



Manuál

## **DNS resolver**

Síťové aplikace a správa sítí

Marcin Sochacki (xsocha03)

20. listopadu 2023

# Obsah

1	Úvod do problematiky	3
1.1	DNS	3
1.2	Funkce DNS	3
2	Implementace	3
2.1	Hlavičkový soubor	3
2.2	Funkce <i>parseCommandLine</i>	4
2.3	Funkce <i>initDNSHeader</i>	4
2.4	Funkce <i>extractDomainName</i>	4
2.5	Funkce <i>printQuestionSection</i>	4
2.6	Funkce <i>printData</i>	5
2.7	Funkce <i>reverseIPinAddrArpa</i>	5
2.8	Funkce <i>performDnsQuery</i>	5
3	Spuštění	5
4	Testování	6
5	Příklady výstupů	6
6	Rozšíření	7
7	Nedostatky	7

# 1 Úvod do problematiky

Zadáním bylo implementovat jednoduchý DNS resolver v jazyce C/C++, který umožňuje odesílat DNS dotazy na servery a zpracovávat odpovědi. Program by měl podporovat různé parametry, které umožňují specifikovat typ dotazů, server pro dotazování a další vlastnosti.

## 1.1 DNS

DNS (Domain Name System) je klíčovým prvkem síťové infrastruktury, který překládá lidsky čitelná doménová jména na IP adresy a umožňuje tak identifikaci síťových zařízení. DNS hraje klíčovou roli při usnadňování přístupu k webovým stránkám a dalším síťovým službám.

## 1.2 Funkce DNS

Při komunikaci přes internet často zadáváme doménová jména (např. `www.vut.cz`), ale pro síťovou komunikaci jsou potřebné IP adresy. DNS tedy slouží k překladu doménových jmen na IP adresy a umožňuje tak správné směrování síťového provozu.

# 2 Implementace

Implementace obsahuje program v jazyce C++ pro provedení DNS dotazu. Program podporuje několik přepínačů, které umožňují konfiguraci dotazu, včetně specifikace serveru, typu dotazu (A nebo AAAA), rekurze, a reverzního dotazu. Dále program zpracovává odpovědi na dotaz a vypisuje relevantní informace, jako jsou záznamy v sekci otázek, odpovědi, autoritativních a dodatečných záznamů.

## 2.1 Hlavičkový soubor

Soubor `dns.hpp` obsahuje deklarace struktur a funkcí. Jsou tam taky definovány makra pro různé typy DNS záznamů, jako jsou A, AAAA, PTR, CNAME, SOA, NS. *DNS\_OPTIONS* je struktura sloužící k uchování nastavení pro provedení DNS dotazu. Další strukturou je *DNS\_HEADER*, reprezentující hlavičku DNS zprávy, která obsahuje informace o identifikátoru zprávy, typ operace (dotaz nebo odpověď), autoritativnosti serveru a další. *QUESTION* reprezentuje část DNS zprávy, která obsahuje informace o dotazu, jako je typ dotazu (*qtype*) a třída dotazu (*qclass*). *QUERY* je struktura, která obsahuje informace o dotazu včetně názvu dotazované entity, typu dotazu a třídy dotazu. *R\_DATA* reprezentuje část DNS záznamu obsahující informace o typ záznamu (*type*), třídě záznamu (*class*), Time To Live (*tll*) a délce dat záznamu (*rdlength*). *RES\_RECORD* představuje část DNS odpovědi nebo autoritativní sekce, která obsahuje informace o záznamu, jako je název (*name*), data záznamu (*rdata*) a strukturu *R\_DATA*.

