. **BREVIER TEORETIC**
2. **Introducere...................................................................page 2**
3. **De ce folosim DHCP? .................................page 2**
4. **Cum a aparut DHCP? ................................page 3**
5. **Functionalitate.............................................................page 4**
6. **Alocarea adresei IP......................................page 4**
7. **.1 Alocare Automată.....................................page 4**
8. **.2 Alocare Dinamică. Leasing......................page 4**
9. **.3 Alocare Manuală.......................................page 5**
10. **Mesaje...........................................................page 5**
11. **.1 Modelul Server/Client..............................page 6**
12. **.2 Formatul Mesajelor..................................page 8**
13. **Opțiuni..........................................................page 9**
14. **CONCEPT APLICATIE**
    * + 1. **Introducere...............................................page 11**
        2. **Modelul State Machine............................page 11**
        3. **Configurare Pachetelor...........................page 12**
        4. **User Interface...........................................page 12**
        5. **Clientul......................................................page 13**
15. **BIBILIOGRAGIE ...........................................................page 13**

## BREVIER TEORETIC

## 1. Introducere

Dynamic Host Configuration Protocol (Protocol Configurare Dinamică Clienți) sau pe scurt DHCP este un protocol de management de rețea folosit de servere pentru a aloca adrese IP automat calculatoarelor și dispozitivelor conectate la ele. Acesta furnizează parametrii de configurare pentru gazdele Internet într-un model client-server. Gazdele serverului DHCP alocă adresele de rețea și furnizează parametrii de configurare altor gazde (client).

DHCP este descris în RFC-urile 2131 și 2132. Este utilizat pe scară largă pe Internet pentru a configura tot felul de parametri pe lângă furnizarea gazdelor cu adrese IP. La fel ca și în rețelele de afaceri și de acasă, DHCP este utilizat de către furnizorii de servicii Internet pentru a seta parametrii dispozitivelor prin legătura de acces la Internet, astfel încât clienții să nu fie nevoiți să-și telefoneze ISP-urile (furnizorii de servicii) pentru a obține aceste informații. Exemple obișnuite de informații care sunt configurate includ masca de rețea, adresa IP a gateway-ului implicit și adresele IP ale serverelor DNS și de timp. DHCP a înlocuit în mare măsură protocoalele anterioare (numite RARP și BOOTP) cu funcționalități mai limitate.

### 1.1 De ce folosim DHCP?

De ce este nevoie de atat de multe protocoale când cu siguranță putem sa ne ocupăm și singuri de gestionarea unor adrese IP? Păi...

* Administratorul de rețea necesită multă muncă manuală, consumând mult timp și e predispus la erori.
* Menținerea parametrilor actualizați nu este un efort unic. Cantitatea de muncă crește odată cu modificările din rețea (de exemplu, computerele portabile care schimbă locația introduc frecvent o mulțime de muncă zilnică pentru administratorii de rețea).
* O modificare a unui parametru comun tuturor computerelor dintr-o subrețea (de exemplu: adresa routerului local) forțează modificările în fiecare computer de pe net.
* Este posibil ca unele sisteme să nu aibă un dispozitiv de stocare permanent (de exemplu, hard disk) pentru a stoca parametrii de configurare - caz în care nu poate fi luată în considerare nicio configurație locală.
* În caz de lipsă de adrese IP și o rețea care se schimbă frecvent, poate fi o cale de a oferi unui computer (care poate fi în afara rețelei pentru o perioadă) o adresă permanentă. O abordare mai bună va fi utilizarea unui grup comun de adrese de către un set de computere. Configurarea manuală nu oferă o modalitate ușoară de a face acest lucru.

Toate aceste motive duc la necesitatea unui mecanism automat pentru configurarea protocoalelor TCP / IP, iar DHCP este cel mai avansat mecanism în acest moment.

