

# PROTOCOALE DE COMUNICATIE: Tema #3

## Client DNS

Termen de predare: 20 MAI 2018

Responsabili tema:  
Denisa SANDU, Dorinel FILIP, Vlad POSTOACA, Florin POP

### Obiectivele temei

Aceasta tema are ca obiectiv familiarizarea cu sistemul numelor de domenii in Internet (DNS) si cu modul in care functioneaza protocolul corespunzator. Se cere scrierea unui program C care sa permita interogarea unui server de DNS folosind specificatiile protocolului (RFC 1035 si RFC 1034).

### Client DNS

Clientul DNS implementat pentru aceasta tema este un program C/C++ care functioneaza ca un resolver DNS, folosind cereri pentru interogarea unor servere DNS. Programul va primi ca argument din linia de comanda un nume de domeniu sau o adresa IP si va afisa o serie de informatii despre acesta / aceasta. Pentru a obtine informatiile se va interoga serverul de DNS, folosind UDP sau TCP ca protocol de transport, si formandu-se mesaje de interogare dupa cum se specifica in sectiunea 4.1 din RFC 1035:

```
+-----+
|      Header      |
+-----+
|      Question    | the question for the name server
+-----+
|      Answer      | RRs answering the question
+-----+
|      Authority   | RRs pointing toward an authority
+-----+
|      Additional  | RRs holding additional information
+-----+
```

Se cere ca programul client sa realizeze urmatoarele tipuri de interogari:

- A (Host Address),
- MX (Mail Exchange),
- NS (Authoritative Name Server),
- CNAME (the canonical name for an alias),
- SOA (Start Of a zone of Authority),
- TXT (Text strings) - *daca in linia de comanda s-a dat un nume de domeniu,*
- PTR (Domain Name Pointer) - *pentru realizarea de reverse lookup, daca in linia de comanda s-a dat o adresa IP.*

Programul va primi doua argumente din linia de comanda: numele de domeniu / adresa IP, si un parametru care sa specifice ce tip de inregistrare ne intereseaza. De exemplu:

```
./dnsclient google.com A
```

```
./dnsclient 141.85.37.5 PTR
```

**Atentie!** Ordinea in care trebuie trimisi octetii intr-o variabila numerica este Big Endian, adica Network Byte Order. Pentru valorile numerice, fie le convertiti folosind functia `hton`, fie puneti octetii “de mana” in ordinea care trebuie.

Cererea catre server va trebui sa contina sectiunile “Header” si “Question”. Raspunsul de la server va contine atat sectiunile “Header” si “Question”, si un numar de sectiuni de tipul inregistrare (“Resource Record”). Se recomanda definirea de structuri pentru fiecare dintre aceste sectiuni. Reprezentarea acestor structuri este descrisa mai jos (este OBLIGATORIE utilizarea acestor definitii in rezolvarea temei de casa).

```
1  /* -- Query & Resource Record Type: -- */
2  #define A      1    /* IPv4 address */
3  #define NS     2    /* Authoritative name server */
4  #define CNAME  5    /* Canonical name for an alias */
5  #define MX     15   /* Mail exchange */
6  #define SOA    6    /* Start Of a zone of Authority */
7  #define TXT     16   /* Text strings */
8
9  /* -- Define DNS message format -- */
10 /* Header section format */
11 /**
12      0  1  2  3  4  5  6  7  8  9  0  1  2  3  4  5
13      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
14      |                                     ID                                     |
15      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
16      |QR|  Opcode  |AA|TC|RD|RA|   Z    |   RCODE   |
17      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
18      |                                     QDCOUNT                               |
19      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
20      |                                     ANCOUNT                               |
21      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
22      |                                     NSCOUNT                               |
23      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
24      |                                     ARCOUNT                               |
25      +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
26 */
27 typedef struct {
28     // schimba (LITTLE/BIG ENDIAN) folosind htons/ntohs
29     unsigned short id; // identification number
30
31     // LITTLE -> BIG ENDIAN: inversare 'manuala' ptr byte-ul 1 din flag-uri
32     unsigned char rd :1; // recursion desired
33     unsigned char tc :1; // truncated message
34     unsigned char aa :1; // authoritative answer
35     unsigned char opcode :4; // purpose of message
36     unsigned char qr :1; // query/response flag: 0=query; 1=response
37
38     // LITTLE -> BIG ENDIAN: inversare 'manuala' ptr byte-ul 2 din flag-uri
39     unsigned char rcode :4;
40     unsigned char z :3;
41     unsigned char ra :1;
42
43     // schimba (LITTLE/BIG ENDIAN) folosind htons/ntohs
44     unsigned short qdcount;
```

```

45 unsigned short amount;
46 unsigned short nscount;
47 unsigned short arcount;
48 } dns_header_t;
49
50 /* Question section format */
51 /**
52     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
53     +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
54     |                                     |
55     /                               QNAME /
56     /                               /
57     +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
58     |                               QTYPE |
59     +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
60     |                               QCLASS |
61     +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
62 */
63 typedef struct {
64     //qname variabil
65     unsigned short qtype;
66     unsigned short qclass;
67 } dns_question_t;
68
69 /* Resource record format */
70 /**
71     0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
72     +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
73     |                                     |
74     /                               NAME /
75     /                               /
76     |                               |
77     +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
78     |                               TYPE |
79     +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
80     |                               CLASS |
81     +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
82     |                               TTL |
83     |                               |
84     +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
85     |                               RDLENGTH |
86     +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
87     /                               RDATA /
88     /                               /
89     +---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
90 */
91 typedef struct {
92     //name variabil
93     unsigned short type;
94     unsigned short class;
95     unsigned int ttl;
96     unsigned short rdlength;
97     //rdata variabil;
98 } dns_rr_t;