## 2.2 Funkce parseCommandLine

Funkce zpracovává argumenty příkazového řádku a vrací nastavení pro provedení DNS dotazu. Inicializuje strukturu *DNS\_OPTIONS*, která obsahuje informace o zpracovaných možnostech, jako je žádoucí rekurze, použití reverzního dotazu, případné použití záznamu AAAA, adresa DNS serveru, port a adresa dotazu. Funkce provádí kontrolu správného počtu argumentů a detekci možností příkazového řádku. Pokud dojde k chybě, funkce vypisuje odpovídající chybové zprávy na standardní chybový výstup a ukončuje program s chybovým kódem. Jako dotazovanou adresu se považuje první argument, který se neshoduje se zadanými možnostmi (flagy, server, port). Pokud je takových argumentů víc, následující argument je považován jako chybný. Pro konverzi doménového jména serveru na IP adresu serveru byla využita knihovna *<netdb.h>* a funkce *gethostbyname*, jelikož tato knihovna byla povolena v souladu s požadavky zadání.

## 2.3 Funkce initDNSHeader

Funkce slouží k inicializaci DNS hlavičky. Tato hlavička je klíčovou součástí DNS zpráv, která obsahuje informace o typu a délce zprávy, počtu dotazů, odpovědí a dalších důležitých parametrech. Funkce zajistí, že vytvořená DNS hlavička má požadované počáteční hodnoty.

## 2.4 Funkce extractDomainName

Funkce má za úkol získat čitelnou podobu doménového jména z DNS dotazu v binárním formátu. DNS zprávy používají kompaktní reprezentaci pro zakódování doménových jmen, a tato funkce provádí dekompresi této reprezentace, čímž umožňuje získat srozumitelné doménové jméno.

## 2.5 Funkce printQuestionSection

Funkce slouží k výpisu informací o sekci otázky DNS zprávy. Tato sekce obsahuje dotaz na konkrétní doménové jméno a typ záznamu. Zjišťuje typ záznamu pomocí kódu v poli *qtype* a přiřazuje mu odpovídající textový popis. Následně vypisuje informace o dotazu, jako je adresa serveru (*address*), typ záznamu (*recordType*) a třída dotazu (*IN* nebo *Other*).

## 2.6 Funkce printData

Funkce postupně čte a vypisuje datové části odpovědi DNS zprávy. Nejprve inicializuje proměnné pro čtení (*reader*) a stanoví ukazatel na konec datové části (*stop*). Následně vstupuje do cyklu, kde postupně čte délky a vypisuje jednotlivé záznamy v datové části, dokud nedojde na konec nebo nedosáhne počtu očekávaných záznamů specifikovaných v hlavičce DNS zprávy. Každý záznam obsahuje informace o délce (*rdlength*) a samotná data odpovědi, která jsou vypsána.

## 2.7 Funkce reverseIPinAddrArpa

Funkce slouží k vytvoření reverzního DNS záznamu na základě vstupní IP adresy. Vstupní IP adresa je předpokládána v textovém formátu a je zpracována do formátu in-addr.arpa. Nejprve se rozdělí vstupní IP adresa na čtyři části, které jsou následně využity k sestavení reverzního DNS záznamu. Pokud se podaří úspěšně načíst všechny části IP adresy, provede sestavení reverzního záznamu ve formátu "n.n.n.n.in-addr.arpa", kde n představuje jednotlivé části IP adresy. V případě neplatné IP adresy funkce vypíše chybové hlášení na standardní chybový výstup a ukončí program s návratovým kódem 1.

## 2.8 Funkce performDnsQuery

Funkce nejprve vytváří a inicializuje DNS hlavičku a otázku pomocí předaných argumentů. Následně odesílá DNS dotaz na specifikovaný DNS server a přijímá odpověď. Po obdržení odpovědi zpracovává DNS hlavičku, extrahuje informace o dotazu a odpovědi a vypisuje relevantní informace, jako je autoritativnost, rekurze, a další. Dále volá funkce printQuestionSection a printData pro analýzu a výpis sekcí.

