Vysoké učení technické v Brně
Fakulta informačních technologií

DNS Resolver

ISA Projekt

Listopad 2019 Jan Chaloupka (xchalo16)

Obsah

1 Úvod do problematiky

1.1 DNS paket

Detailní informace o DNS paketu jsou popsány v dokumentu RFC 1035[1], ze kterého se v tomto projektu vychází. Navíc se v projektu využívají záznamy a struktury související s IPv6, které jsou definovány v dokumentu RFC 3596[2]

1.1.1 Přenos

Pro komunikaci využívá DNS protokol UDP, v základu na portu 53. Není tak zaručeno, že zpráva dorazí bez chyby. Zpráva je kódována ve formátu big-endian, proto je potřeba při implementaci použít standardní funkce pro konverzi mezi endiany zařízení a síťové vrstvy.

1.1.2 Struktura

Samotný DNS paket je rozdělen na pět základních částí. Detailně je DNS zpráva popsána v RFC 1035, sekce 4.

Na začátku každé DNS zprávy je vždy přítomná **hlavička**. Obsahuje základní informace jako ID, stavový kód, možnosti dotazu a údaje o počtu záznamů v jednotlivých sekcí. Po hlavičce následuje **sekce dotazů** od klienta. Každý dotaz obsahuje název domény ke které se dotaz vztahuje a typ dotazu. Odpovědi serveru na dotazy klienta jsou v **sekci odpovědí**. Odpověď je struktura **záznam**, stejná struktura je využita i v sekci **autorit** a sekci **dalších záznamů**.

1.1.3 Komprese doménových jmen

Aby zpráva zabírala co nejméně místa, je využita základní komprese doménových jmen. Pokud se část doménového jména již před tím ve zprávě vyskytla, je na tuto část uložen ukazatel. Ukazatel se může vyskytovat v kterékoliv části zprávy místo specifikace délky labelu jména. Ukazatel je ve formátu `11XX XXXX XXXX XXXX`, kde **X** je index doménového jména od začátku zprávy. Pokud tedy je dělka labelu větší než 191, jedná se o ukazatel. Více o kompresi v RFC1035, sekce 4.1.4.

1.2 Reverzní DNS dotaz

Reverzní záznam slouží k přiřazení doménového jména k IP adrese. Tento záznam je označený jako `PTR` a obsahuje v datové sekci doménové jméno. Pro získání tohoto záznamu je třeba poslat dotaz typu `PTR` s IP adresou ve speciálním tvaru jako doménové jméno.

[1] Celé znění dokumentu RFC 1035: `www.ietf.org/rfc/rfc1035`
[2] Celé znění dokumentu RFC 3596: `www.ietf.org/rfc/rfc3596`

1.2.1 Převedení IPv4 adresy do PTR formátu

Adresa typu IPv4 se převede obrácením pořadí jednotlivých oktetů a přidáním řetězce `.in-addr.arpa.` na konec (RFC1035 sekce 3.5).

Příklad: `203.99.78.77.in-addr.arpa.` odpovídá IP adrese `77.78.99.203`

1.2.2 Převedení IPv6 adresy do PTR formátu

Převedení IPv6 adresy funguje na podobném principu jako v4 adresy. Adresa se rozdělí po půl bajtech (jeden hexa znak) a zapíše se v plné délce v obráceném pořadí. Znaky jsou dděleny tečkou. Nakonec se za adresu přidá řetězec `.ip6.arpa.` (RFC3596 sekce 2.5).

Příklad: Adrese `2001:67c:1220:809::93e5:917` odpovídá zápis `7.1.9.0.5.e.3.9.0.0.0.0.0.0.0.0.9.0.8.0.0.2.2.1.c.7.6.0.1.0.0.2.ip6.arpa.`

2 Návrh aplikace

Aplikace je rozdělena na čtyři funkční celky, je využit objektový přístup. Jednotlivé části jsou navrhnuty tak, aby byly na sobě nezávislé a samostatně funkční. Části jako **DNS knihovna**, **UDP klient** a **Zpracování argumentů** je možné použít jako knihovny v jiném projektu.

2.1 DNS knihovna

Knihovna ulehčující práci s DNS zprávami, vytvořena především za pomoci referenčního standardu RFC 1035. Obdoba C knihovny **resolv.h**, s tím rozdílem, že je zde využito možností jazyka C++ a celá knihovna je objektově založená.