### 1.2 Cum a apărut DHCP?

La început, cele mai multe rețele TCP/IP erau mici și statice, iar setarea manuală de adrese IP nu era prea dificilă, fiecare stație reținându-se propria adresa undeva în memoria secundară. Odată ce o adresă trebuia sa fie schimbată aceasta necesita acțiunea manuală a unui administrator de rețea de obicei direct de la consolă ceea ce implica și un reboot in cele mai multe cazuri.

Puțin după, cu cat rețele tot mai sofisticate începeau sa se stabilească, și începeau să apară și clienții care nu mai utilizează memorie secundară pentru a reține adrese IP, nevoia de administrare centrală a hardware-ului la legăturile de adrese IP a devenit evidentă.

Astfel a fost conceput un protocol special (**RARP**) pentru astfel de legături. A permis unei mașini dintr-un segment de rețea să învețe propria adresă IP și apoi să înceapă operarea normală TCP / IP.

Un alt protocol, **BOOTP**, a fost, de asemenea, dezvoltat pentru a permite stațiilor fără disc să recupereze toți parametrii de configurare TCP / IP și alte date ale sistemului de operare, necesare pentru a începe să funcționeze normal după o pornire. A permis configurarea pe rețele mai largi, deoarece nu a fost limitată la un singur segment. BOOTP definește un concept de agent de releu BOOTP care specifică modul în care traficul BOOTP este redirecționat între mai multe segmente.

Una dintre extensiile ce au urmat BOOTP-ului a fost protocolul DHCP. Există două diferențe principale între DHCP și BOOTP.

1. DHCP definește mecanismele prin care clienților li se poate atribui o adresă de rețea pentru o închiriere finită, permițând realocarea în serie a adreselor de rețea către diferiți clienți. (lease)
2. DHCP oferă mecanismul pentru ca un client să dobândească toți parametrii de configurare IP de care are nevoie pentru a funcționa.

DHCP se bazează pe BOOTP, adăugând capacitatea de alocare automată a adreselor de rețea reutilizabile și opțiuni de configurare suplimentare.

## 2. Funcționalitate

### 2.1 Alocarea adresei IP

DHCP acceptă trei mecanisme pentru alocarea adresei IP.

* Alocare automată - în care o adresă IP permanentă este atribuită clientului.
* Alocare dinamică - în care adresa este alocată pentru o perioadă limitată de timp (un „leasing”).
* Alocare manuală - în care adresa este atribuită manual de către administratorul de rețea.

### 2.1.1. Alocare Automată

Alocarea automată permite DHCP să ofere unei gazde o adresă de rețea permanentă, dar totuși o face automat, fără interferențe umane.

### 2.1.2. Alocare Dinamică.Leasing

O problemă care apare cu atribuirea automată a adreselor IP dintr-un pool este pentru cât timp ar trebui alocată o adresă IP. Dacă o gazdă părăsește rețeaua și nu returnează adresa IP către serverul DHCP, acea adresă se va pierde definitiv. După o perioadă de timp, se pot pierde multe adrese. Pentru a preveni acest lucru, atribuirea adresei IP poate fi pentru o perioadă determinată de timp, o tehnică numită leasing. Chiar înainte de expirarea contractului de închiriere, gazda trebuie să solicite reînnoirea DHCP. În cazul în care nu reușește să facă o cerere sau cererea este respinsă, gazda nu mai poate folosi adresa IP pe care a fost dată anterior

Mecanismul de bază pentru alocarea dinamică a adreselor de rețea este simplu: un client solicită utilizarea unei adrese pentru o anumită perioadă de timp (care se numește leasing). Mecanismul de alocare garantează nu realocarea acelei adrese în timpul solicitat și încearcă să returneze aceeași adresă de rețea de fiecare dată când clientul solicită o adresă.

Perioada de valabilitate a unei adrese este strâns legată de tipul de rețea, pentru rețele de tip Hotspot valabilitatea recomandată este de 8 ore, Wireless 24 de ore, iar pentru dispozitivele legate prin cablu de 8 zile. Clientul DHCP ar trebui să isi reînnoiască valabilitatea după ce a trecut jumătate din timp.