```

Listing 1: Exemplu de structurare a mesajelor DNS.

Adresa IP a serverului de DNS la care se conecteaza programul va fi data intr-un fisier de configurare numit `dns_servers.conf`, aflat in acelasi director cu programul. Fisierul `dns_servers.conf` va contine mai

multe adrese de servere, cate una pe fiecare linie si posibile comentarii, pe o singura linie care incepe cu caracterul special '#'. Daca nu se pot obtine informatii de la primul server din fisier (adresa e incorecta, eroare la conectare in cazul TCP, raspunsul nu ajunge in timp util etc.), se trece la serverul urmator, etc.

```

1 # primul exemplu de servere
2 141.85.128.1
3 8.8.8.8
4
5 # server de rezerva (decomentati linia pentru a-l folosi)
6 # 64.22.110.162

```

Listing 2: Exemplu de fisier cu servere DNS folosit de clientul DNS.

Puteti folosi adresa oricarui server de DNS doriti. Cateva servere recomandate de noi:

- pub.pub.ro (141.85.128.1),
- ns.rnc.ro (192.162.16.21),
- ns-a.rnc.ro (192.162.16.31) sau
- ns.pcnet.ro (213.154.128.1),
- o lista cu servere DNS din Romania: <http://www.tackesoft.ro/domenii/index.php/ns>,
- serverul/servele de DNS pe care le utilizeaza calculatorul vostru de acasa, care pot fi gasite in fisierul /etc/resolv.conf.

*Formatul datelor de iesire prelucrate:* trebuie sa afisati inregistrările (Resource Record) din sectiunea “Answer”, dar si pe cele din “Authority” si “Additional”. Formatul de afisare cerut este asemanator cu cel intors de comanda:

```
host -v -t type name
```

In functie de tipul inregistrarii, formatul rezultatelor va fi:

```

1 NS:
2 Name  Class Type  NameServer
3
4 A:
5 Name  Class Type  Addr
6
7 MX:
8 Name  Class Type  Preference  MailExchange
9
10 CNAME:
11 Name  Class Type  PrimaryName
12
13 SOA:
14 Name  Class Type  PriName AuthoMailbox Serial Refresh Retry Expiration Minimum
15
16 TXT:
17 Name  Class Type  Text
18
19 PTR:
20 Name  Class Type  Addr

```

Listing 3: Formatul de afisare al raspunsului catre serverul DNS.

Toate mesajele trimise catre serverul DNS trebuie logate intr-un fisier numit `message.log` si trebuie sa fie in format hexa. Fiecare mesaj este logat doar o data, chiar daca interogati mai multe servere DNS in caz ca nu se primeste raspunsul in timp util (este la alegerea voastra valoarea timeout-ului).