Obsahuje objekty pro vytvoření DNS zprávy, dotazu, záznamu a metody pro převod těchto objektů z a do binární podoby. Při konverzi není použita struktura s pevně daným obsahem, ale výsledné bajty se vytváří bitovými a zapisováním do pole bajtů. Toto řešení je zvoleno proto, že dle mého názoru je toto řešení robustnější s ohledem na různé endiany systémů, a v případě že by se někdo v budoucnu rozhodl refaktorovat tuto knihovnu, přeskládáním atributů se nezmění chování programu. Objekty obsahují i metody pro výpis jejich dat do čitelné podoby, toho lze využít pro výpis dat uživateli na standardní výstup programu.

Kromě objektů knihovna zahrnuje i výčty používané v DNS zprávě, jako například: typ zprávy, stavový kód, typ záznamu a k těmto výčtům je i funkce pro výpis kódů v čitelné podobě.

2.2 UDP klient

Jednoduchý UDP klient, který odešle zprávu na specifikovaný server a uloží do bufferu odpověď. Jedná se o zabalení BSD soketů pro snadnější používání. Součástí je i převod adres z řetězce do struktury adresy, stačí tak specifikovat jméno serveru, například `kazi.fit.vutbr.cz` a klient se o vše postará.

Zahrnuta je i podpora pro komunikaci se serverem IPv6 adresou. Pokud nastane chyba při převodu adresy nebo soketové komunikaci, je vyvolána výjimka. Stejně tak je vyvolána vyjímka, pokud server neodpoví do určitého intervalu. Kód je z části převzatý z manuálových stránek funkce `getaddrinfo`[3].

2.3 Zpracování argumentů programu

Pomocný modul pro zpracování argumentů DNS resolveru. Interně ke zpracování využívá POSIX verzi funkce `getopt`[4]. Atributy jsou psané přímo pro tento projekt, avšak s minimální změnou lze pro zpracování jiných argumentů. Funkčnost je jednoduchá, stačí předat argumenty programu a pokud jsou v pořádku, je vrácen objekt se získanými parametry.

[3]Man stránka funkce `getaddrinfo`: `man7.org/linux/man-pages/man3/getaddrinfo.3.html`
[4]Man stránka funkce `getopt`: `man.openbsd.org/getopt.3`

2.4 DNS Resolver

Hlavní část projektu, je to samotná logika programu. Pracuje s ostatními moduly a obsluhuje vstup a výstup programu. Získá možnosti programu z modulu pro zpracování argumentů, následně podle vstupních parametrů sestaví za pomocí vlastní DNS knihovny DNS zprávu s dotazem, oděšle tuto zprávu DNS serveru pomocí UDP klienta a odpověď převede na DNS zprávu, kterou vypíše uživateli na standardní výstup.

Toto všechno proběhne za předpokladu, že v procesu nenastala žádná výjimka. Pro obsluhu výjimek program obsahuje funkci, která výjimku zachytí a vypíše ji na standardní chybový výstup do uživatelem čitelné podoby.

3 Popis implementace

Zdrojový kód aplikace je psán v programovacím jazyce C++, konkrétně jeho standardu C++11. Je využit objektově orientovaný přístup.

Zdrojový kód knihovny DNS je v souborech: `dns_message.cpp` (Celá DNS zpráva), `dns_resource.cpp` (DNS záznam), `dns_question.cpp` (DNS dotaz), `dns_utils.cpp` (Pomocné struktury a funkce pro práci s DNS zprávou).

Zdrojový kód UDP klienta je v souboru `udpclient.cpp`, kód zpracování argumentů je v souboru `options.cpp` a samotné tělo programu je v souboru `main.cpp`.

3.1 Struktura adresáře projektu

- `obj/` Složka pro přeložené `.o` soubory
- `src/` Složka obsahující zdrojový kód a hlavičky
- `test/` Složka s automatickými blackbox testy
- `Makefile` Cíle pro program **GNU Make**
- `README` Základní informace o projektu
- `manual.pdf` Podrobná dokumentace projektu (tento soubor)

3.2 Seznam použitých knihoven

Aplikace využívá následující standardní knihovny jazyka C++[5]: `algorithm`, `cerrno`, `cstdint`, `cstdlib`, `cstring`, `ctime`, `iomanip`, `iostream`, `list`, `sstream`, `stdexcept`, `string` a `vector`.

Kromě toho je v projektu využito několika POSIX knihoven jazkya C[6], zejména pro práci s BSD sockety. Použité C POSIX knihovny jsou: `arpa/inet.h`, `netdb.h`, `netinet/in.h`, `sys/socket.h`, `sys/time.h` a `unistd.h`.

