Dokumentace k projektu:
WHOIS tazatel

Adam Sedláček
xsedla1e@stud.fit.vutbr.cz

ISA 2019
VUT FIT v Brně

18.11.2019

Obsah

1 Whois protokol

Slouží k uchovávání údajů o majitelích IP adres a domén. Každý nadnárodní registrátor vlastní vede svůj WHOIS server (např. RIPE). Rozděleno je hiearchicky, kdy nejvyšší doménu má vždy daný registrátor a pod ní jsou další menší "podregistrátoři". Tento protokol běží na **portu 43** a jedná se o klient-server komunikaci.
Zejména byla snaha mít nějakou databázi, v které by se uchovávali informace o daných majitelích a nešlo tak jednoduše zneužít k nekalým činnostem. S tím se váže i poskytnout co nejvíce informací, které by mohli případně pomoci, při zjišťování informací o dané doméně.
Bohužel formát není nikterak standardizovaný, a občas může být menší problém získat dostatečné informace.

1.1 Whois zpráva

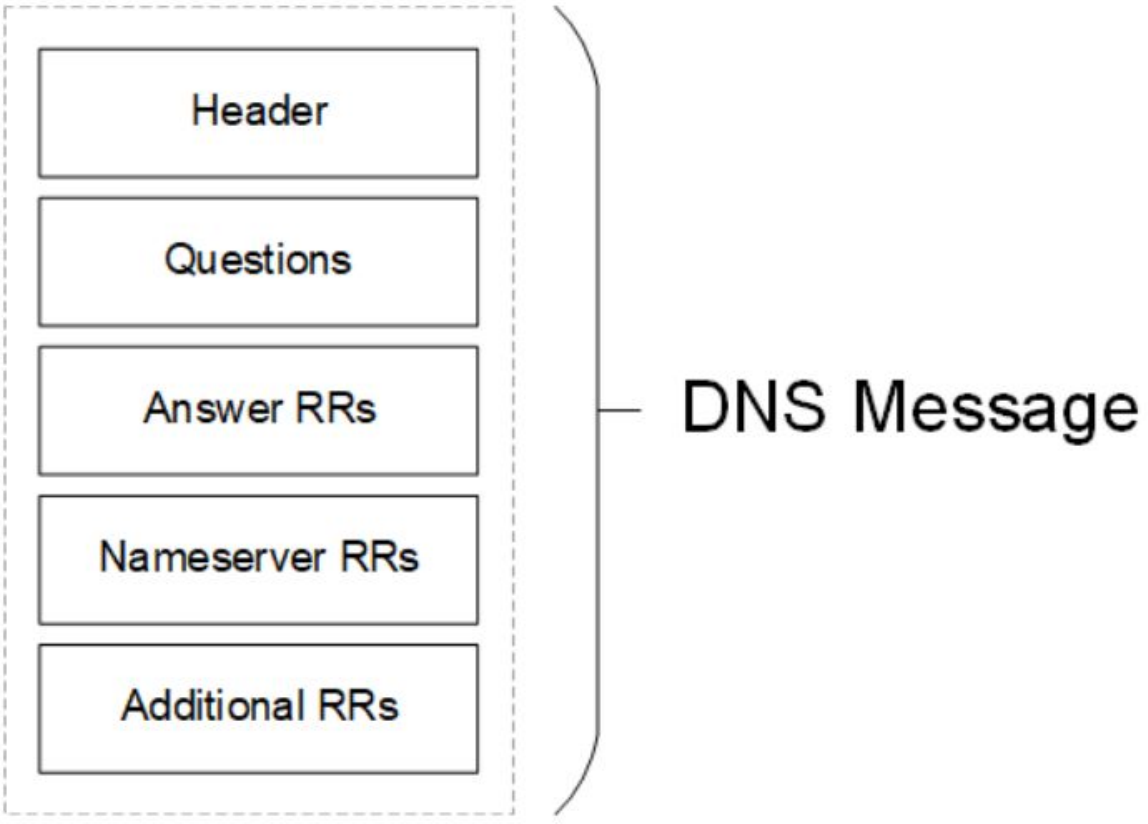
Plnou specifikaci můžeme najít v RFC https://tools.ietf.org/html/rfc3912.
Vesměs stačí otevřít správně nakonfigurovaný socket, do něj následně zapsat požadovaný údaj např. seznam.cz a ukončit znaky \r\n, server takovou zprávu zpracuje a pokud má daný záznam, tak je zapíše do socketu, který si vyčteme a údaje zpracujeme a následně vytiskneme.

2 Protokol DNS

DNS servery komunikují přes protokol UDP na **portu 53.** Je možné i komunikace protokolem TCP, která se využívá pro přenos příliš velkých zpráv. Maximální velikost zprávy pro UDP je 512 bajtů, pak tedy maximální délka doménového jména je 256 bajtů.

2.1 DNS zpráva

DNS servery s klienty i mezi sebou komunikují pomocí zpráv. Každá DNS zpráva sestává z hlavičky zprávy, sekce otázek, sekce odpovědí a případných dalších užitečných záznamů. Její délka je proměnlivá, maximálně však 512 bajtů.



2.2 DNS question

DNS “otázka“, specifikuje co za záznamy po daném serveru požadujeme. Tedy obsahuje následující:

- **Name** - Pole proměnlivé délky udávající dotazované doménové jméno v DNS formátu doménových jmen.