Raspunsul primit va fi adaugat la fisierul `dns.log`. Un exemplu de `dns.log` gasiti aici (este obtinut in urma rularii programului de 2 ori: google.com A si yahoo.com MX) (server de DNS folosit: 8.8.8.8).

**Observatie:** Daca o sectiune nu are inregistrari, aceasta nu este afisata in `dns.log`.

```
1 ; 8.8.8.8 - google.com A
2
3 ;; ANSWER SECTION:
4 google.com. IN A 62.231.75.241
5 google.com. IN A 62.231.75.221
6 google.com. IN A 62.231.75.247
7 google.com. IN A 62.231.75.231
8 google.com. IN A 62.231.75.216
9 google.com. IN A 62.231.75.237
10 google.com. IN A 62.231.75.222
11 google.com. IN A 62.231.75.217
12 google.com. IN A 62.231.75.226
13 google.com. IN A 62.231.75.242
14 google.com. IN A 62.231.75.236
15 google.com. IN A 62.231.75.232
16 google.com. IN A 62.231.75.246
17 google.com. IN A 62.231.75.212
18 google.com. IN A 62.231.75.251
19 google.com. IN A 62.231.75.227
20
21
22 ; 8.8.8.8 - yahoo.com MX
23
24 ;; ANSWER SECTION:
25 yahoo.com. IN MX 1 mta5.am0.yahoodns.net.
26 yahoo.com. IN MX 1 mta6.am0.yahoodns.net.
27 yahoo.com. IN MX 1 mta7.am0.yahoodns.net.
```

Listing 4: Exemplu de fisier de log.

## Indicatii suplimentare

1. Pentru realizarea temei nu este neaparat nevoie sa parcurgeti ambele RFC-uri; dar trebuie sa cititi cel putin urmatoarele sectiuni din RFC 1035: 3.2 - 3.5, 4.1, 4.2;
  2. Trebuie tratata comprimarea mesajelor (vezi sectiunea 4.1.4 din RFC 1035);
  3. Trebuie sa tratati doar cazul cand serverul "stie" sa faca interogari recursive, ceea ce este valabil pentru marea majoritate a serverelor actuale. Astfel, in header-ul cererii veti seta bitul RD=1 si puteti presupune ca veti primi un raspuns corect de la server;
  4. Pagini utile ce descriu formatul mesajelor DNS:
    - <http://www.networksorcery.com/enp/protocol/dns.htm#Authority%20RR>
    - <http://www.zytrax.com/books/dns/ch15/>
- Daca folositi TCP ca protocol de transport: atentie la faptul ca fiecare mesaj trebuie prefixat cu 2 octeti in care se pune lungimea mesajului (vezi sectiunea 4.2.2 din RFC 1035);
5. Pentru reverse lookup: va trebui sa faceti un query de tip PTR, iar numele de domeniu din query va fi adresa IP "inversata" la care se adauga sirul `.in-addr.arpa`; de exemplu, daca vrem sa aflam numele de domeniu pentru IP-ul 1.2.3.4, vom face un query de tip PTR pentru `4.3.2.1.in-addr.arpa` (vezi sectiunea 3.5 din RFC 1035);
  6. Ca model de referinta, atat pentru modul de afisare pe ecran a informatiilor obtinute, cat si pentru verificarea corectitudinii informatiilor pe care le obtineti, puteti folosi comenzile `dig` sau `host`;
  7. Pentru debugging puteti folosi utilitarul Wireshark (<http://www.wireshark.org/>) care va arata continutul exact al mesajelor transmise sau primite prin retea; astfel puteti compara cererile DNS "fabricate" de voi cu cele realizate de exemplu prin comenzile `dig` sau `host`;

8. **IMPORTANT!** Nu aveti voie sa apelati functiile `getaddrinfo()`, `getnameinfo()`, `gethostbyname()` si `gethostbyaddr()`.

## Testare si notare

Arhiva trebuie sa aiba numele conform regulamentului si trebuie sa contina pe langa sursele C:

- Makefile cu target-urile **build**, **run** si **clean** obligatoriu
- README in care sa se specifice modul de implementare a temei

Nerespectarea cerintei de mai sus conduce la necorectarea temei de catre asistent. Corectarea se va realiza automat: se vor verifica fisierele `dns.log` si `message.log`.