[5]Seznam standardních C++ knihoven: `en.cppreference.com/w/cpp/header`
[6]Seznam C POSIX knihoven: `pubs.opengroup.org/onlinepubs/9699919799/idx/head.html`

4 Informace o programu

4.1 Formát výpisu

Pokud program skončí úspěšně, vypíše na standardní výstup informace z DNS serveru v následujícím formátu:

```
Authoritative: XX, Recursive: XX, Truncated: XX
Question section (n)
  <Dotazy>
Answer section (n)
  <Záznamy>
Authority section (n)
  <Záznamy>
Additional section (n)
  <Záznamy>
```

Kde **Authoritative** značí, zda je DNS server autoritativní, **Recursive** značí, zda byl dotaz proveden rekurzivně (klient vyžádal rekurzivní zpracování a server toto podporuje) a **Truncated** značí, že se přijatá zpráva nevlezla do maximální velikosti DNS zprávy (512 bajtů) a byla oříznuta. Místo XX je dosazeno **Yes** nebo **No**.

Následuje výpis všech záznamů a dotazů. Za **n** je dosazen počet záznamů v dané sekci. V každé sekci je záznam odsazen dvěma mezerami a jeden záznam je na jeden řádek. Za **<Dotazy>** jsou dosazeny konkrétní dotazy, stejně tak pro **<Záznamy>**.

4.1.1 Formát dotazu

```
JMÉNO, TYP, TŘÍDA
```

JMÉNO je doménové jméno, **TYP** je typ dotazu, **TŘÍDA** je třída dotazu (vždy IN).

4.1.2 Formát záznamu

```
JMÉNO, TYP, TŘÍDA, TTL, DATA
```

JMÉNO je doménové jméno, **TYP** je typ dotazu, **TŘÍDA** je třída dotazu (vždy IN). **TTL** značí platnost záznamu v sekundách a **DATA** jsou data záznamu - jednotlivé položky dat jsou odděleny mezerou.

5 Návod na použití

Pořadí všech parametrů je libovolné. Program umožňuje zadávání parametrů více způsoby tak, jak je zvykem u jiných programů. Například `-s8.8.8.8 -6rp 53` je validní zápis argumentů programu.

```
dns  [-rx6tmc] -s server [-p port] adresa
```

5.1 Parametry

5.1.1 Typ dotazu

Pokud není specifikován typ dotazu, je jako výchozí použít A dotaz (získání IPv4 adres). V jednom dotazu (spoštění programu) **není povoleno kombinovat více typů dotazů**.

- `-6` IPv6 adresy (typ AAAA)
- `-t` Textové záznamy (typ TXT)
- `-m` E-mailové servery obsluhující doménu (typ MX)
- `-c` Aliasy domény (typ CNAME)
- `-x` Reverzní záznamy pro IP adresu (typ PTR)

5.1.2 Možnosti dotazu

- `-r` Signalizace serveru, že se dotaz má provést rekurzivně, pokud server tuto volbu podporuje
- `-s server` Adresa dotazovaného DNS serveru **(povinné)**
- `-p port` Port na kterém dotazovaný DNS server naslouchá (výchozí = 53)
- `adresa` Dotazovaná adresa ve formátu doménového jména **(povinné)**

  Pokud je dotaz na reverzní záznam (přepínač `-x`), očekává se místo doménového jména IPv4 nebo IPv6 adresa ve standardním tvaru

5.2 Příklady použití

5.2.1 Dotaz na IPv4 (A záznamy) domény

```
> dns -rs 8.8.8.8 www.fit.vutbr.cz
Authoritative: No, Recursive: Yes, Truncated: No
Question section (1)
  www.fit.vutbr.cz., A, IN
Answer section (1)
  www.fit.vutbr.cz., A, IN, 14144, 147.229.9.23
Authority section (0)
Additional section (0)
```

5.2.2 Reverzní dotaz na IP adresu

```
> dns -rx -s 8.8.8.8 147.229.9.23
Authoritative: No, Recursive: Yes, Truncated: No
Question section (1)
  23.9.229.147.in-addr.arpa., PTR, IN
Answer section (1)
  23.9.229.147.in-addr.arpa., PTR, IN, 10581, www.fit.vutbr.cz.
Authority section (0)
Additional section (0)
```

5.2.3 Dotaz na MX záznamy domény

```
> dns -rm -s 8.8.8.8 seznam.cz
Authoritative: No, Recursive: Yes, Truncated: No
Question section (1)
  seznam.cz., MX, IN
Answer section (2)
  seznam.cz., MX, IN, 76, 10 mx1.seznam.cz.
  seznam.cz., MX, IN, 76, 20 mx2.seznam.cz.
Authority section (0)
Additional section (0)
```