**Clientul poate:**

* Extinde contractul de închiriere cu cererile ulterioare.
* Soliciti o misiune permanentă cerând un contract de închiriere infinită.
* Emite un mesaj pentru a elibera adresa înapoi la server atunci când clientul nu mai are nevoie de adresă.

### 2.1.3. Alocare Manuală

Alocarea manuală permite DHCP să fie utilizat pentru a elimina procesul predispus la erori de configurare manuală a gazdelor cu adrese IP în medii în careeste de dorit să gestionați atribuirea adresei IP în afara mecanismelor DHCP. Sunt utilizate în mod obișnuit pentru servere și alte dispozitive care trebuie să aibă aceeași adresă IP tot timpul, deși există multe motive pentru a avea o adresă IP statică pe stația dvs. de lucru.

### 2.2 Mesaje

Când un dispozitiv se conectează la o rețea acesta folosește un client DHCP pentru a cere serverului DHCP un IP unic în cadrul acelei rețele împreună cu alte informații în funcție de nevoile gazdei.

Aceste date sunt obținute într-un număr finit de pași ce implică un schimb de mesaje între client și server cunoscut și sub numele de **ciclul DORA** (Discover, Offer, Request, Acknowledge) ce reprezinta abrevierea celor mai importante 4 mesaje prin care comunică clientul și serverul.

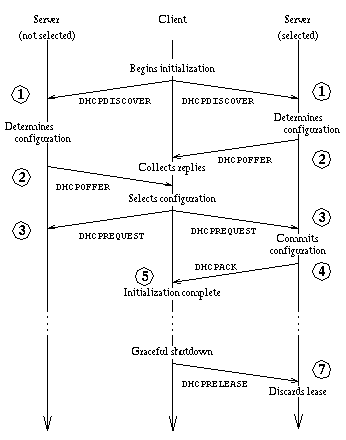
Cele 8 mesaje sunt:

1. **DHCPDISCOVER** – clientul trimite acest mesaj cu parametrii de configurare, către toate dispozitivele conectate la acea rețea pentru a identifica un server DHCP.
2. **DHCPOFFER** – serverul răspunde clientului cu propria adresa și parametrii de configurare pe care serverul îi poate oferi.
3. **DHCPREQUEST** –clientul cere serverului:
   1. solicitarea parametrilor oferiți de la un server și implicit refuzul ofertelor de la toți ceilalți;
   2. confirmarea corectitudinii adresei alocate anterior după, de exemplu, repornirea sistemului;
   3. extinderea contractului de leasing pe o anumită adresă de rețea.
4. **DHCPACK** - serverul confirma parametrii de configurare
5. **DHCPNAK** – serverul infirma parametrii de configurare (de exemplu, clientul s-a mutat într-o subrețea nouă sau contractul de închiriere a clientului expirat)
6. **DHCPDECLINE** – clientul semnalează că adresa de rețea este deja utilizată.
7. **DHCPRELEASE** – clientul comunica serverului renunțarea la adresa rețelei și anularea contractului de leasing rămas.
8. **DHCPINFORM** – clientul solicită numai parametrii de configurare locali; clientul are deja o adresă de rețea configurată extern.

### 2.2.1 Modelul Server/Client

Clientul și serverul negociază într-o serie de mesaje pentru ca clientul să obțină parametrii de care are nevoie.

Următoarea diagramă arată mesajele schimbate între clientul DHCP și servere la alocarea unei noi adrese de rețea. Următoarea este o explicație detaliată a tuturor mesajelor diferite și o descriere a etapelor de comunicare.