## 3 Spuštění

Nejdřív je potřeba použít příkaz *make* pro vytvoření spustitelného programu *dns*. Následně aplikaci spouštíme v tvaru: `./dns [-r] [-x] [-6] -s server [-p port] adresa`, kde:

- -r: Požadována rekurze (Recursion Desired = 1), jinak bez rekurze.
- -x: Reverzní dotaz místo přímého.
- -6: Dotaz typu AAAA místo výchozího A.
- -s: IPv4 adresa nebo doménové jméno serveru, kam se má zaslat dotaz (aplikace nepodporuje IPv6 adresu, jako adresu serveru). Musí obsahovat maximálně 100 znaků.
- -p port: Číslo portu, na který se má poslat dotaz, výchozí 53.
- adresa: IPv4 adresa nebo doménové jméno dotazované adresy (aplikace nepodporuje IPv6 adresu, jako dotazovanou adresu pro všechny záznamy). Musí obsahovat maximálně 100 znaků.

Každý argument může být zapsán maximálně jednou v jednom příkazu. Není povolena kombinace argumentů -6 a -x, protože by došlo k konfliktu záznamů PTR a AAAA. V případě neplatného počtu parametrů, neplatných argumentů nebo nepovolených kombinací program vrátí návratový kód 1. Jako port se považuje argument, který se nachází za argumentem -p. Musí to být celé číslo od 1 do 65535. Parametry mohou být v libovolném pořadí. Pokud program ukončíme pomocí signálu *SIGINT*, uzavře se soket a dostaneme návratový kód 0.

## 4 Testování

Pro kontrolu úniků paměti byl využit nástroj Valgrind. Díky této analýze bylo dosaženo vyladěné a spolehlivé implementace bez úniků paměti. Pro ověření správnosti funkcí a chování aplikace byly napsány automatické testy. V první části testy kontrolují nejrozličnější chybné vstupy. Porovnávají výstup aplikace s očekávaným výstupem. V druhé části testují se záznamy AAAA a PTR. Testy porovnávají Answer sekce aplikace a Answer sekce po zavolání příkazu *dig* na stejnou dotazovanou adresu. Automatické testy byly implementovány v jazyce Python s využitím knihovny *subprocess*. Kromě automatických testů bylo provedeno mnoho manuálních testů pomocí nástroje *dig*. Testy se spouští pomocí příkazu *make test*. Pro úspěšné provedení testů je nutné mít k dispozici příkaz *dig* na testovacím stroji.

## 5 Příklady výstupů

1. Autoritativní a dodatečná sekce:

```
xsocha03@merlin: ~/vut/3BIT/isa/projekt$ ./dns -s kazi.fit.vutbr.cz google.com
Authoritative: No, Recursive: No, Truncated: No
Question section (1)
  google.com., A, IN
Answer section (1)
  google.com., A, IN, 201, 142.251.36.78
Authority section (4)
  google.com., NS, IN, 40903, ns2.google.com.
  google.com., NS, IN, 40903, ns1.google.com.
  google.com., NS, IN, 40903, ns4.google.com.
  google.com., NS, IN, 40903, ns3.google.com.
Additional section (8)
  ns1.google.com., A, IN, 40758, 216.239.32.10
  ns2.google.com., A, IN, 40758, 216.239.34.10
  ns3.google.com., A, IN, 40758, 216.239.36.10
  ns4.google.com., A, IN, 40758, 216.239.38.10
  ns1.google.com., AAAA, IN, 40758, 2001:4860:4802:0032:0000:0000:0000:000A
  ns2.google.com., AAAA, IN, 40758, 2001:4860:4802:0034:0000:0000:0000:000A
  ns3.google.com., AAAA, IN, 40758, 2001:4860:4802:0036:0000:0000:0000:000A
  ns4.google.com., AAAA, IN, 40758, 2001:4860:4802:0038:0000:0000:0000:000A
```

