

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

PROJEKT DO PŘEDMĚTU ISA

# DNS resolver

*Autor:*

Hynek Bernard  
xberna16

*Datum:* 6.11. 2019

# Obsah

<b>1</b>	<b>Zadání</b>	<b>2</b>
<b>2</b>	<b>Problematika</b>	<b>2</b>
2.1	DNS Protokol . . . . .	2
2.1.1	Popis . . . . .	2
2.1.2	Funkce resolveru . . . . .	2
<b>3</b>	<b>Implementace</b>	<b>3</b>
3.1	Popis . . . . .	3
3.2	Vlastní úpravy . . . . .	3
3.2.1	inverse query . . . . .	3
3.2.2	server zadaný adresou . . . . .	3
<b>4</b>	<b>Demonstrace použití</b>	<b>4</b>
4.1	Neexistující doménové jméno . . . . .	4
4.2	Existující doménové jméno . . . . .	6
<b>5</b>	<b>Literatura</b>	<b>7</b>

# 1 Zadání

Napište program dns, který bude umět zasílat dotazy na DNS servery a v čitelné podobě vypisovat přijaté odpovědi na standardní výstup. Sestavení a analýza DNS paketů musí být implementována přímo v programu dns. Stačí uvažovat pouze komunikaci pomocí UDP.

## 2 Problematika

Je zapotřebí vytvořit resolver komunikující protokolem DNS, ten je specifikovaný v RFC1035 s rozšířením IPv6 v RFC3596 a udává nám veškerý formát komunikace.

### 2.1 DNS Protokol

#### 2.1.1 Popis

DNS je hierarchický systém doménových jmen který má primárně za úkol převádět IP adresy a doménová jména. Protokol používá port 53 a adresy jsou vyjádřeny pomocí IPv4(32bitů pro A záznam) nebo IPv6(128bitů pro AAAA záznam). Servery DNS jsou organizovány hierarchicky, stejně jako jsou hierarchicky tvořeny názvy domén. DNS servery umí komunikovat s použitím více typů záznamů, ale adresy a domény jsou nejrozšířenější. Mezi další používané záznamy patří například NS - autoritativní name server, MX - poštovní výměna (Mail Exchange) a TXT - textové řetězce.

#### 2.1.2 Funkce resolveru

Resolver má za úkol zpracovat uživatelský vstup a optat se zadaného serveru na jeho záznamy, nebo server zažádat o rekurzivní vyhledání záznamu. Pokud se ptáme serveru na záznam nerekurzivně, vrátí nám nejlepší odpověď kterou v danou chvíli dokáže. Pokud má server relevantní data v cache, vrátí tyto záznamy, jinak odkáže uživatele na root server, nebo nejbližší autoritativní server DNS zóny který zná. Pokud se ptáme rekurzivně a server tuto funkci podporuje, server vždy uživateli vrátí odpověď nebo chybu o nenalezení. Pokud server nezná odpověď, započne rekurzivně vyhledávat od root

serverů, dokud nenalezne autoritativní name server který drží záznam hledaný uživatelem.

## 3 Implementace

### 3.1 Popis

Program je rozdělený na třídy `main` a `DNSPacket`. V `DNSPacket` je celá logika stavění, přijímání a výpisu packetu, v `main` je zajištěna komunikace se serverem a parsování argumentů.

### 3.2 Vlastní úpravy

#### 3.2.1 inverse query

O inverzní dotaz nežádá program skrze `OPCODE` inverse query jak je specifikováno v RFC1035, ale zasílá PTR dotaz s invertovanou adresou. V kódu je převod adresy prováděn funkcí `PrepareReverseString(address)`, která je volána při tvoření packetu ve funkci `SetQuestion`. Důvod je vydání RFC3425 které upravuje inverzní dotazy a díky němu je RFC1035 zastaralé.

#### 3.2.2 server zadaný adresou

Pokud se parametr server zadá programu v textovém tvaru (doménové jméno), provede se systémové volání `getaddrinfo` které přeloží adresu dns serveru.

## 4 Demonstrace použití

### 4.1 Neexistující doménové jméno

Při spuštění programu žádáme o přeložení neexistující domény nerekurzivně, server vrátí odkaz na ROOT servery

```
./dns -s 192.168.0.1 neexistujicidomena.neex
```

--Odpověď--

Authoritative: No, Recursive: No, Truncated: No

Question section (1)

neexistujicidomena.neex, A, IN

Answer section (0)

Authority section (13)

<<ROOT>>, IN, TTL: 82612, f.root-servers.net

<<ROOT>>, IN, TTL: 82612, b.root-servers.net

<<ROOT>>, IN, TTL: 82612, m.root-servers.net

<<ROOT>>, IN, TTL: 82612, h.root-servers.net

<<ROOT>>, IN, TTL: 82612, c.root-servers.net

<<ROOT>>, IN, TTL: 82612, l.root-servers.net

<<ROOT>>, IN, TTL: 82612, j.root-servers.net

<<ROOT>>, IN, TTL: 82612, i.root-servers.net

<<ROOT>>, IN, TTL: 82612, k.root-servers.net

<<ROOT>>, IN, TTL: 82612, e.root-servers.net

<<ROOT>>, IN, TTL: 82612, d.root-servers.net

<<ROOT>>, IN, TTL: 82612, g.root-servers.net

<<ROOT>>, IN, TTL: 82612, a.root-servers.net

Additional section (11)

a.root-servers.net, A, IN, TTL: 16393, 198.41.0.4

a.root-servers.net, AAAA, IN, TTL: 96984, 2001:503:ba3e::2:30

b.root-servers.net, A, IN, TTL: 96984, 199.9.14.201

b.root-servers.net, AAAA, IN, TTL: 96984, 2001:500:200::b

c.root-servers.net, A, IN, TTL: 96984, 192.33.4.12

c.root-servers.net, AAAA, IN, TTL: 96984, 2001:500:2::c

d.root-servers.net, A, IN, TTL: 96984, 199.7.91.13

d.root-servers.net, AAAA, IN, TTL: 96984, 2001:500:2d::d

e.root-servers.net, A, IN, TTL: 346813, 192.203.230.10

```
e.root-servers.net, AAAA, IN, TTL: 96984, 2001:500:a8::e  
f.root-servers.net, A, IN, TTL: 96984, 192.5.5.241
```

Při spuštění rekurzivně, vypíše program chybu

```
./dns -s 192.168.0.1 neexistujicidomena.neex -r
```

```
--Odpověď--
```

Chyba komunikace! Hledané zařízení neexistuje

## 4.2 Existující doménové jméno

```
./dns -r -s kazi.fit.vutbr.cz www.ietf.org
```

--Odpověď--

Authoritative: No, Recursive: Yes, Truncated: No

Question section (1)

www.ietf.org, A, IN

Answer section (3)

www.ietf.org, CNAME, IN, TTL: 300, www.ietf.org.cdn.cloudflare.net

www.ietf.org.cdn.cloudflare.net, A, IN, TTL: 300, 104.20.1.85

www.ietf.org.cdn.cloudflare.net, A, IN, TTL: 300, 104.20.0.85

Authority section (0)

Additional section (0)

## 5 Literatura

### Reference

- [1035] RFC1035: *DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION*.  
<https://tools.ietf.org/html/rfc1035>
- [3425] RFC3425: *Obsoleting IQUERY*  
<https://tools.ietf.org/html/rfc3425>
- [3425] RFC3596: *DNS Extensions to Support IP Version 6*  
<https://tools.ietf.org/html/rfc3596>