Ordinea mesajelor in majoritatea cazurilor de conexiune la o rețea este următoarea:

1. Clientul transmite un **DHCPDISCOVER**.
2. Fiecare server poate răspunde cu un mesaj **DHCPOFFER**.
3. Clientul primește unul sau mai multe mesaje **DHCPOFFER** de la unul sau mai multe servere și alege un server din care să solicite parametrii de configurare.
4. Clientul transmite un mesaj **DHCPREQUEST**.
5. Acele servere care nu sunt selectate de mesajul **DHCPREQUEST** folosesc mesajul ca notificare că clientul a refuzat oferta serverului respectiv.
6. Serverul selectat în mesajul **DHCPREQUEST** angajează răspunsurile cu un mesaj **DHCPACK** care conține parametrii de configurare pentru clientul solicitant.
7. Clientul primește mesajul **DHCPACK** cu parametrii de configurare. În acest moment, clientul este configurat.
8. Dacă clientul primește un mesaj **DHCPNAK**, clientul repornește procesul de configurare.
9. Clientul poate alege să renunțe la contractul de închiriere pe o adresă de rețea, trimițând un mesaj **DHCPRELEASE** către server (de exemplu, la închidere).
10. Serverul primește mesajul **DHCPRELEASE** și marchează contractul de închiriere ca fiind gratuit.

Există două variații principale ale scenariului de interacțiune client / server prezentat:

1. Reutilizarea unei adrese de rețea alocate anterior:

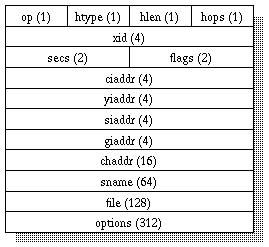
Dacă un client își amintește (în memoria cache) și dorește să reutilizeze o adresă de rețea alocată anterior, un client poate alege să omită unii dintre pașii luați în cazul unei noi alocări.În prima **DHCPREQUEST** clientul include adresa de rețea în opțiunea „adresă IP solicitată”. Serverul care are cunoștințele despre configurația clientului răspunde cu un mesaj **DHCPACK** și de atunci diagrama continuă de la pasul (5).

1. Obținerea parametrilor cu adresa de rețea configurată extern:

Dacă un client a obținut o adresă de rețea prin alte mijloace (de exemplu, configurare manuală), acesta poate utiliza un mesaj de solicitare **DHCPINFORM** pentru a obține alți

parametri de configurare locală. Serverele care primesc un mesaj **DHCPINFORM** construiesc un mesaj **DHCPACK** cu parametrii de configurare locali corespunzători pentru client, fără a aloca o adresă nouă.

### 2.2.2 Formatul Mesajelor



Numărul din paranteză indică mărimea fiecărui câmp, în octeți.

Descrierea fiecărui câmp:

* op(1) - Cod op mesaj / tip mesaj(1 = BOOTREQUEST, 2 = BOOTREPLY)
* htype(1) - Hardware address type (e.g., '1' = 10Mb Ethernet)
* hlen(1) - Hardware address length (e.g. '6' for 10Mb Ethernet)
* hops(1) - Clientul se setează la zero, utilizat opțional de agenții de releu atunci când pornește prin intermediul unui agent de releu.
* xid(4) - Transaction ID. Un număr aleatoriu ales de client, utilizat de client și server pentru a asocia mesajul de solicitare cu acesta și cu răspunsul.
* secs(2) - Secunde de când clientul a început procesul de Request. Completat de client.
* flags(2) - Flaguri.
* ciaddr(4) - Adresa IP a clientului. Completat de client dacă își cunoaște adresa IP (din cereri anterioare sau din configurații manuale)
* yiaddr(4) - Adresă IP. Răspunsul serverului la client.
* siaddr(4) - Adresa IP a serverului. Adresa serverului de trimitere sau a următorului server de utilizat în etapa următoare a procesului de bootstrap.
* giaddr(4) - Adresa IP a agentului de releu, utilizată la pornirea prin intermediul unui agent de releu.
* chaddr(16) - Adresa hardware a clientului.
* sname(64) - Numele de gazda al serverului, e opțional și se termină în null.
* file(128) -Numele fișierului de pornire. Șir terminat null; nume "generic" sau null în cerere, nume complet calificat al directorului în răspuns.
* options(var-312) - Aici va fi măcar o optiune.

### 2.3 Optiuni

Furnizarea de opțiuni DHCP este un mod inteligent de configurare a clienților de rețea în faza incipientă a implementării accesului la rețea. În plus față de furnizarea adresei IP, protocolul DHCP este capabil să seteze o grămadă mare de opțiuni care sunt foarte utile pentru configurarea dispozitivului. Fiecare opțiune are un nume și un identificator numeric pentru a fi transportate în cadrele de protocol.