## 2. Rekurzivní AAAA dotaz:

```
xsocha03@merlin: ~/vut/3BIT/isa/projekt$ ./dns -s 147.229.8.12 rhino.cis.vutbr.cz -6 -r
Authoritative: No, Recursive: Yes, Truncated: No
Question section (1)
  rhino.cis.vutbr.cz., AAAA, IN
Answer section (1)
  rhino.cis.vutbr.cz., AAAA, IN, 300, 2001:67c:1220:e000::93e5:30a
Authority section (0)
Additional section (0)
```

## 3. Reverzní dotaz:

```
xsocha03@merlin: ~/vut/3BIT/isa/projekt$ ./dns -s 147.229.8.12 -x 77.75.79.222 -r
Authoritative: No, Recursive: Yes, Truncated: No
Question section (1)
  222.79.75.77.in-addr.arpa, PTR, IN
Answer section (1)
  222.79.75.77.in-addr.arpa., PTR, IN, 30, www.seznam.cz.
Authority section (0)
Additional section (0)
```

## 4. Autoritativní dotaz:

```
xsocha03@merlin: ~/vut/3BIT/isa/projekt$ ./dns -s 147.229.8.12 www.fit.vut.cz
Authoritative: Yes, Recursive: No, Truncated: No
Question section (1)
  www.fit.vut.cz., A, IN
Answer section (1)
  www.fit.vut.cz., A, IN, 14400, 147.229.9.26
Authority section (4)
  fit.vut.cz., NS, IN, 14400, guta.fit.vutbr.cz.
  fit.vut.cz., NS, IN, 14400, kazi.fit.vutbr.cz.
  fit.vut.cz., NS, IN, 14400, rhino.cis.vutbr.cz.
  fit.vut.cz., NS, IN, 14400, gate.feec.vutbr.cz.
Additional section (0)
```

# 6 Rozšíření

- Aplikace podporuje nevyžadovaný ze zadání záznam NS v autoritativní sekci a je schopna zpracovávat a vypisovat informace obsažené v tomto záznamu.
- Aplikace podporuje nevyžadovaný ze zadání záznam SOA v autoritativní sekci a je schopna zpracovávat a vypisovat všechny informace obsažené v tomto záznamu.

# 7 Nedostatky

- Aplikace nepodporuje IPv6 adresu jako adresu serveru.
- Aplikace nepodporuje IPv6 adresu jako dotazovací adresu.

## Literatura

- [1] Přednáška ISA – základní informace o DNS. Dostupné z: [https://www.youtube.com/watch?v=BV5AcujCkyM&list=PL\\_eb8wrKJwYvSZY8TdV2IQ\\_UdDzw92hgk&index=7&ab\\_channel=P%C5%99edn%C3%A1%C5%A1kyFITVUT](https://www.youtube.com/watch?v=BV5AcujCkyM&list=PL_eb8wrKJwYvSZY8TdV2IQ_UdDzw92hgk&index=7&ab_channel=P%C5%99edn%C3%A1%C5%A1kyFITVUT)
- [2] RFC 1035 – struktura a fungování DNS. Dostupné z: <https://datatracker.ietf.org/doc/html/rfc1035>
- [3] Struktury pro DNS. Dostupné z: <https://www.binarytides.com/dns-query-code-in-c-with-linux-sockets/>
- [4] Manuálová stránka pro *inet\_pton*. Dostupné z: [https://man7.org/linux/man-pages/man3/inet\\_pton.3.html](https://man7.org/linux/man-pages/man3/inet_pton.3.html)
- [5] Manuálová stránka pro *inet\_ntoa*. Dostupné z: [https://man7.org/linux/man-pages/man3/inet\\_ntoa.3.html](https://man7.org/linux/man-pages/man3/inet_ntoa.3.html)
- [6] Manuálová stránka pro *gethostbyname*. Dostupné z: <https://man7.org/linux/man-pages/man3/gethostbyname.3.html>
- [7] Konvertování mezi formáty. Dostupné z: <https://man7.org/linux/man-pages/man3/ntohs.3.html>