- **Type** - Typ dotazovaného záznamu - pro naše účely (A = 1; NS = 2; CNAME = 5; PTR = 12; AAAA = 28).

- **Class** - Třída dotazovaného záznamu, v našem případě vždy 1 - třída internetu.

Význam dat v poli **Resource Data** je závislé na typu záznamu, pro naše účely jsou důležité následující:

- A - RDATA obsahuje 4 bajty udávající IPv4 adresu.
- AAAA - RDATA obsahuje 16 bajtů udávající IPv6 adresu.
- CNAME - RDATA obsahující doménové jméno v DNS formátu udávající tzv. *Canonical Name*, neboli alias hledaného doménového jména.
- NS - RDATA obsahuje doménové jméno v DNS formátu name serveru pro hledané doménové jméno.
- SOA - RDATA obsahuje doménové jméno v DNS formátu primárního name serveru pro zónu autority a další informace
- PTR - RDATA obsahuje adresu, která se překládá na doménu nebo název hostitele.
- MX - RDATA obsahuje emailovou adresu pro daný DNS záznam.

3 Implementace

Program je napsán v jazyce C++, i když se pracuje hlavně s knihovny, které jsou primárně určené pro C, tak toto nečinilo žádné problémy. A využití vyššího jazyku se nakonec ukázalo jako správné rozhodnutí. Ulehčilo to práci zejména při ošetřování chyb a usnadnilo práci se stringy, také nebylo třeba využívat malloc funkce, takže ani nedochází k memory leakům z nepozornosti a zapomnětlivosti free.

Hlavní popis co které funkce/třída/metoda/struktura dělá by se mělo nacházet přímo v kódu. Údaje níže jsou jen obecné přiblížení.

- **Třída socket** - stará se o veškerou práci se socketem a o základní parsování odpovědi, tak aby byla více pro uživatele čitelná.
    - Connect - nastavuje daný socket, tak aby do něj šlo zapisovat/číst.
    - Write - zapisuje data do socketu
    - Read - čte data ze socketu
- **Pomocné funkce**, které využívám
    - trim_space
    - to_lowercase
    - find_refer
    - get_country
    - get_full_address
- **Parse_response** - parsuje danou odpověď, tak aby z ní "vytáhla" co nejvíce informací pomocí regexu, zde je "menší" problém, že odpověď není nikterak standardizovaná, a tudíž výpis někdy může být velmi, ale velmi omezený. Proto doporučuji spíše používat *-v true* přepínač.
- **Whois** - funkce pro tisk whois informací, dělá případné přesměrování na jiný whois server
- **Parametr_validation** - ověřuje zadané argumenty, zda splňují požadavky ze zadání.
- **Dns_query** - celá práce s DNS je ukrytá zde, všechny záznamy se dotazují/parsují/sestavují zde.
- **Dns** - pomocná funkce, která jen volá pro každý záznam ve vektoru dns_query, a následně tiskne výsledky.

3.1 Překlad/běh aplikace

Překlad - pomocí make (soubor je přibalen a je plně funkční).
Spuštění - tak jak je v zadání a nebo v README.md, případně pod přepínačem -h

Překlad + spuštění s *verbose false*

```
olok@olok:~/VUT/ISA$ make
g++ -std=c++17 isa-tazatel.cpp -lpcap -lresolv -o isa-tazatel
olok@olok:~/VUT/ISA$ ./isa-tazatel -q seznam.cz -w whois.nic.cz -v false
[Querying whois.nic.cz]
[Response from whois.nic.cz]

=== WHOIS ===
admin-c:      SEZNAM-CZ-AS-TECH
org:          Seznam.cz, a.s.
address:      Radlická 3294/10
address:      Praha 5
country:      CZ

=== DNS ===
NS:     ams.seznam.cz
NS:     ans.seznam.cz
A:      77.75.75.176
A:      77.75.75.172
AAAA:   2a02:598:4444:1::2
AAAA:   2a02:598:4444:1::1
SOA:    ans.seznam.cz
MX:     seznam.cz.              5M IN MX        20 mx2.seznam.cz.
MX:     seznam.cz.              5M IN MX        10 mx1.seznam.cz.
```

3.2 Omezení a rozšíření

Omezení spočívá pouze v nestihnutí rozšíření DNS PTR záznamu.
Také rozšíření pro změnu DNS serveru nebylo implementováno.

Moje implementace očekává od uživatele, že "ví co chce", tudíž neodstraňuji www předponu, ani se nesnažím dotazovat na IPv4, pak IPv6 a poté až na hostname. Uživatel si musí být vědom, co žádá.

Naopak rozšířeno o verbose switch, který se snaží dlouho odpověď trochu "vyzobat" a zobrazit užitečné informace.
Také automatické přesměrování na referenční whois server.

4 Literatura a pomocné odkazy

- RFC 954: NICNAME/WHOIS
- RFC 1580: Guide to Network Resource Tools
- RFC 1834: Whois and Network Information Lookup Service, Whois++
- RFC 3912: WHOIS protocol Specification
- https://docstore.mik.ua/orelly/networking_2ndEd/dns/ch15_02.htm
- https://www.geeksforgeeks.org/regex-regular-expression-in-c/