Cele mai comune opțiuni DHCP schimbate între server și clienți sunt:

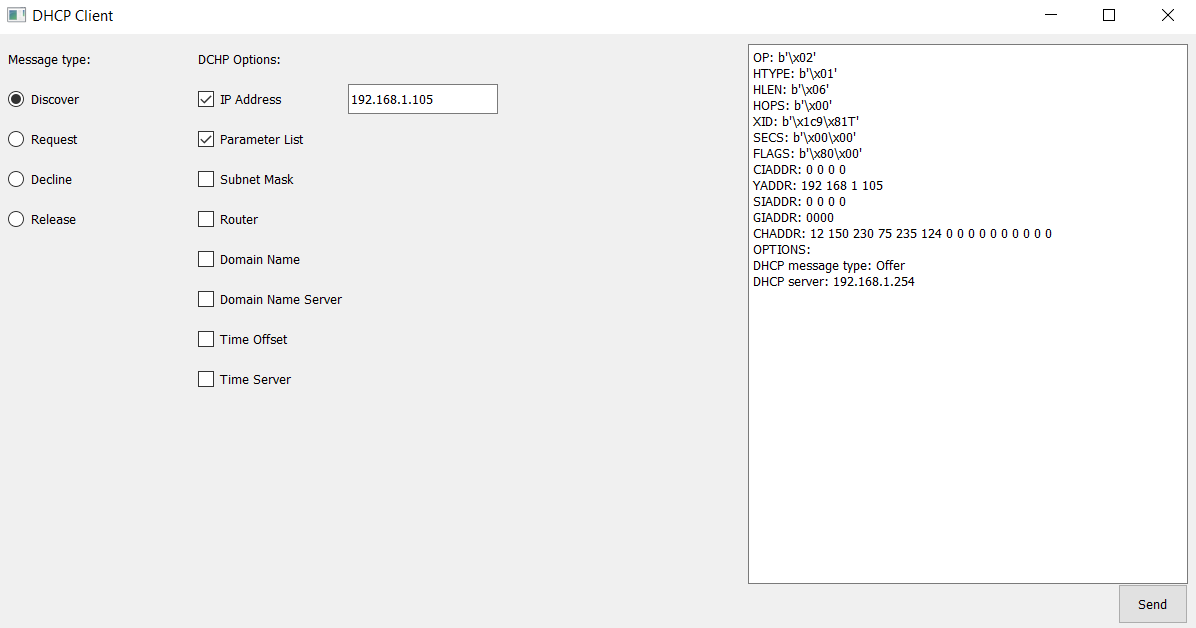
* Opțiunea DHCP 1: mască de subrețea care trebuie aplicată pe interfața care solicită o adresă IP
* Opțiunea DHCP 2: decalarea timpului în secunde de la UTC care trebuie aplicată la ora curentă (notă: depreciată de RFC4833 - opțiunile 100 și 101)
* Opțiunea DHCP 3: router implicit sau gateway de ultimă instanță pentru această interfață
* Opțiunea DHCP 4: lista serverului de timp, așa cum se menționează în RFC868 (Time Protocol)
* Opțiunea DHCP 6: ce DNS (Domain Name Server) să includă în configurația IP pentru rezoluția numelui
* Opțiunea DHCP 51: timpul de închiriere pentru această adresă IP
* Opțiunea DHCP 12: numele gazdei clientului, foarte util pentru IoT și orice dispozitiv fără utilizator
* Opțiunea DHCP 15: specifică numele de domeniu pe care clientul ar trebui să îl utilizeze ca sufix atunci când rezolvă numele gazdei prin intermediul sistemului de nume de domeniu
* Opțiunea DHCP 42: lista serverelor NTP după ordinea preferințelor, utilizată pentru sincronizarea orei a clientului
* Opțiunile DHCP 58 și 59: Valoarea timpului de reînnoire (T1) și Valoarea timpului de reîncărcare (T2). Consultați capitolul „Administrarea timpului de închiriere DHCP” despre Ce este DHCP?
* Opțiunile DHCP 69 și 70: respectiv pentru serverele SMTP și POP3 pentru trimiterea și primirea e-mailurilor. Aceste opțiuni le vedem adesea pe imprimante și scanere
* Opțiunea DHCP 81: Numele de domeniu complet calificat al clientului - această opțiune permite efectuarea actualizării automate a înregistrărilor DNS asociate clientului, în principal A și PTR. În opțiune putem specifica dacă clientul sau serverul vor actualiza înregistrările și FQDN-ul asociat clientului. Este definit în RFC4702
* Opțiunea DHCP 100: fus orar șir POSIX ca în IEEE 1003.1
* Opțiunea DHCP 101: fus orar ca șir ca în baza de date TZ (de exemplu: Europa / Paris)
* Opțiunea DHCP 119: listă de căutare a domeniului DNS care va fi utilizată pentru a efectua cereri DNS bazate pe numele scurt folosind sufixele furnizate în această listă.
* Opțiunea DHCP 121: tabel de rute statice fără clase compus din mai multe rețele și mască de subrețea, această opțiune înlocuiește cea originală numerotată 33 (vezi RFC3442)

1. **CONCEPT APLICAȚIE**
2. **Introducere**

Aplicația constă în construirea unui client DHCP, configurarea acestuia și comunicarea acestuia cu un server, în cazul de față fiind reprezentat de cel oferit de routerul personal, simulând transmiterea a mai multor tipuri de mesaje precum: **“Discover”, “Request”, “Decline”, “Release”**.

Aceasta este concepută să ruleze simultan pe 2 threaduri diferite, unul de comunicare a clientului cu serverul și unul destinat comunicării utilizatorului cu interfața grafică.

Aplicația pune la dispoziția utilizatorului o interfață grafică construită cu ajutorul librăriei **PyQT5** care facilitează selectarea tipului de mesaj pe care acesta dorește să îl transmită serverului și diferitele opțiuni pe care să le includă în pachet.

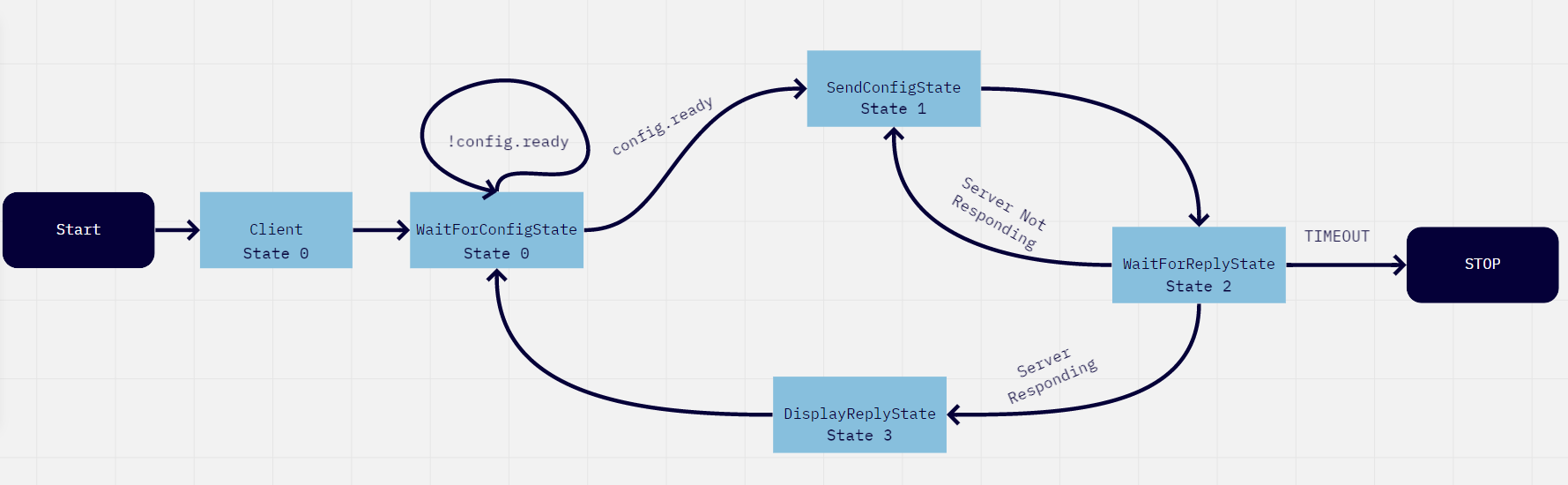


1. **Modelul State Machine**

Aplicația prezintă de asemenea un model de *state machine* reprezentat de 4 clase în care clientul se poate afla la un moment dat.:

1. **WaitForConfigState** – Clientul așteaptă configurarea pachetelor/mesajelor care vor fi trimise către server. În acest moment, clientul este ca și stare curentă în 0. Atunci când utilizatorul va semnala că procesul de configurare a luat sfârșit prin apăsarea butonului “SEND”, flagul config.ready va returna true și se va putea intra în state 1.
2. **SendConfigState –** Clientul se află în starea 1. Se face broadcast serverului. Se trimite pachetul format în stateul anterior folosind socket. Se intra in starea 2.
3. **WaitForReplyState –** Clientul este in starea 2. Se asteapta confirmarea de la server. În cazul în care serverul nu răspunde, clientul reintră în starea 1 unde va încerca să trimită pachetul către server. După un număr finit de ori soldat cu lipsa unui semnal din partea serverului, clientul se oprește. Dacă totul este bine se trece în starea 3.
4. **DisplayReplyState –** Clientul primește de la server datele cerute de acesta. Ele sunt afișate la ecran. Se intră în starea 0.

Starea inițială a serverului, state 0 este atribuită în constructorul clasei Client.



1. **Configurarea pachetelor**

Pachetele sunt clase ce au ca și membri toate câmpurile unui mesaj DHCP stocate în biți. Clasele principale care se ocupa de configurarea pachetelor sunt **Package** și **PacketManager**.

PacketManager face legătura dintre interfață și client. Astfel, această clasă prezintă 2 metode fundamentale: **convertToPackets** ce transformă alegerile făcute de utilizator din interfață într-un șir de biți corespunzători fiecărui câmp din pachetul transmis; **convertToDisplay** care actualizează într-un mod mai ușor de citit câmpurile afișate la interfață și de asemenea actualizează IP-ul oferit de server clientului, atât la interfață, cât și în fișierul text: **IP\_History.txt**.

De menționat este faptul că până când nu se vor converti alegerile făcute de către utilizator în pachete, Clientul se va afla în starea 0: WaitForConfigState.

1. **User Interface**

Interfața este făcută cu ajutorul librăriei **PyQT5** și servește ca modul principal de interacțiune cu utilizatorul. Utilizatorului îi sunt permise alegerea unui tip de mesaj pe care dorește să îl trimită serverului și, în funcție de mesaj o serie de opțiuni precum adresa IP sau alte date pe care dorim să le primim de la server. Fiecare optiune va adăuga un set de ID-uri stringului *options* care va putea ulterior fi transformat de **PacketManager** într-un șir de octeți corespunzători pachetului.

1. **Clientul**

Clientul reprezintă clasa fundamentală care funcționează în paralel cu interfața pe un thread diferit. Acesta va conține un pachet și va parcurge prin cele 4 stări în funcție de cerințele utilizatorului. El comunică folosind socket cu serverul.

1. **BIBLIOGRAFIE**

<https://www.tarunz.org/~vassilii/TAU/protocols/dhcp/toc.htm>

<https://www.digitalcitizen.ro/dhcp/>

<https://www.efficientip.com/glossary/dhcp-option/>

<https://tools.ietf.org/html/rfc2131>

Computer Networks Andrew S. Tanenbaum David J. Wetherall Fifth Edition